Analysis of the Malware in Piracy Networks: Static Analysis of Executable Files Distributed via the Torrent Network in 2026

19th of May 2026

UPDATED: 26th of May 2026

Contents

Abstract

The distribution of pirated software and media via peer-to-peer (P2P) torrent networks remains a primary vector for malware distribution. Threat actors frequently use high demand for “cracked” executables to bypass security measures of users seeking free software. This study outlines a methodology for safe isolation, extraction and analysis of suspicious payloads downloaded via torrents to better understand risks of executing untrusted files.

Outline

The main goal of our study is to highlight how easy it is to obtain pirated software applications but also how it might lead to compromise of the user's device, their data or possibly their funds of their cryptocurrency wallets or bank accounts.

For this study, we selected two well-known file-sharing platforms (BitTorrent trackers), The Pirate Bay and LimeTorrents and searched for the top 100 most popular downloads on each.

Our testing process involved first setting up a secure and quarantined testing space. This allowed us to safely open and inspect the downloaded files without any risk of malware spreading to our actual computer or our home network. It also ensured that every file was tested under identical, clean conditions.

Analyzing each sample involves a few fairly simple steps and all of them can be reproduced by regular users to greatly improve their chances of not getting infected if they still decide to obtain software through this way.

Our specific setup used a standard version of Windows running inside a virtual machine (a simulated computer inside a real computer) alongside a custom automation program to handle most of the repetitive testing steps automatically.

The first step was to obtain the software we wanted to analyze, using the qBittorrent[1] app to download the files. Once downloaded, the files were automatically scanned by our custom automation program. We focused our analysis primarily on runnable program files (executables), or files that had been modified by the torrent creators to potentially hide harmful malware.

For future reference, we used the Wayback Machine, a service provided by the non-profit organization Internet Archive. This tool captured a digital snapshot of the exact download pages at the time of our analysis. These snapshots allow us to revisit and reference the exact state of the website in our study, even if the original pages are later changed or taken offline.

[1] qBittorrent is one of the most popular and free applications used to download torrents.

After a successful snapshot we upload certain files matching our filters for analysis to VirusTotal. This service provides a standardized way of analyzing our samples with over 70 antivirus scanners and the community can also vote on each sample, marking it either harmless or malicious.

Our analysis process also included other measurements, such as extracting the text content from the samples. This text often contains IP addresses or domain names for servers used by the attackers to secretly download secondary malicious files.

All of our findings were stored, categorized and later used for further analysis. These results are presented in a clear, easy to read format in the **Analyzed samples** sections of this document on page **22**.

The Persistence of BitTorrent in Software Piracy

Despite the rise of legitimate subscription models the BitTorrent protocol remains a primary vector for online software piracy. The main two factors are the protocol's architecture and the economics.

Reasons for continued use

Traditional client server distribution requires a massive bandwidth and centralized hosting, which creates one central weak spot. Supporting the bandwidth is costly and prone to legal takedowns. BitTorrent utilizes P2P swarm where users download and upload file chunks simultaneously effectively taking the costs down to zero. The decentralized nature makes it practically impossible for authorities to completely eradicate a file once it is well seeded. Because downloaders pull data from multiple peers simultaneously, highly seeded software (e.g., cracked operating systems or popular games) can be downloaded at speeds that rival or exceed premium direct-download services.

The cons

Because there is no centralized quality control or authentication, malicious actors can easily inject trojans, ransomware, or cryptominers into heavily demanded software executables. Older or niche software inevitably suffers from "link rot" in a P2P network. If no peers are seeding the complete file, the download cannot be completed.

Torrent Trackers

In our study we have elected to analyze torrents from two different trackers.

The Pirate Bay (TPB)

- **History:** Founded in 2003 by the Swedish pro-piracy think tank *Piratbyrån*, TPB is the most infamous file-sharing site in internet history. It survived a massive police raid in 2006, the imprisonment of its founders, and continuous ISP blocks. It achieves this by constantly rotating domains and operating behind proxies.
- **Why it is popular:** TPB is a cultural icon of digital defiance. It acts as a massive, largely uncurated index. TPB is the perfect baseline because its sheer volume and lack of strict moderation make it highly susceptible to malicious uploads, representing the raw edge of P2P file sharing.

Search Torrents | Browse Torrents | Recent Torrents | Top 100

top100:301 | Pirate Search | All

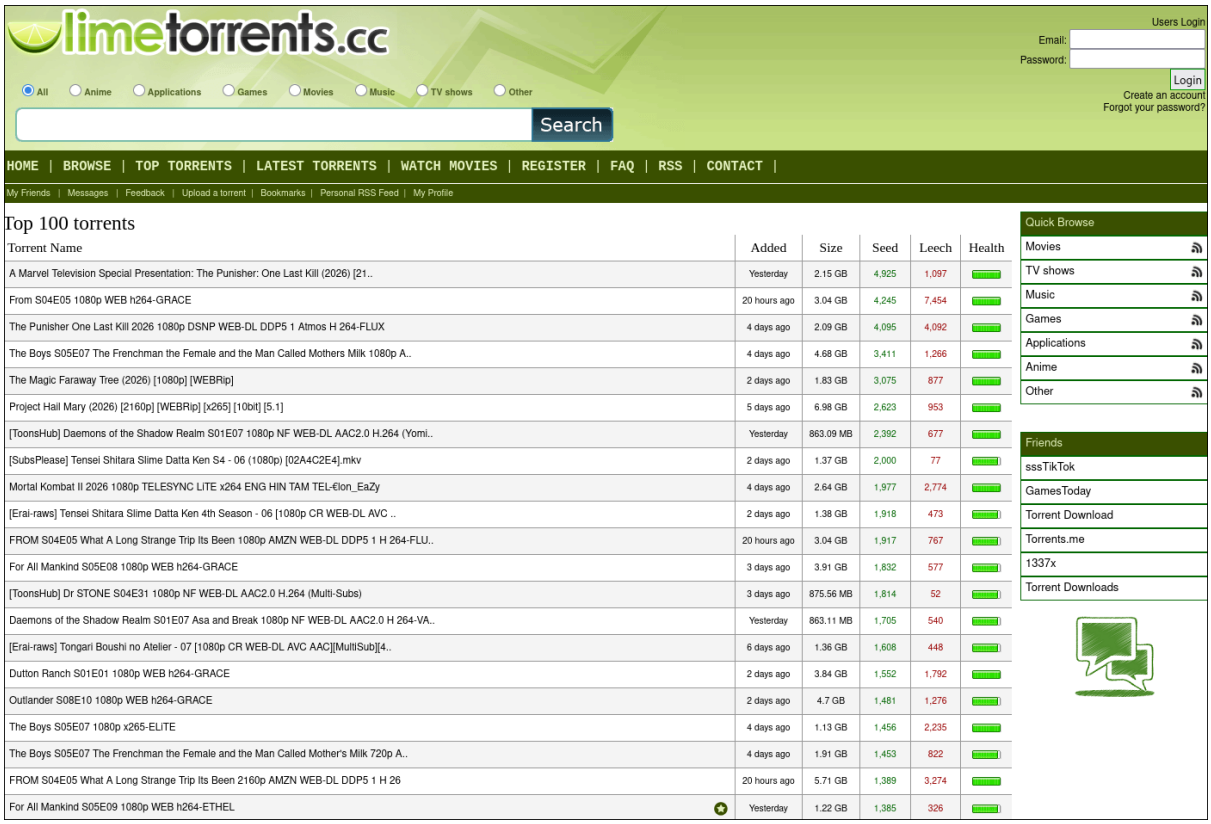
Top 100: Applications > Windows

Filter names.. Quick Filters: 720p 1080p 2160p x264 h264 x265 h265 WEBRip BluRay HDR HEVC

