

CryptoGuard: An asymmetric approach to the ransomware battle

[S news.sophos.com/en-us/2023/12/20/cryptoguard-an-asymmetric-approach-to-the-ransomware-battle/](https://news.sophos.com/en-us/2023/12/20/cryptoguard-an-asymmetric-approach-to-the-ransomware-battle/)

December 20, 2023



Ransomware is one of the most significant threats facing organizations today. Battling it is no easy task, particularly given that threat actors are continually refining their techniques and approaches. Recent shifts, for example, include tweaks to ransomware-as-a-service (RaaS) models; the adoption of new programming languages; evolutions in targeting and deployment; and increasingly launching attacks after business hours and at weekends to hinder detection and incident response efforts.

One of the more substantial developments is an increase in remote ransomware: leveraging an organization's domain architecture to encrypt data on managed domain-joined machines. All the malicious activity – ingress, payload execution, and encryption – occurs on an unmanaged machine, therefore bypassing modern security stacks, with the only indication of compromise being the transmission of documents to and from other machines. Our telemetry indicates that there has been a 62% year-on-year increase in intentional remote encryption attacks since 2022. And Microsoft's 2023 Digital Defense Report states that around 60% of

human-operated ransomware attacks involve remote encryption, with 80% of all compromises originating from unmanaged devices, indicating a lack of active asset management. Ransomware families known to support remote encryption include Akira, ALPHV/BlackCat, BlackMatter, LockBit, and Royal, and it's a technique that's been around for some time – as far back as 2013, CryptoLocker was targeting network shares.

