The Crytica Security Principles of Malware Detection

White Paper

Crytica Security
7655 Town Square Way
Suite 212
Reno, NV 89523

Website: CryticaSecurity.com
Email: info@cryticasecurity.com

Principles of Malware Detection

Contents

1. INTRODUCTION – MALWARE DETECTION IN A HOSTILE ENVIRONMENT

“If you cannot detect, you cannot protect!”

Malware detection is not as simple as sticking a thermometer into a roast to see if it ready. The challenge is more akin to that of detection of enemy incursions into a military camp in the midst of hostile territory. One must assume that everything outside the established perimeter is dangerous. One must assume that the enemy will do all they can to penetrate the perimeter and, once inside, inflict maximum damage. Therefore, one must assume that the enemy is not only outside the perimeter but may be inside as well.

Attacks may include blatant and obvious frontal assaults, such as Denial of Service (DoS) attacks. They may take the form of more surreptitious penetration attacks; even to the point of establishing a fifth column within the camp, such as insidious Advanced Persistent Threats (APTs). As in any well-run military, effective security mandates that awareness and vigilance must be constant. These must be ingrained into every successful cybersecurity culture. Indeed, much can be directly transferred from the world of military strategy to that of cybersecurity strategy. In both worlds, it is eminently obvious that without the ability to detect the enemy, there can be no effective defenses. “If you cannot detect, you cannot protect.”

However, the problem is not merely that of “detection” *per se*, not merely that of measuring certain operational parameters. It is one of being able to sense, as rapidly as possible, the presence of a hostile, elusive, destructive enemy, one who will use all its significant resources to avoid detection and to destroy all defensive detection capabilities. Consequently, there must be two essential, equally important, aspects to every malware detection system:

- An ability to discern and recognize all malware penetration attempts, even those never before seen.
- An ability to withstand the disabling, destructive attacks that will inevitably be launched against a detection system.

Crytica’s Principles of Malware Detection mandate that successful detection requires both of these functions operate effectively and efficiently.

2. MALWARE DETECTION

The Past is Not a reliable Predictor of the Future

Within the cybersecurity community, there exists a critical misconception. It is that malware must be recognized and previously identified in order to be detected. This misconception has engendered a generation of malware detection systems that are rearward looking. They attempt to find evidence of malware that has already been seen and identified. These detection systems seek computer code that is either an exact match or is similar to previously known malware code. Or they may seek attack vectors that exactly match or are similar to the attack vectors of previously known malware. In this, they easily fall prey to totally new and polymorphic (shape/action shifting) malware.

This inability to detect the new and the dynamically changing has, for a long time, been regarded as the necessary but unfortunate Achilles heel of malware detection systems. In an attempt to mitigate the shortcomings of these rearward looking detection systems Artificial Intelligence (AI) technology has been employed extensively to enhance those systems. AI is used to look for "patterns", code and/or behavior similar to previously identified malware. It is also used to anticipate trends and designs in new malware. Predictably, AI has met with only very limited successes and, it has brought with it a whole raft of new concomitant problems, e.g., a plague of false positives.

A Paradigm Shift

If one thinks about malware as being only overtly malicious computer code, it is easy to become locked into the mindset that malware detection consists in seeking only malicious code. Under this mindset one knows that code is malicious either because it has been seen before or it is attempting to engage in malicious behavior. However, there is another way to define and perceive malware. If malware is regarded as being the unauthorized addition, modification, or deletion of a computer system's instruction set(s), that is, any change to a system's instruction sets by those who do not have the authority to control the code base of a system, the paradigm shifts.

Armed with this new perspective, it is possible to rapidly flag any new malware and all the variations of polymorphic malware; and it is even possible to help IT managers lock down their environments with controlled and well-managed software updates, policies, and procedures.