Category	Name	Uploaded		Size	SE	LE	ULed by
Applications > Windows	Avira System Speedup Pro 5.3.0.9960 + Crack [TechTools]	2019-03-12		7.98 MiB	4754	3229	ThumperTM
Applications > Windows	Windows 11 Pro 24H2 Build 26100.3194 With Office 2021 Pro Plus (Non-TPM) En-US	2025-03-09		7.18 GiB	1884	268	Prom3th3uS
Applications > Windows	Windows 11 Pro 25H2 Build 26200.7171 With Office 2024 Pro Plus [Non-TPM] (x64)	2025-11-29		8.01 GiB	1709	277	Prom3th3uS
Applications > Windows	Windows 10 Pro 22H2 Build 19045.7184 With Office 2024 Pro Plus (x64) EN-US	2026-04-19		9.89 GiB	1630	244	Prom3th3uS
Applications > Windows	Windows 10 Pro 22H2 Build 19044.6466 (x64) EN-US + Activator [FTUApps]	2025-11-28		7.12 GiB	1584	246	Prom3th3uS
Applications > Windows	MICROSOFT Office PRO Plus 2016 v16.0.4266.1003 RTM + Activator	2015-09-27		2.21 GiB	1488	19	ThumperTM
Applications > Windows	Adobe Photoshop 2023 24.2.0.315 (x64) + Crack [TheWindowsForum]	2023-02-24		2.88 GiB	1454	31	ThumperTM
Applications > Windows	MICROSOFT Office PRO Plus 2024 + Activator [TheWindowsForum]	2024-10-09		5.15 GiB	840	106	ThumperTM
Applications > Windows	Windows 11 Pro English x64 BIT Activated	2023-06-09		4.37 GiB	725	5	abdenna
Applications > Windows	Adobe Photoshop 2020 v21.2.2.289 (x64) Pre-Cracked - Team Rjaa	2020-09-01		1.78 GiB	573	8	rjaa
Applications > Windows	Adobe Illustrator 2022 v26.1.0.185 (x64) [2022, MULTILANG] -Rjaa	2022-03-22		1.81 GiB	551	5	rjaa
Applications > Windows	Autodesk AutoCAD 2024 x64 - Rjaa	2023-04-19		2.82 GiB	539	7	rjaa
Applications > Windows	Adobe Acrobat XI Pro 11.0.20 FINAL + Crack [TechTools]	2017-04-10		744.2 MiB	524	17	ThumperTM
Applications > Windows	Adobe Premiere Pro 2022 v22.2.0.128 (x64) [2022, MULTILANG]	2022-04-10		1.86 GiB	441	6	rjaa
Applications > Windows	KMSpico 10.1.8 FINAL + Portable (Office and Windows 10 Activator	2015-11-01		8.6 MiB	306	19	ThumperTM
Applications > Windows	Adobe Lightroom Classic 2023 12.4.0 (x64) FULL [TheWindowsForum]	2023-06-13		1.42 GiB	314	4	ThumperTM
Applications > Windows	AUTODESK AUTOCAD 2023 (X64) PORTABLE	2022-04-10		1.78 GiB	302	5	rjaa
Applications > Windows	FL Studio Producer Edition 24.1.2.4394 x64 All Plugins Edition + Crack- [haxNode	2024-10-17		3.25 GiB	285	3	haxnode
Applications > Windows	Wondershare Filmora v14.2.9.11061 (x64) + Crack - [haxNode]	2025-01-08		807.71 MiB	160	1	haxnode
Applications > Windows	IBM SPSS Statistics 26.0 IF006 + Crack [TheWindowsForum]	2019-10-16		2.02 GiB	159	1	ThumperTM
Applications > Windows	Blackmagic Design DaVinci Resolve Studio v20.0.49 (x64) FINAL + Crack [AppDoze]	2025-05-29		8.14 GiB	157	2	appdoze
Applications > Windows	Adobe Photoshop 2023 v24.1.0.166 (x64) FULL [TheWindowsForum.com]	2022-12-14		2.73 GiB	155	4	ThumperTM
Applications > Windows	Adobe Photoshop 2026 v27.1.0 (x64) FINAL Pre-Activated [AppDoze]	2025-11-21		5.47 GiB	150	9	appdoze
Applications > Windows	Microsoft Windows 10 PRO FULL (x64-x86) Sep 2015 [TechTools]	2015-09-13		3.37 GiB	156	2	ThumperTM
Applications > Windows	Microsoft Windows 11 build 21996.1 x64 + Activator [TheWindowsF	2021-06-18		4.54 GiB	131	27	ThumperTM

LimeTorrents

- **History:** Operating for well over a decade, LimeTorrents has maintained a steady presence by frequently shifting its domain structure (using TLDs like .fun, .cc, and .pro) to evade legal pressure.
- **Why it is popular:** Unlike TPB, LimeTorrents attempts a degree of curation. It features a cleaner interface and often utilizes visual markers for "verified" uploaders. It serves as a vital fallback for users when TPB is down.



Common P2P Malware and User Impact

Torrents with pirated software, video games or cracked operating systems are prime vectors for malware delivery. The most common threats distributed via P2P networks primarily target user credentials and computational resources.

Information Stealers (InfoStealers)

Infostealers are currently the most dominant threat, they operate silently and scrape user passwords, website log in sessions or cryptocurrency wallets. These infostealers often execute once and upon completion delete themselves making it harder for users to become aware they have been compromised often finding out after their accounts have been breached or fund drained.

Remote Access Trojans (RATs)

These types of malware provide the attacker a backdoor to the user's device granting them complete administrative control over the infected device. The attackers might monitor the user's screen, webcams, microphones or use the device as a node in a larger botnet to launch large-scale Distributed Denial of Service (DDoS) attacks[2].

Ransomware

While more commonly targeted at enterprises this type of malware encrypts user's data and later presents the victim a digital ransom note demanding some form of payment, typically cryptocurrency, in exchange for the decryption keys.

[2] DDoS attacks flood the victim with traffic originating from many different sources. To mitigate these kinds of attacks a more sophisticated are required as banning a single IP address is insufficient.

Cryptominers

Attackers hijack the user's CPU and GPU to mine some form of cryptocurrency. The immediate impact is degraded performance, device overheating and increased electricity consumption.

The Illusion of Security

A common misconception among users of P2P networks is that having an Antivirus (AV) installed guarantees safety when pirating software. In reality relying solely on AV in the context of software piracy is fundamentally flawed.

User as the Vulnerability

The most significant vulnerability is the user. The instructions to install pirated software typically tell the user to turn off their AV or take out files out of a quarantine. This conditions the user to blindly trust any other software creating false positives thus rendering the AV useless actual embedded malware.