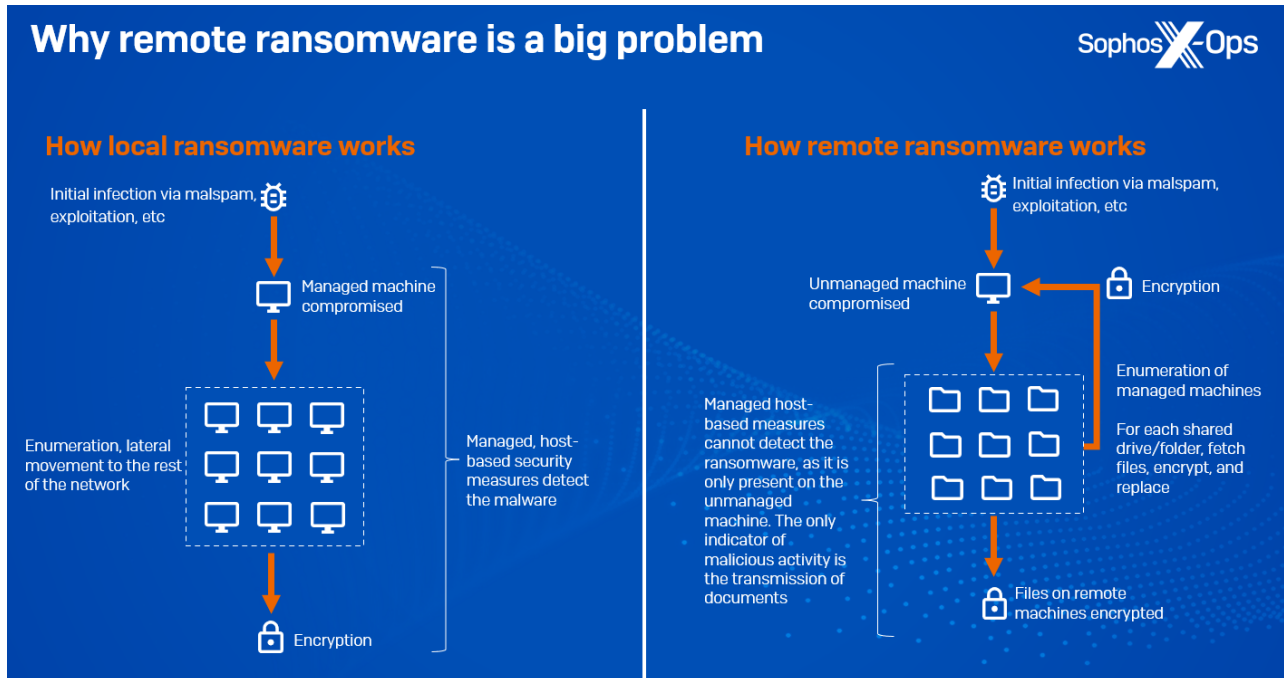


Figure 1: A simplified explanation of how remote ransomware works

Unsurprisingly, the rise and continuing development of ransomware has led to a plethora of research aimed at detecting and preventing it – with academics, security researchers, and vendors all proposing various solutions. Ransomware, as a form of malware, presents unique practical and intellectual challenges, and the range of solutions reflects this. Many such solutions target one or more of ransomware's distinct behavioral traits: enumerating filesystems, accessing and encrypting files, and generating ransom notes. Others are more generic, applying common anti-malware techniques to ransomware.

In this, the second issue of our new technical thought leadership series (the first, on memory scanning, is available [here](#)), we'll provide a brief overview of some of these techniques and their advantages and disadvantages, before taking an in-depth look at our own contribution to the field: CryptoGuard.

Before we start, one thing to note: a ransomware attack has multiple stages, and the majority of these will occur before the solutions we discuss in this article come into play. A well-defended enterprise will have multiple layers of protection which should stop attacks at various points, meaning that in many cases specific anti-ransomware solutions shouldn't be

required. But when all else fails, and a determined adversary reaches the encryption stage, we need a technology to prevent irreparable damage. Other phases of an attack – initial infection, persistence, lateral movement, and so on – are reversible, but encryption is not.

Anti-ransomware methods

Static solutions

Static techniques (i.e., those which can be conducted passively, without requiring execution of the malware) for ransomware detection are not markedly different from those used to detect any other kind of malware. Solutions in this vein include signature-matching, comparing strings; comparing file operations; examining behavioral traits; deep learning techniques; and examining PE headers.

While static methods have the advantage of being relatively rapid and low-cost, determined attackers can also evade them by modifying code until signature detections are broken. They are also less effective against new variants, packers, obfuscators, and in-memory threats, as well as remote ransomware.

Dynamic solutions

Dynamic solutions, on the other hand, tend to be more computationally expensive, but offer greater coverage. Dynamic anti-ransomware solutions in this vein include the following:

Filesystem interactions

Some security solutions will monitor for changes to file extensions, high-frequency read/write and renaming operations, or new files which have extensions associated with ransomware variants. On the other hand, some solutions leverage other interactions; the open-source project [Raccine](#), for example, is based on the premise that many ransomware variants delete shadow copies using **vssadmin**. Raccine works by intercepting requests to **vssadmin** and killing the process responsible.

Since ransomware targets files, it seems logical that numerous approaches should focus on filesystem interactions. However, many of them are reliant on analysis within a sandboxed environment; are predicated on anomalous patterns which threat actors may try to avoid generating; or can be resource-intensive due to the amount of monitoring involved (although it is possible to dynamically adapt the degree of monitoring) Some filesystem-based techniques may also not be effective when it comes to remote ransomware.

Folder shielding

While solutions like Controlled Folder Access (CAF) in Windows Defender limit access to folders to specific applications, such an approach is primarily geared towards individual users. CAF helps protect against ransomware by restricting unauthorized access to designated folders, allowing only trusted applications to modify files within them. However, for business networks, this method may be less practical due to the ongoing need for meticulous management of folders and applications. Additionally, it does not address the risk of attacks seizing control of trusted apps, a prevalent tactic in ransomware attacks

API calls

Some security solutions will assess API calls invoked by a process, either by flagging suspicious and seldom-seen calls or by determining potentially malicious call sequences.

Most ransomware employs API calls, although some variants use evasive measures to disguise these (particularly for API calls which are known to be suspicious, such as `CreateRemoteThread` or `VirtualAllocEx`, commonly used in process injection; or API calls related to encryption). Monitoring API calls at the kernel level certainly seems to be a worthwhile approach, but such monitoring is resource-intensive, can generate false positives, and is challenging to implement at scale. Additionally, when it comes to remote ransomware, the process itself may not be on the host being attacked, which can frustrate this approach.