This new paradigm, supports the detection of malware at the moment of "injection", that is, detection of malware at the moment when the malware first infects the system. There is no longer a need to wait for and experience malicious behavior prior to detection. It becomes possible to detect sophisticated malware that can infect a system with a timebomb attack; those attacks that, when they are injected, appear to be totally innocuous. "Malicious" behavior-based detection systems cannot detect such attacks until they launch. Infection-based detection systems can detect such attacks immediately; usually in plenty of time to thwart them.

The Principles

Thus, the Crytica principles of malware detection include:

- Any unauthorized change to a system's instruction set is malware. It matters not what the change(s) affect(s).
- Malware must be detected at the moment of infection, rather than subsequently.
- A well-organized and well-managed IT environment is essential for a safe cybersecurity environment.

3. ATTACK RESILIENCE

Impregnable Walls – "Something there is that doesn't love a wall"[1]

History is littered with innumerable failed attempts to build impregnable walls. These "walls of folly" are a testament to humanity's love of "easy solutions" and its equally common predilection to ignore inconvenient realities. Hubris and denial have also played a prominent part in this long litany of ineffective walls. So too is it with the majority of cybersecurity systems that currently dominate the marketplace. Most cybersecurity systems are "walls", erected to detect and remediate malware infections, but they are not designed to withstand serious attacks against themselves.

As such, they are easy targets and constitute highly vulnerable candidates for "single-point-catastrophic failure." Indeed, many of them use code that is designed to, and must, run with administrator ("root") permissions. As such, when they are compromised, the attackers gain complete control of the entire "protected" system. The recent growing spate of Preemptive Malware (malware that disables the security defenses as part of its attack on a system) bears witness to the vulnerability of so many of the popular cybersecurity systems.

The Principles of Resiliency

Essential to the secure functioning of any system are its abilities to withstand the inevitable attacks that will be directed against it. If cogent Principles of Resiliency are adhered to, attacks are to be expected. Attacks may even disable/destroy various components of a system, but the system itself will remain functional and operational. The following are a few of those principles:

Assume that EVERY Component Shall be the Target of Attacks

In a hostile environment, every component can be, and will be, the target of attacks. Those attacks can originate from outside the defensive perimeter or from inside. No element is safe. No element can ever be regarded as being free from attack. Consequently, no element can ever be unquestionably trusted.

Assume that the Enemy is Intelligent, Powerful, Resourceful, and Experienced

No single factor has contributed to more disastrous defeats than the underestimation of the enemy. Every system must be designed with the assumption that the enemy can penetrate all of the defenses. If, for example, the defenses use AI to protect the system, it must be assumed that the attackers have better AI.

Create No Possibility for Single-Point-Catastrophic Failure

There can be no element in the entire system such that, if it fails or is compromised, the entire, or a significant portion of a, system fails.

Support Every Component by at least One Other Redundant Component

All components must be redundant, so that the failure of one does not constitute a failure of the entire system. Optimally, redundant components should not be co-located.

[1] Robert Frost, *Mending Wall*, 1914.

Distributing the targets of attacks minimizes the possibility that all the targets will be hit simultaneously.

Distribute the "Intelligence"

Every component should perform only one specific, well-defined, function. Thus, if a component is disabled or destroyed, only one, well-defined function is lost. Recovery in such an environment becomes a much easier task, and the number of targets an attacker must hit are multiplied.

Create No Unguarded Components

Every component/element of a system must be monitored to ensure that it has not been attacked, compromised, or destroyed.

4. HOW CRYTICA APPLIES ITS OWN PRINCIPLES TO DETECT MALWARE

Crytica Security applies its own principles and is able to:

- Detect malware that others cannot.
- Be resilient to attacks in ways others cannot.

Malware Detection – Detecting the Previously Undetectable

Crytica's detection platform is designed upon the principle that: malware is any unauthorized change to a system's instruction set. Therefore, it is designed to flag any and all unauthorized changes to a system's instruction set. It detects these changes at the moment that the change occurs in the system. It does not wait to see what the effects of the change are. It detects them at the time of infection, "Infection Detection™", and not at the time of malware launch.