Evasion and Zero-Day Threats

- **Fileless Malware and LoLBin Abuse:** Advanced threats may not drop a traditional `.exe` file at all. Instead, they exploit "Living off the Land Binaries" (LoLBins). For example, a malicious script embedded in a pirated software installer might hijack legitimate system tools like Windows PowerShell or MSBuild to execute malicious commands directly in the memory. Because the tool executing the command is a trusted part of the operating system, traditional AV often fails to block the behavior.
- **Sandbox Evasion:** Sophisticated malware can detect if it is being scanned by an AV engine or running in a virtual machine. If it detects these environments, it will either stall its execution via a sleep timer or terminate itself to appear benign, only detonating when it confirms it is on a genuine user's machine.

Real-World Case Studies

The "CracksNow" Rogue Uploader Campaign

Several torrent sites have banned the account of the popular software uploader CracksNow. The software and cracks were repeatedly flagged as malicious and some came with the GandCrab ransomware. While malicious torrents are nothing new, it's rare for this to happen via a "trusted" uploader. (TorrentFreak)

CracksNow possessed trusted "VIP" status across major indexing platforms, including The Pirate Bay, 1337x, and TorrentGalaxy. Over several months, the account abused its reputation to mass-distribute the **GandCrab ransomware** and various information stealers bundled inside legitimate-looking software patches. This incident was a turning point for the P2P community, demonstrating that threat actors will patiently build account equity over years just to burn it on a massive payload distribution campaign.

Automated InfoStealer Swarms

Threat actors deploy large-scale automated networks to compromise P2P networks. Using bot accounts, attackers mass-upload thousands of highly sought-after digital assets. These files are laced with modern, fast-executing info-stealers like **LummaC2, RedLine, and NWHStealer**. The uploaders often employ search engine optimization (SEO) techniques within the trackers to ensure their malicious torrents appear at the top of search queries with faked seeder numbers.

How Torrent Trackers Combat Malware

Torrent trackers are not entirely lawless; their administrators employ several decentralized and centralized moderation techniques to preserve the integrity of their platforms.

Public Trackers

To counter anonymous malicious uploaders, public trackers such as TPB and LimeTorrents utilize a visual verification system. TPB uses “VIP” (Pink Skull) and “Trusted” (Green Skull) icons, while others use colored badges. These indicate that the uploader has a long history of uploading clean files.

Some public trackers display a user comment section which is a highly effective, community driven early warning system. When an infected file is uploaded, technical users who analyze it via VirusTotal or sandbox environments will quickly post warnings.

Administrators rely heavily on user report flags. Once a file is confirmed malicious, moderators act retroactively to delete the torrent and ban the uploader's IP and account hash.

Private Trackers

These kinds of trackers (invite only, specialized communities) offer a drastically more secure environment. Users must maintain a strict upload-to-download ratio to retain access. Because getting invited is highly exclusive, the "cost" of losing an account due to malicious behavior is too high for casual hackers.

On elite private trackers, newly uploaded software files do not go live instantly. They enter a curation queue where specialized staff review the contents, verify the digital signatures, and check for malicious artifacts before the torrent is broadcast to the swarm.

User-Side Risk Mitigation

While P2P software piracy carries inherent risks, advanced users adopt extensive defense strategies to significantly minimize the probability of infection.

- **Strict Reputational Auditing:** Users can mitigate an estimated 90% of public torrent threats simply by adhering to a strict rule: *never download executable assets from unverified or anonymous uploaders.*
- **File Extension and Volumetric Verification:** Malicious torrents often betray themselves through metadata discrepancies. Attackers frequently use double extensions (e.g., `Photoshop_Crack.pdf.exe` or `Setup.dmg.exe`) hoping the user has file extensions hidden in their OS settings. Furthermore, checking file size is a crucial defense. A compressed archive claiming to be a 60GB AAA video game that downloads as a 15MB `.exe` file is a definitive indicator of a malicious loader stub.
- **Isolated Virtualization (Sandboxing):** High-risk software acquisition requires strict environment isolation. Secure practices involve executing cracks and keygens inside a temporary virtual machine (using hypervisors like VMware or VirtualBox) or utilizing features like **Windows Sandbox**. If the file detonates an info-stealer or ransomware payload, the damage is completely contained within the volatile virtual machine and wiped upon closing, leaving the host operating system untouched.
- **Pre-Execution Static Analysis (The VirusTotal Framework):** Advanced users don't rely blindly on their desktop AV, instead they upload the contents to VirusTotal prior to launching them. By checking the results the user can preemptively isolate zero-day threats.

Documented Infection Rates

To establish the baseline threat level of P2P networks, this study draws on comprehensive empirical data from major academic and industrial tracking initiatives.

- **The 99% Weaponization Rate:** In a massive, foundational study analyzing tens of thousands of active torrent files, researchers developed *TorrentGuard* to track file intent. The study revealed that **35% of all analyzed torrents uploaded to public networks were entirely fake**. Crucially, **more than 99% of those fake files were directly linked to malware delivery** or aggressive phishing/scam operations. Furthermore, despite rapid moderator takedowns, the study noted that *every fourth file successfully downloaded* by an average user from these public portals consisted of malicious, fraudulent content (Kryczka et al.).
- **The 43% Software Poisoning Metric:** For users downloading executable binaries—such as cracked productivity suites or operating systems—the numbers become even more severe. Enterprise security firm Bitsight monitored P2P traffic traversing corporate and vendor networks. Their telemetry discovered that **43% of torrented application files** and **39% of torrented video games** contained active, malicious payloads capable of entirely compromising host networks and initiating botnet infections (bitsight).

These high percentages are not accidental or organic; they are the result of highly centralized "swarm poisoning" campaigns run by a remarkably small group of actors.

According to the *TorrentGuard* data, **just 20 distinct publishers (tracked by unique IP groupings) were responsible for injecting over 90% of the malicious and fake files** across the ecosystem.

Methodology

To safely examine analyzed torrent files, we established a strict, automated analysis pipeline operating from within an isolated environment.

Environment Setup

Analyzing untrusted files requires a clean, heavily controlled and monitored environment to prevent host and local network compromise. We utilized a hardened Windows Virtual Machine (VM) configured strictly for malware analysis.

- **Network Segmentation:** The VM was deployed on an isolated virtual switch to prevent local network scanning.
- **Hypervisor Artifact Mitigation:** Modern malware often uses anti-analysis techniques such as checking for artifacts like VMware tools, Virtualbox drivers or specific MAC addresses to avoid executing malicious payloads while under observation. We changed the VM's configuration files to hide these signatures.
- **Disabled Integrations:** Host to guest integrations, such as shared folders, drag-and-drop and shared clipboards were completely disabled to prevent guest to host escapes and as mentioned in previous point these features usually require easy to spot drivers.
- **Snapshotting:** After each analysis the VM was restored to a clean state using snapshots.

Automated Static Analysis Pipeline

We elected to strictly analyze samples using static analysis rather than dynamic analysis, this comes from the decision to not focus on what each sample does but more broadly understand the landscape. We developed a custom script to parse downloaded executables without executing them and relied on mathematically derived metrics and heuristic indicators of compromise.