Honeyfiles

Many security products employ ‘honeyfiles’, ‘decoy files’, ‘bait files’, or ‘canary files’ as an anti-ransomware solution – inconspicuous files which are placed in a directory and which legitimate users are asked not to touch. A separate monitoring system, either at the user-level or the kernel-level, is triggered if those files are accessed or changed by any process, at which point an alert is generated.

Honeyfiles are lightweight, low-effort, and can provide an early warning that an attack may be in progress. However, they do come with some caveats. Defenders must ensure that any alert is received and acted upon quickly enough, as by design an attack will already be in progress when a honeyfile is triggered. They also have to be strategically placed – deep enough within filesystems to ensure that normal, legitimate users and processes won’t accidentally trip them, but not so deep that important documents are encrypted before they’re accessed.

Fingerprinting

A less common technique is to ‘fingerprint’ certain malicious patterns – in network (C2) traffic, CPU consumption, or CPU signals.

With regards to network traffic, it’s worth noting that in modern human-led ransomware attacks, threat actors tailor and compile the ransomware binary uniquely for each victim, a strategic move intended to impede detection and complicate the decryption process. This

custom-built ransomware typically contains a victim-specific ransom note and is deployed in a 'fire-and-forget' manner, omitting the need for direct communication back to the threat actor, as the encryption process is self-contained within the malware, leveraging a victim-specific embedded public key.

An emerging technology from Intel called TDT (Threat Detection Technology) offers the ability to detect ransomware at the hardware level. [A review by SE Labs](#) demonstrates a remarkable effectiveness against a diverse array of encryption schemes. However, this is confined to specific Intel CPUs, excluding ARM and AMD architectures. This limitation stems from TDT's reliance on a machine learning model trained on CPU performance signals from specific ransomware families' encryption profiles. The model, trained by Intel, is dependent on vendor support and does not work with remote encryption. A disadvantage of this technology is that some ransomware strains, such as LockBit and Akira, are deliberately configured to encrypt only a portion of each file. This accelerates the impact of the attack, affecting more files in less time. It also means that detection by Intel TDT occurs after a significant number of files have already been compromised.

```
Process Trace
1 C:\Users\administrator.\Desktop\lock.\lock.\win_locker.exe [9612]
  win_locker.exe -remote -n=3 -p=\\172.\C$
2 C:\Windows\System32\cmd.exe [1316] *
  C:\WINDOWS\system32\cmd.exe /c ""C:\Users\administrator.\Desktop\lock.\lock_sr_all.bat" ""
3 C:\Windows\explorer.exe [2508] *
  "C:\WINDOWS\Explorer.EXE" /NoUACCheck
4 C:\Windows\System32\svchost.exe [1804] *
  C:\WINDOWS\system32\svchost.exe -k netsvcs -p -s Schedule
5 C:\Windows\System32\services.exe [712] *
6 C:\Windows\System32\wininit.exe [600] *
  wininit.exe
```




Figure 2: Akira ransomware, specifically attacking only remote data, and encrypting only 3% of each file

Automated telemetry-driven containment

Most modern endpoint protection solutions transmit data to the cloud for incident response and alert analysis. However, automatically piecing together the details of an active human-led ransomware attack from alert telemetry can take anywhere from a few minutes to several hours. This latency depends on the configured telemetry reporting frequency, the presence of other alert signals, and the cloud's processing capacity to assemble and correlate specific events from multiple protected machines.

Following detection, an automated response can involve deploying a containment policy to managed devices, to isolate a specific user account suspected of compromise by the attacker. While this action aims to prevent an imminent or ongoing (remote) ransomware encryption attack originating from the identified account, it is important to note that the distribution of this policy also requires time (up to hours). Moreover, in scenarios where the attacker starts encryption without triggering prior alerts on managed machines (as noted

above, 80% of attacks involve unmanaged machines) or opts to begin the encryption process from an alternate user account, the conditions do not always favour an effective cloud-driven dynamic containment strategy. But it can be helpful in some instances.