Since Crytica does not rely upon any previously recorded malware signatures, nor upon any previously recorded behavioral patterns, Crytica can detect totally new strains of malware, and is not fooled by "shapeshifting" polymorphic malware.

Since Crytica detects malware at the time of infection, typically before the instruction set changes begin to perform their malicious behavior, Crytica detects preemptive malware prior to that malware being able to disable any of Crytica's detection systems.

By incorporating this paradigm shift into its detection engine, Crytica is able to detect malware attacks that were previously undetectable.

System Resiliency – Attack Absorption

Crytica's detection platform is designed upon the principles resiliency to attack. It is not designed to be an impregnable wall that the enemy will not be able to penetrate. Rather, it is designed to absorb all the attacks against, even at the cost of many compromised and destroyed components.

The Crytica Components – Distributed Intelligence & Modular Functionality

The primary Crytica components include:

- The Crytica Probe – One, or more, Crytica Probes reside in every Crytica protected device. Each Crytica Probe is assigned to one, and only one, Crytica Detector (see below). Its function is to continuously scan the protected device and report its findings back to the Probe's assigned Crytica Detector. A Crytica Probe is tiny. On Linux systems it is under 70 KB. A Probe is fast. Typically, on a Linux system it can scan a system with a million files in less than 20 seconds. A Probe is efficient. It consumes minimal resources, so that it can run continuously in the background without negatively impacting a host device's throughput and performance. A Probe is disposable. If it compromised or destroyed, it can be dynamically replaced. A Probe is an "Application Layer" process, so that it is not very Operating System version dependent, and it does not need to operate with "Administrator" permissions. Even if compromised, it can do very little damage.
- The Crytica Detector – A Crytica Detector interprets the information sent it by its assigned Probes. It can be either a virtual or a physical appliance. Each Crytica Detector has multiple protected devices assigned to it. Within each of the protected devices assigned to a Detector, the Detector manages a single Crytica Probe. The Detector requests "scans" from each of its subordinate Probes, that is, it requests raw information about each protected detected device. It then analyzes scans sent by the Probes in order to determine if there have been any unauthorized changes to the instruction sets on a device. The changes it looks for include:

  - Changes (modification) to an existing instruction set.
  - Additions to the number of instruction sets.
  - Deletion of an existing instruction set.
  - Changes to the permissions, owners, certain metadata, *et cetera* to an instruction set.

  The Detector issues an alert on any anomalies it discerns. These may include:

  - Unauthorized changes to the instruction sets on a protected device.
  - Disruption in the communication with the Probes. Note: all Probe-Detector communications are encrypted, digitally signed, and monitored via a proprietary heartbeat.
  - Disruption in the functioning of a Probe.

  In the event that a Detector determines that a Probe is not functioning as it should, the Detector is capable of "killing" that Probe and launching another one in its place.

Redundancy

In accordance with the Crytica principles of redundancy:

- Each protected device can have two or more Probes assigned to it, each reporting to a different Detector. The Probes are small enough and efficient enough that, in most environments, having two or more Probes operating will not negatively impact host system performance.

- Each Crytica platform device, e.g., each Crytica Detector, also has Probes operating within it, Probes that report to other Crytica Detectors. This creates a mutually monitoring mesh of components. An attack on any one of these will be detected by its monitoring partners.

Resiliency

The Crytica design assumes that all of its components will be attacked, and some may even be compromised and destroyed. However, due to the platform's redundancy, its ability to dynamically detect and replace compromised/destroyed components, its "Application Layer" Probes, and its distributed intelligence, the overall Crytica system can continue to function in the face of attacks and compromise.

Crytica's Detection Platform and the Crytica Design Principles

Crytica Security, Inc. has created a unique malware detection platform that has been designed and implemented based upon the above principles. For more information on Crytica's easily understood platform and its ability to detect malware at the time of injection, prior to launch, to detect polymorphic, preemptive, AI-Enhanced, and AI-Generated malware, and to be resilient to attacks launched against it, contact Crytica Security, Inc.:

Website: CryticaSecurity.com
Email: info@cryticasecurity.com