The following pseudo-code represents our scripts, our implementation used TypeScript with the Bun runtime for ease of development and the option to bundle into a simple executable.

```
main(samplePath){
        var sha256 = calculateSha256(samplePath);
        var entropy = calculateEntropy(samplePath);
        var vtResult = scanWithVirusTotal(sha256);
        var strings = extractStrings(samplePath);
        writeToFile(sha256,entropy,vtResult,strings);
}
```

The script first calculates a SHA-256 of the provided sample which is later used for sending to VirusTotal and as a unique identifier for the results.
The next step is to calculate the entropy.The previously calculated SHA-256 is used to obtain results from VirusTotal, we pick out the most important data to us such as how many community votes the sample has received or how many virus engines have found the sample to be malicious.
All of the collected data is then stored to an output file and later used for the analysis. When storing the output data we also take a snapshot of the current state of the webpage we used to obtain the torrent file. For this we use the Wayback Machine which is designed to preserve websites, images, videos or any other form of data in an immutable way which can be later used as a reference in the case of the original being deleted.

VirusTotal vs. Alternative Solutions

In static analysis, the choice of scanning and aggregation tools heavily dictates the validity of the findings. VirusTotal (VT) was selected as the primary analytical engine over alternatives due to its breadth of consensus and standardization of static metadata.

How does it work

VirusTotal acts as an aggregator, pushing submitted files and URLs through more than 70 distinct antivirus engines and blocklisting services. It strips the executable to extract static metadata without executing the payload.

Alternatives

Malware sandboxes are exceptional tools for *dynamic*, interactive analysis. They execute the malware in a virtual machine to monitor network traffic, registry changes, and API calls in real-time. Our research focuses on *static* analysis proving VirusTotal to be a perfect fit for us.

Methodological Boundaries

While reverse engineering and disassembling malicious payloads provide the deepest possible understanding of a malware's inner workings, it was deliberately excluded from the scope of this study in favor of bulk static analysis.

Why Disassembly was Bypassed

- **Scalability:** Disassembling an executable (converting machine code back into assembly language using tools like IDA Pro or Ghidra) is an intensely manual, time consuming process. Given that the methodology involved downloading a large volume of torrents to establish statistical trends, manual reverse engineering would severely bottleneck the sample size.
- **Scope of the Study:** The primary objective was to observe the distribution metrics, detection rates, and surface-level indicators (entropy and plain-text

strings) of infected P2P software, rather than mapping the specific API calls or logic flow of individual payloads.

The Barrier of Obfuscation

Malware authors actively anticipate disassembly and employ sophisticated obfuscation techniques to make research and detection more complex. These techniques make static disassembly virtually impossible without first pivoting to dynamic execution.

- **Packers and Crypters:** Tools like UPX, Themida, or custom runtime packers compress and encrypt the original malicious code. The executable only decrypts itself in the system's memory at runtime. Disassembling the file on disk only reveals the decompression stub, not the actual malware and this is the reason why we also included entropy as one of our metrics.
- **Junk Code Insertion:** Injecting meaningless "dead code" to bloat the file size and confuse the reverse engineer or automated analysis engines.

Payload Classification and Ground Truth Establishment

Relying on a single, isolated antivirus engine introduces structural biases and gaps in signature coverage. We elected to use VirusTotal's aggregated framework for three primary reasons.

- **Diverse Detection Methodologies:** The integrated engines utilize distinct, complementary approaches—ranging from traditional static byte-sequence signatures to heuristic behavior analysis and cloud-based machine learning models. This multi-layered coverage ensures that variants utilizing novel packing methods are captured by engines specialized in generic unpackers or dynamic heuristics.
- **Continuous Synchronization:** Threat definitions are updated in near-real-time across the platform's infrastructure. This minimizes the "window of vulnerability" often found in localized setups, ensuring that newly discovered malware strains active in BitTorrent swarms are evaluated against the most current global definitions.
- **Academic and Industry Standard:** Utilizing multi-engine aggregation as a ground-truth labeler is an established methodology in peer-reviewed cybersecurity literature (Choo et al.). It eliminates subjective, manual classification biases and provides a highly reproducible testing framework.

Mitigation of False Positives

A primary concern when utilizing automated scanning platforms is the risk of false positives, where legitimate software,such as game cracks, cracked productivity tools, or administrative utilities common on BitTorrent networks, is mistakenly flagged as malicious.

We have employed enforcing a threshold strategy with multiple confidence layers. Later in the document we describe the specified levels and how we filtered out certain flags and what it meant for the final results. A sample meeting the highest confidence layer represents a consensus verdict among competing security vendors, giving us certainty that the analyzed samples contain genuinely malicious, non-benign artifacts.

Statistically, if each engine has an independent ~3% false-positive rate, the chance two engines **both** false-flag the same file is about 0.09% (resec). Requiring *n* engines to agree makes that probability drop exponentially (e.g. 0.027% for three engines). In contrast, detection (true-positive) rates improve: combining two independent AVs can reduce the malware escape rate from ~0.6% to 0.0036%. This means even one or two uncorrelated engines missing malware becomes extremely unlikely as more are added.

Modern AV engines rely on multiple detection technologies including:

- static signature matching,
- heuristic analysis,
- behavioral monitoring,
- machine learning classification,
- cloud reputation systems,
- sandbox execution

These systems are trained on enormous datasets of known malware samples and suspicious behavioral patterns. As a result, detections are usually based on observable technical indicators rather than arbitrary assumptions.

However, isolated detections are fundamentally different from broad multi-engine consensus. If only one or two antivirus vendors flag a file as a “HackTool” or “Patcher,” the result may indeed represent a benign crack triggering heuristic rules. In contrast, when dozens of independent engines classify the same sample as a trojan, downloader, stealer, or ransomware family, the probability of a false positive becomes significantly lower.

This distinction is especially important in platforms such as VirusTotal, where files are scanned by over 70 separate security vendors using independent detection models. A sample flagged by 30–50 engines, particularly when combined with suspicious behaviors such as:

- sandbox evasion,
- network beaconing,
- payload downloading,
- registry persistence,
- code obfuscation,
- or credential harvesting

should be treated as highly suspicious regardless of claims that it is “only a crack.”

In practical terms, antivirus engines are not infallible, but large-scale detection consensus remains one of the strongest publicly available indicators that a file is genuinely malicious rather than merely intrusive.

Why False Positives Occur

Modern endpoint detection tools have evolved past basic static byte-matching (SHA-256 signatures) (Huntress). They heavily rely on **heuristic and behavioral analysis** to catch zero-day threats.
Legitimate software often mirrors the exact behavioral patterns of malware. For instance, a benign software updater, an administrative remote-access tool, or a BitTorrent digital rights management (DRM) bypass ("crack") must inject code, modify registry keys, or establish low-level network hooks.
Many legitimate developers compress or pack their executables using tools like UPX to reduce file sizes or protect their intellectual property. However, because malware authors use those exact same packing tools to hide payloads, security engines frequently issue blanket generic verdicts (e.g., `Generic.Packer.UPX`) purely based on the file wrapper, ignoring whether the payload inside is benign.

Analytical Metrics and Indicators

Our script extracted several core metrics from each analyzed sample to determine its threat level.

Cryptographic Hashing (SHA-256)

What it is: SHA-256 (Secure Hash Algorithm 256-bit) is a cryptographic algorithm that processes file's content and outputs a fixed length unique 64-character string.
Why it is used: A hash acts as a digital finger print of sorts for a file. Even if the threat actor changes the filename the SHA-256 hash stays the same. The hash is also used for querying a threat intelligence database, VirusTotal, to see if the global cybersecurity community has already analyzed and flagged this exact file.