Rollback

In general, dynamic anti-ransomware solutions commonly require some level of encryption or data manipulation to have taken place before detecting the attack. Consequently, a certain number of files will likely become encrypted, necessitating a backup and restore function to recover affected files.

To revert unencrypted file versions, some endpoint protection products leverage Volume Shadow Copies, a Windows feature that generates data snapshots at specific time points. These 'shadow copies' capture file or volume states, even while they're in use. Nevertheless, this method has its limitations: attackers commonly delete the shadow copies; they do not protect files on network mapped drives; and effective rollback relies on detecting and addressing the ransomware incident before the subsequent scheduled snapshot (which typically occurs every four hours). And, as noted previously, most attacks happen after office hours, which can complicate recovery attempts using this method.

Summary

Generally, many of these approaches focus on looking for 'badness': characterizing and identifying behavioral traits which are indicative of ransomware activity. While this seems like a rational decision, it does have a crucial weakness, in that threat actors have an incentive to disguise or obfuscate those traits and therefore evade detection. CryptoGuard, on the other hand, takes a different approach.

CryptoGuard

CryptoGuard – formerly known as HitmanPro.Alert, and part of Intercept X since 2016 – was first developed in 2013, and is intended to be a last layer of defence against both local and remote ransomware, when determined threat actors have evaded all other protections and are in a position to begin encryption. Its notable successes include blocking WannaCry, LockBit, and REvil ransomware. While we keep a very watchful eye on developments in the ransomware space, CryptoGuard hasn't changed substantially over the years, primarily because it hasn't needed to.

An asymmetric approach

Unlike the majority of the approaches described above, CryptoGuard doesn't look for attackers, ransomware executables, or malicious behavioral patterns at all. Other security solutions, including Sophos products, do these things, of course – it's a fundamental part of a

layered defence, which ideally prevents attackers from getting to the encryption stage – but CryptoGuard itself employs a more asymmetric approach, for when those layers have been circumvented.

Rather than looking for ‘badness,’ CryptoGuard focuses on the contents of files, by analyzing their patterns with a mathematical algorithm. Whenever a process opens a file for reading and writing, CryptoGuard’s minifilter driver – which operates within the Windows operating system kernel – continuously generates histograms of the read and written data. These histograms serve to understand the overall pattern and characteristics of the data. They undergo evaluation to determine their entropy and statistically analyze whether the read and written data is unencrypted, compressed, or encrypted. The built-in evaluators employ mathematical models to classify data. Since the analysis uses the same memory buffers provided by the operating system for the requesting process, it is very efficient as it does not cause additional disk input/output (I/O).