Shannon Entropy

What it is: In computing entropy is used to measure the degree of randomness or unpredictability of data within a file, it is calculated on a scale from 0 to 8. Standard, uncompiled English text has low entropy, while heavily randomized data would typically reach an entropy of 8.
Why it is used: Legitimate, uncompressed executables usually have moderate entropy anywhere between 4.0 and 6.0. An unusually high entropy score, typically above 6.5 or 7.0 would suggest the file has been compressed, packed or encrypted. Malware authors frequently use "packers" to obfuscate their code, hiding the malicious payload from traditional signature based antivirus scanners.

Strings

What it is: In computing *strings* are sequences of printable characters embedded within a binary file. The extracting tools parse a file, skip non-printable machine code. By default they look for sequences of at least 4 characters that end in a null terminator or a newline.
Why it is used: Extracting strings is a foundational step in static analysis and it allows us to peek inside a program without actually running it. It can reveal Network indicators such as IP addresses, URLs or domains used for Command & Control (C2)

communication . File paths of where the program might store logs, drop payloads or look for various files.

Our script looked for these four criteria:

- **hasShell (Command Execution):** Triggered by strings like powershell, `cmd.exe` or sh. This indicated the executable attempts to interface directly with the command line.
- **hasNetwork (C2 communication):** Triggered by `http://` or `https://`. While common in legitimate software, in cracked executables this often points to the malware fetching additional payloads from external servers or establishing C2 beacons.
- **hasCrypto (Obfuscation):** Triggered by terms like Encrypt, Decrypt, AES, or RSA. The presence of cryptographic libraries in a simple software crack strongly correlates with ransomware behavior or secure communication with a C2 server.
- **hasRegistry (Persistence Mechanisms):** Triggered by paths such as `Software\Microsoft\Windows\CurrentVersion`. Malware heavily targets this registry path (specifically the Run or RunOnce keys) to ensure the malicious payload automatically survives a system reboot.

The Selection of Static Over Dynamic Analysis

In our investigation into distribution of malware via torrent networks selecting the appropriate analysis was critical. As stated before we elected to rely exclusively on static analysis. While dynamic analysis can provide deep insights into how a malicious program interacts with a system, several significant constraints made it impractical for the scope of this study.

Time and Scalability

The most prohibitive factor against dynamic analysis was time and manpower. Dynamic analysis is an inherently slow and complicated process. To safely observe a piece of malware, it must be run inside a sandbox, the same way we used sandboxes for our static analysis, however each sample must be executed. This step is mostly manual as it would require us to follow the installation instructions. To fully understand how the sample behaves it must be monitored for a significant amount of time, minutes or even hours. After collecting every single change, action, network request, etc that happened during

the observation the sandbox has to be wiped and the data manually analyzed. Modern Windows installs are extremely noisy making the analysis even more complex. When multiplied by the amount of samples we wanted to analyze, the data collecting step would quickly turn into months of continuous processing. Static analysis, by contract, can extract essential indicators of compromise in fractions of second per file.

Initial Testing and Tool Limitations

Before finalizing our methodology, we conducted a pilot phase where we explored a hybrid approach using several industry-standard tools:

- **Detect-It-Easy (DIE):** We utilized this static tool to quickly identify file types and determine if the malware was "packed." Packing is a technique malware authors use to compress or scramble their code to hide it from antivirus software. DIE was highly effective for quick triage.
- **Ghidra:** We explored using Ghidra, an advanced software reverse-engineering suite developed by the NSA. While incredibly powerful for taking apart a program's code line-by-line (a form of deep static analysis), it requires significant manual effort and time per sample, making it unsuited for bulk analysis.
- **CAPE Sandbox:** We attempted to automate dynamic analysis using CAPE (Configuration and Payload Extraction), a highly regarded open-source sandbox. While CAPE is excellent at what it does, we immediately ran into the scalability bottlenecks mentioned above.

Ultimately, the sheer volume of samples acquired from torrent ecosystems required an approach that prioritized speed, safety, and automated scalability. By utilizing static analysis we were able to process the entire dataset efficiently while still extracting the high-confidence data required to draw meaningful conclusions for this study.

Our initial analysis pipeline

Before executing or decompiling any binary, we utilized DIE to determine the file's architecture, compiler, and the presence of any packers or obfuscation layers. This step remained largely the same in the actual pipeline we ended up using.

The next step was to load samples into Ghidra. We utilized Ghidra's automated analysis to parse the PE (Portable Executable) headers, map functions, and generate pseudo-C code using its built-in decompiler. The goal was to identify hardcoded Strings, API import tables, and command-and-control (C2) infrastructure markers.

To observe the malware's runtime behavior, samples were submitted to our CAPE environment.

Ultimately, this approach did not produce actionable telemetry for the broader scope of this study.

Example result

The analyzed sample uses an advanced packer or a crypter to evade traditional antivirus signatures.

- DIE flagged Anomalous resources. This indicates a presence of an encrypted payload, likely an actual malware, hidden within the resource section of the analyzed sample.
- The analyzed sample modified the following registry `\REGISTRY\USER\S-1-5-21-2714754311-527593155-1440850301-1000_Classes\WOW6432Node\CLSID\{018D5C66-4533-4307-9B53-224DE2ED1FE6}\Instance\`. This CLSID is associated with Microsoft OneDrive. By modifying the "Instance" subkey, the malware attempts to hijack the OneDrive shell extension. This allows the malicious code to be executed automatically by explorer.exe whenever the user interacts with the file system, providing a stealthy method of persistence that survives reboots.

MITRE ATT&CK

- T1082 System Information Discovery
- T1614 System Location Discovery
- T1614.001 System Language Discovery

Conclusion

The analyzed sample is possibly a Trojanized Loader. The dormant or "clean" appearance to many scanners is indicative of modern credentials stealers or ransomware droppers as also suggested by the registry persistence and sandbox evasions.

Case Study: “Avira System Speedup Crack”

To illustrate the effectiveness of this pipeline, we can observe a sample drawn from the torrent network, advertised as a cracked version of "Avira System Speedup."

The analysis yielded the following data points:

- **File Name:** Avira System Speedup Crack.exe
- **SHA-256 Hash:** 2e0ae36d94bc3bf0557e8234b1e82c1bf5d6d4510cfe69126bc50c757e4b3dbe
- **Entropy:** 6.37
- **VirusTotal Detection:** 37 out of 72 security vendors flagged the file as malicious.
- **Community Verdict:** Inconclusive (1 malicious and 1 harmless vote)
- **Heuristic Flags:** hasShell, hasNetwork, hasCrypto, hasRegistry

Analysis of the Sample:

Despite masquerading as a system utility crack, the static analysis overwhelmingly points to malicious intent. A VirusTotal score of 37/72 confirms its widespread recognition as malware. The entropy score of 6.37 is moderately high, suggesting potential code obfuscation or packing.
Most alarmingly, the string analysis triggered all four heuristic flags. The combination of network capabilities, shell execution, registry manipulation, and cryptographic functions within a single "crack" executable strongly suggests a sophisticated payload. This behavioral profile is consistent with ransomware or a remote access trojan (RAT) that establishes persistence via the registry, communicates over the network, and utilizes cryptography to lock user files or hide its traffic.