Sophos CryptoGuard

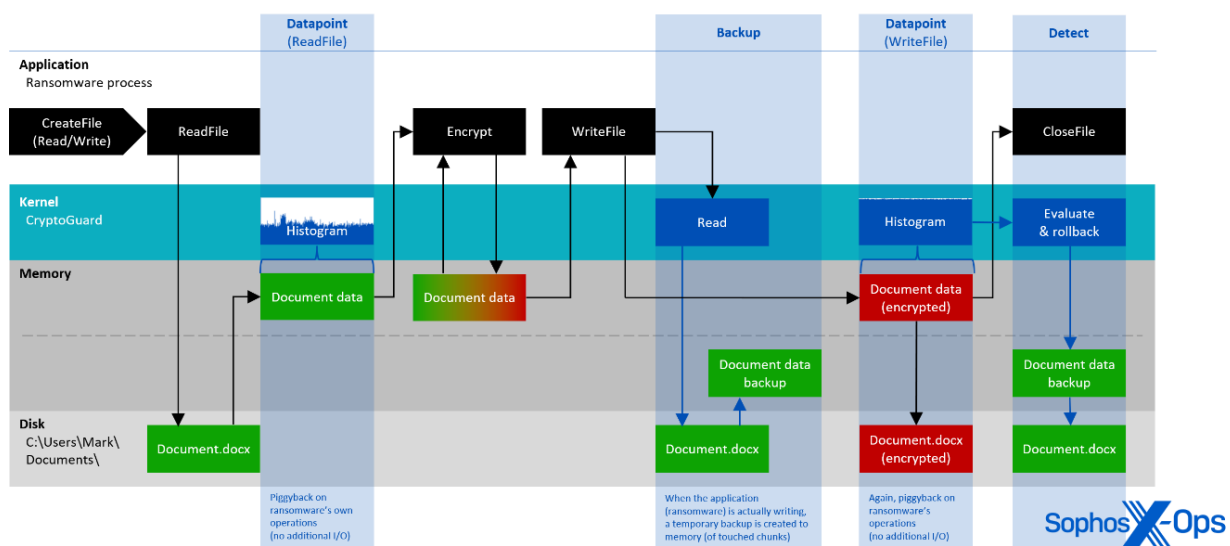


Figure 3: An overview of CryptoGuard’s operations

This capability provides asymmetric protection, even in scenarios where an unprotected remote machine on the network is attacking shared documents on a Sophos-protected file server, for example. As noted above, most human-led ransomware attacks aim to also encrypt shared data on remote machines. In such cases, the ransomware itself is not executed on the protected remote machine (either because it wasn’t deployed there by the attacker or was blocked by endpoint protection). As a result, the ransomware binary itself or the attacker-controlled process (that performs the encryption) cannot be observed from the machine that holds the targeted data.

So, because there is no malicious code to be detected on the attacked machine, technologies like antivirus, machine learning, indicators of breach, etc.—all focused on identifying adversaries and their malicious code—are completely sidelined and not in play (even if it is a well-known years-old sample responsible for the encryption). However, CryptoGuard can recognize when a remote machine replaces documents in the shared folder with encrypted versions, and automatically takes action by blocking the IP address of the remote machine and reversing the changes it made. It creates temporary backups of any modified files, so that the changes can be rolled back if mass encryption is detected, and can also detect the deployment of ransom notes within the folders where the ransomware has encrypted files. Consequently, it sometimes identifies instances of data exfiltration, even though it was not explicitly designed for that purpose.

Zero-trust

Adversaries will sometimes abuse an existing process, or package a normally benign process that loads a malicious DLL ([known as DLL side-loading](#)), in order to perform encryption. The encryption activity is performed under the identity of the benign process, now running attacker-code, and encrypting documents.

A real-world example of this is [the Kaseya VSA incident](#), where the REvil threat actor embedded a malicious DLL to be side-loaded in an outdated but vulnerable Windows Defender executable. The threat actor purposely chose Defender, because protections typically trust code signed by Microsoft. Additionally, a DLL cannot be examined as thoroughly as an executable in a sandbox environment, meaning it may be ‘approved’ sooner.

On that occasion, Sophos detected both the REvil payload itself, as well as an REvil-specific code certificate. And while Kayesa’s protection exclusions allowed the REvil dropper to be installed on machines, CryptoGuard detected the ransomware, because it’s not constrained by such exclusions and blocks file encryption anywhere on protected drives.

A walkthrough

Remote Ransomware vs Sophos CryptoGuard

Mark Loman
VP, Software Development



[Watch Video At:](#)

<https://youtu.be/JPWU-yCaShg>

Conclusion

There is no panacea when it comes to battling ransomware. An effective defence should include a myriad of layers, from vulnerability remediation and configuration reviews to user education and security solutions. But, regardless of which layers organizations employ, and how many, an important aspect to consider is the robustness and effectiveness of the last layer, when all other measures have failed and threat actors are in a position to execute their ransomware. At that point, the solutions we've covered here come into their own.

These solutions are diverse, covering numerous different behavioral traits and activity. Many vary widely in terms of their scalability, versatility, and cost-benefit ratios, and have distinct strengths and weaknesses. A key commonality is that most solutions focus on 'detecting badness' in some way – whether through API call analysis, honeyfiles, or some sort of fingerprinting. That's not necessarily a disadvantage, and a layered and diverse defence stack is a solid approach. But, as we've shown, the CryptoGuard approach within Intercept X is slightly different, and more asymmetric: focusing on file contents rather than the behaviors of ransomware or its operators.

Ransomware continues to evolve, and more and more solutions and techniques are likely to appear in response. As we've been doing for the last ten years, we'll continue to track changes in both ransomware and the solutions designed to detect and prevent it.