Analyzed samples

We have analyzed a wide range of applications from the most popular torrents (as of writing) on popular torrent trackers **piratebay** and **limetorrents**. Some of the analyzed softwares are Microsoft Office, Adobe Photoshop or Adobe Acrobat.

For each torrent file we tried to identify files users would either be told to execute or copy paste into the installation directory to crack the application.
Usually users were told to install applications as if they obtained them legally, after they would run a specialized crack executable. Another approach was to copy Dynamically Linked Library (DLL) files or alternatively patched executables.

The following tables contain the names of analyzed samples, their SHA-256 hash, link to an archived version of the website we used to obtain the torrent files. The last row contains the entropy, VirusTotal score at the time of analysis, the VirusTotal community score at the time of analysis and the results of checking for strings containing Shell, Network, Crypto and Registry.

Microsoft office

We have analyzed 23 samples in total and 6 were flagged containing at least one possible malware finding.

setup.exe						
2cae07a5ffae3a8b4cac401750c9345f52baa1d282fe0ee71edd57552270b61e						archive
En: **3.14946**	VT: **1/70**	Com: **1/7**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup32.exe

1ce1bdb84a08c1225b382c2043820091004efe76452833b6bdce54365e20ba56						archive
En: **6.58645**	VT: **0/72**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup64.exe						
6e188011a060831757da0ab9b5e849f9d48f897836c53a931136fd50cef13c29						archive
En: **6.25385**	VT: **0/71**	Com: **0/5**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

KMSAUT~1.EXE						
69a8ae6352cffd366409df8e566e84315b4bffcf5865a4b8079c446123ba1d26						archive
En: **7.08023**	VT: **37/62**	Com: **92/311**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

KMS_VL_ALL_AIO_v53.cmd	
e6fb3d61504524a484044122401317e89a9e5bd783d79d11dba19db7edeaad44	archive

En: **6.01268**	VT: **32/61**	Com: **5/3**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe						
03feb1535f7b14b2aa20707a9889dd5905d24582a6a0a0a406f1175819d3e75a						archive
En: **6.55234**	VT: **0/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Office 365 Setup.cmd						
d9bad799edf54f91bbc4cf2151ba910bf179d6c9174e175400543b76f2dbb06f						archive
En: **4.95168**	VT: **0/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Microsoft Office 2010 Powerpoint x64 64bit.iso						
98b6ab2a7e1cad63efd31f4d320c250f80ae14099883ea1f7414db644b62bea7						archive
En: **7.93045**	VT: **0/48**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup.exe						
a9698cd1db7c8f96658cb34abcfa02589131c16b09dc0f7bf5d161f66379080c						archive
En: **5.01800**	VT: **0/72**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup64.exe						
8fe6926e83e4edabd2704e5c55a7584a059c9b396e941c8612d8ada016b2f48d						archive
En: **6.58540**	VT: **0/72**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup32.exe						
695631fdd5de4b53fecc0fdb82bee8aa73d376e7cbf50c099c034dd2933a1bfc						archive
En: **6.58540**	VT: **0/73**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

MAS_AIO-CRC32_8C3AA7E0.cmd	
d666a4c7810b9d3fe9749f2d4e15c5a65d4ac0d7f0b14a144d6631ce61cc5df3	archive

En: **5.68395**	VT: **19/61**	Com: **3/3**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

KMS_VL_ALL_AIO.exe						
2ce96dd0e86edbad2d62af8ccd66247fcbaa928ffd47efff08db131254ce7e74						archive
En: **6.71926**	VT: **52/71**	Com: **2/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Check_Activation_Status.cmd						
b808591c2933f483149c0f46101fd9649baa1abbf5a9f1b1cd770c0ff6c97be0						archive
En: **5.61628**	VT: **0/60**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

MAS_AIO_3.11.cmd						
419975aac5c6c1159542f4feb48b723c6c867fa2f1acfbc1a454c8415dc28f72						archive
En: **5.47645**	VT: **11/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe						
6cd73bc9ec16e9dd1eb07493c464a0537f82d5096bae20f04c41cf5486ce9b88						archive
En: **6.61030**	VT: **0/62**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Office 365 Setup.cmd						
406f3c99b48cc9d38ca81139e6d17233e5e5bd65403321c5be368e73623fe02e						archive
En: **4.94322**	VT: **0/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Activation.pkg						
083b505d9ac880629d7ef560e2baaaf59343721ac7e892be737a2403d7a63de1						archive
En: **7.98538**	VT: **0/61**	Com: **1/6**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

msvcr100.dll	
60c06e0fa4449314da3a0a87c1a9d9577df99226f943637e06f61188e5862efa	archive

En: **6.90157**	VT: **0/71**	Com: **16/41**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

cleanospp.exe						
04ba4487f95290e0b0557b44300c18f637fbaf0872ee96e3111013b8a1539f25						archive
En: **6.31517**	VT: **0/72**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Check_Activation_Status.cmd						
b808591c2933f483149c0f46101fd9649baa1abbf5a9f1b1cd770c0ff6c97be0						archive
En: **5.61628**	VT: **0/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe						
670487a863512962a964da9f8dcce00052f65a26f2676880e7eb74ad464ba104						archive
En: **6.57138**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Office 365 Setup.cmd						
e06b4a8dfe0aa1886028b6a2b75dc8b90298f8d5630632e162ecc1be91169221						archive
En: **4.94325**	VT: **0/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Check_Activation_Status.cmd						
b808591c2933f483149c0f46101fd9649baa1abbf5a9f1b1cd770c0ff6c97be0						archive
En: **5.61628**	VT: **0/60**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

MAS_AIO_3.11.cmd						
419975aac5c6c1159542f4feb48b723c6c867fa2f1acfbc1a454c8415dc28f72						archive
En: **5.47645**	VT: **11/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe	
6cd73bc9ec16e9dd1eb07493c464a0537f82d5096bae20f04c41cf5486ce9b88	archive

En: **6.61030**	VT: **0/62**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Office 365 Setup.cmd						
406f3c99b48cc9d38ca81139e6d17233e5e5bd65403321c5be368e73623fe02e						archive
En: **4.94322**	VT: **0/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Activation.pkg						
083b505d9ac880629d7ef560e2baaaf59343721ac7e892be737a2403d7a63de1						archive
En: **7.98538**	VT: **0/61**	Com: **1/6**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

msvcr100.dll						
60c06e0fa4449314da3a0a87c1a9d9577df99226f943637e06f61188e5862efa						archive
En: **6.90157**	VT: **0/71**	Com: **16/41**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

cleanospp.exe						
04ba4487f95290e0b0557b44300c18f637fbaf0872ee96e3111013b8a1539f25						archive
En: **6.31517**	VT: **0/72**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Check_Activation_Status.cmd						
b808591c2933f483149c0f46101fd9649baa1abbf5a9f1b1cd770c0ff6c97be0						archive
En: **5.61628**	VT: **0/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe Photoshop

We have analyzed 36 samples in total and 16 were flagged containing at least one possible malware finding.

Photoshop.exe						
0b3a173c70e9c7c69af181286c43b5d01a88cfe3d146f6257018ea38b4b03443						archive
En: **6.63082**	VT: **1/67**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
24d2666c00ecd02350af0d70c8a9b71ed2bf0ce2553e61506fc1cbba0a9156b3						archive
En: **6.43392**	VT: **2/70**	Com: **17/24**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
b9a187b59c758ead0022e50bbaae4133d2e37b769a054249afc0b6aa2e26774d						archive
En: **6.42908**	VT: **0/66**	Com: **18/27**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
4460dd8114b5609ea4e9644a659de0f5b188696d27dc8846d633628b3ade7c31						archive
En: **6.43393**	VT: **1/70**	Com: **51/143**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Photoshop.exe						
6a1c8e092b7955d1df9875fbb7c0cd68bc30d799c6b9c3fda2f4f3755bef3c3b						archive
En: **6.62107**	VT: **1/70**	Com: **1/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePatchInstaller.exe						
011a0e9487b2f387ff11e96d62023bf58dd16443fdb129b15285d0707701746c						archive
En: **7.00565**	VT: **0/72**	Com: **0/22**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

amtlib.dll						
f7c93c9f262a94360ecef3725ed20dc3b43bfad4243ab3fdaf5b8e56222e3f54						archive

En: **6.32304**	VT: **0/71**	Com: **378/1338**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

amtlib.dll						
e1189544e7fe546133d119a141f1d1ec74a1fed2317c7dfb211c2b779887c03e						archive
En: **6.68674**	VT: **0/71**	Com: **534/1416**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
c2802a86bfbb37288f00d03425a60e0efd60db9ed113ff39c4e6df0efba66a5d						archive
En: **5.90885**	VT: **0/70**	Com: **45/167**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

updaterinventory.dll						
25df5b89e67dff662889f4cb971ed187a5edd79cb17078034b194f7102d28a85						archive
En: **6.50789**	VT: **0/72**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePIM.dll						
48140a6d158c3da0e42bd745405360917241c5d52c779d97501e218c205042fa						archive
En: **6.63821**	VT: **0/56**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
582536081e67975055ccf8de5353502d7bda56f2aafcbefbb400f3d9012019c9						archive
En: **6.36856**	VT: **0/71**	Com: **16/55**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePIM.dll						
b10756942a010d67ae7d01fe759d9e94261d2f93cdf7bdb971fc78c146d0674a						archive
En: **6.66688**	VT: **0/72**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe	
65e076a8e2e20cdadee7f0ef907ae1ca1860345412ffe94603cfa8f0d29fec08	archive

En: **6.34222**	VT: **0/69**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

oem.exe						
c72e51f10ad3fd7b7231bda95e0a938e381a4a119b70b7c6d73c73c2ec163381						archive
En: **6.76147**	VT: **0/71**	Com: **2/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Photoshop.exe						
a7bb90034ddacbc8d8d77aa0adb40719da92ea48a40cd39f5b5462c73a299c5a						archive
En: **6.72313**	VT: **1/70**	Com: **5/12**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39						archive
En: **6.43080**	VT: **0/71**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
b72bf33ae94f3e91acc279c2a106382762ccc5bc0e7e7a02a148d4ef531eea92						archive
En: **6.43388**	VT: **37/71**	Com: **5/3**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobeGenP.exe						
cf3bf1973ed8a75fc816781cd39dd19f8c2c72f27a50d7506ada4e7c87b5913a						archive
En: **7.81775**	VT: **50/71**	Com: **2/8**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe	
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82	archive

En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
50c728125c297e0bd5eaada1364e8ba6eb1089ec2a346853674cd61c87d02633						archive
En: **6.43391**	VT: **20/71**	Com: **10/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePIM.dll						
7917708a9f343067e01242423eaa73ae981e20ce07ace6274fdcc71ee2b03b51						archive
En: **6.68496**	VT: **0/70**	Com: **2/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
732cea9b7e280d864e7134d27b8b8384d44c09100d6557aedfc17841838e3f3e						archive
En: **6.58014**	VT: **1/70**	Com: **5/13**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

amtlib.dll						
9b510f38acb2a70c840b17a9fed2584f08bf38395f087212dd41b059f791126f						archive
En: **6.33750**	VT: **0/71**	Com: **360/704**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
732cea9b7e280d864e7134d27b8b8384d44c09100d6557aedfc17841838e3f3e						archive
En: **6.58014**	VT: **1/70**	Com: **5/13**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

amtlib.dll						
f42f0088578e0454acba11691dfaf25bec1124f32070245a855293fbda6af621						archive
En: **6.68624**	VT: **0/72**	Com: **231/375**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

TechTools.NET – Adobe Photoshop CC 2015 FULL Portable.exe	
3c5018a01c8001d297df7ccfb1bf4e1ee256c22a94442b325e301e57aaa5d274	archive

En: **8.00000**	VT: **1/53**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
210608aa2b791cf156325a34ffbaad33cc4fd8dad2ae89dec6126c7f433b2382						archive
En: **7.33444**	VT: **4/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39						archive
En: **6.43080**	VT: **0/71**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePIM.dll						
7917708a9f343067e01242423eaa73ae981e20ce07ace6274fdcc71ee2b03b51						archive
En: **6.68496**	VT: **0/70**	Com: **2/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39						archive
En: **6.43080**	VT: **0/71**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePIM.dll						
7917708a9f343067e01242423eaa73ae981e20ce07ace6274fdcc71ee2b03b51						archive
En: **6.68496**	VT: **0/70**	Com: **2/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

GenP-v3.8.0.exe						
6721aaaa5b8a0bbc1bc68797e6b5f32b65129896c4e2e082f995446f7a9b5f8d						archive
En: **7.81807**	VT: **44/71**	Com: **1/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe exe firewall block windows.bat	
169fb0435fb83e23f7f5e46ca6aeea4410fc293c32c92e770c05e1155cf80e04	archive

En: **4.86673**	VT: **0/60**	Com: **1/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe.Photoshop.2026.v27.5.0.13.part1.exe						
fa3f35540280a42a587b84c224bca15a8592a8821f4b8960a2b89bf2024134b4						archive
En: **7.33404**	VT: **3/69**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe Acrobat

We have analyzed 36 samples in total and 17 were flagged containing at least one possible malware finding.

amtemu.v0.9.1-painter.exe						
c2f6d462a20f92b97c49c3af19872fc4df6aabd4f66f4b8b298a1303881422f6						archive
En: **7.41964**	VT: **47/70**	Com: **14/9**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AcrobatPro.exe						
9e665c575c0e145580cb38ad4aa69621225ac0e2d5c63a2983bae37c706519fe						archive
En: **7.99998**	VT: **0/64**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AcrobatUpd11020.msp						
46008fe68e2c7d2a00da29a9a2c85b52dea7fdcac3f84a71c6d3caa2e15223cd						archive
En: **7.98962**	VT: **0/57**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe.Acrobat.Reader.v2023.006.20380.exe						
9d4eec6495582c752b076a24fae37eff4e0643c8d8cb2b331936aa7e6dce1b80						archive
En: **7.99974**	VT: **12/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

INSTALL.cmd						
00cb805a58d2fda034f768f9e8a799654e58a10e63176dd1c5154228f6536cdf						archive
En: **2.93416**	VT: **0/61**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

ADBEAR~1.EXE						
f134cdafd92d95428e8b5795851621e493936def21c2c1e6bb084c8630d826f4						archive
En: **5.81791**	VT: **0/72**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe	
26001d6af714fcae74af8ecbee993d017e0005b1a5ae0cc7213963d882d279f6	archive

En: **6.02657**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

WINDOW~1.EXE						
69b61b2c00323cea3686315617d0f452e205dae10c47e02cbe1ea96fea38f582						archive
En: **7.97622**	VT: **0/70**	Com: **23/50**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AcroPro.msi						
089dc9a23f5eb868a6f6b8a6a901c2a29faef113ad296c8d40f6ce9de60f4b2d						archive
En: **5.58371**	VT: **0/62**	Com: **1/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

ACROBA~1.MSP						
39e806ff65b1fd805e4361333941bcedf9f0184f8540909894a376a3a89d4e4c						archive
En: **7.99596**	VT: **0/57**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobeGenP.exe						
64f1ea7bf1b94f612d72ab74b36c11108b4b798adba3f2db79f4d5923e6d580a						archive
En: **6.42632**	VT: **27/71**	Com: **2/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

RunAsTI.exe						
a3e0ba70ba908de8a75825c3a1ff36147e02c686280993c2caa8a9a6968764b0						archive
En: **5.73005**	VT: **2/70**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

ADOBEG~1.EXE						
64f1ea7bf1b94f612d72ab74b36c11108b4b798adba3f2db79f4d5923e6d580a						archive
En: **6.42632**	VT: **27/71**	Com: **2/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe	
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82	archive

En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

WindowsInstaller-KB893803-v2-x86.exe						
69b61b2c00323cea3686315617d0f452e205dae10c47e02cbe1ea96fea38f582						archive
En: **7.97622**	VT: **0/70**	Com: **23/50**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup.exe						
04889538532f10f910029a5ef65068aed713019fb50462546e82ee8ced81350b						archive
En: **6.08184**	VT: **0/70**	Com: **0/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

crack.exe						
c75b9e099c891b53ef180f0ac26f5188c9fc9fc07d019cf19687fe05116b0957						archive
En: **7.98167**	VT: **22/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup.exe						
04889538532f10f910029a5ef65068aed713019fb50462546e82ee8ced81350b						archive
En: **6.08184**	VT: **0/70**	Com: **0/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

crack.exe						
9f7bde927e552aa82b0ce168b2e59845b89eb600ec290d5e8b3fb1a9e73325a7						archive
En: **7.99593**	VT: **18/68**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

WindowsInstaller-KB893803-v2-x86.exe	
69b61b2c00323cea3686315617d0f452e205dae10c47e02cbe1ea96fea38f582	archive

En: **7.97622**	VT: **0/70**	Com: **23/50**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe						
c8b7f3add72e8b3b6a89bacd763d81c769e15ed0bc25eaf48776f286ad95d5f0						archive
En: **6.03094**	VT: **0/70**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/70**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup.exe						
04889538532f10f910029a5ef65068aed713019fb50462546e82ee8ced81350b						archive
En: **6.08184**	VT: **0/70**	Com: **0/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

crack.exe						
9f7bde927e552aa82b0ce168b2e59845b89eb600ec290d5e8b3fb1a9e73325a7						archive
En: **7.99593**	VT: **18/68**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

GenP-v3.8.0.exe						
6721aaaa5b8a0bbc1bc68797e6b5f32b65129896c4e2e082f995446f7a9b5f8d						archive
En: **7.81807**	VT: **44/71**	Com: **1/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe exe firewall block windows.bat						
169fb0435fb83e23f7f5e46ca6aeea4410fc293c32c92e770c05e1155cf80e04						archive
En: **4.86673**	VT: **0/60**	Com: **1/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe	
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82	archive

En: **6.19240**	VT: **1/70**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup.exe						
04889538532f10f910029a5ef65068aed713019fb50462546e82ee8ced81350b						archive
En: **6.08184**	VT: **0/70**	Com: **0/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

crack.exe						
3e2855ca5f3b34b3d4e152ec6dfd55831d77383a8bca7c68950d0712af72d3b8						archive
En: **7.99592**	VT: **14/69**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/70**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup.exe						
04889538532f10f910029a5ef65068aed713019fb50462546e82ee8ced81350b						archive
En: **6.08184**	VT: **0/70**	Com: **0/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

crack.exe						
3e2855ca5f3b34b3d4e152ec6dfd55831d77383a8bca7c68950d0712af72d3b8						archive
En: **7.99592**	VT: **14/69**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

WindowsInstaller-KB893803-v2-x86.exe	
69b61b2c00323cea3686315617d0f452e205dae10c47e02cbe1ea96fea38f582	archive

En: **7.97622**	VT: **0/70**	Com: **23/50**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup.exe						
04889538532f10f910029a5ef65068aed713019fb50462546e82ee8ced81350b						archive
En: **6.08184**	VT: **0/70**	Com: **0/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Wondershare Filmora

We have analyzed 7 samples in total and 7 were flagged containing at least one possible malware finding.

FCommonView.dll						
023a4a0a4ff41d7be85edd39f2ec23d2c649d3a60f99f7cba5ff5dbbb7589573						archive
En: **7.84225**	VT: **6/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FCore.dll						
4113842f8d43c1c4909374bd02a6521b146988f52c324c965747753c409ff878						archive
En: **7.84278**	VT: **1/67**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FExportView.dll						
20f79cd8cb14a912aa4d677b8094833394644b75cc6b4eeeb3268eec72a89a0f						archive
En: **7.90342**	VT: **1/59**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FFWsRegister.dll						
9ef511c46261a6122097dc38179c1f944f0b5090d449642418dc844df9b3f39a						archive
En: **7.85005**	VT: **37/71**	Com: **2/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FFWsUpgrade.dll						
54770bbda84d6809c64fea15f7575a07eadda865216fbce5ec486fd4ca45dcc6						archive
En: **7.84324**	VT: **37/71**	Com: **1/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

version.dll						
37685f2dd6318feb074544d0166ab80303c89c1a7b28a4892e003cfa7a1278ee						archive
En: **7.96073**	VT: **35/71**	Com: **1/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

version.dll	
37685f2dd6318feb074544d0166ab80303c89c1a7b28a4892e003cfa7a1278ee	archive

En: **7.96073**	VT: **35/71**	Com: **1/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FL Studio

We have analyzed 5 samples in total and 1 were flagged containing at least one possible malware finding.

FL64.exe						
9eeb6c08d235d41f18a2d37a305cbb3513e6974765a84669f093b8b45cee6b71						archive
En: **7.24215**	VT: **0/72**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FL64 (scaled).exe						
f12290c79a9ce16f6f61633628e1d22200187910c520be77139ede388f657aa0						archive
En: **7.23037**	VT: **0/71**	Com: **1/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

fl_patch_installer_20_7_2_1852.exe						
33d9b96fac6369e24d59a53538d0c4e5c0e256521bc1326c6e8fbe4373ed5aea						archive
En: **7.99998**	VT: **16/69**	Com: **11/63**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FLEngine_x64.dll						
07d929cbfaa818f0c1d3879cf89f893b162ab9058d858230e5a20cf6ec975afb						archive
En: **6.87640**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FL64.exe						
680cfb77eaecde89e90c88a3b7d477a7b24402a4e12e583c853822d88db4731d						archive
En: **6.66271**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

FL64 (scaled).exe						
d6007791cdf1ba71c03319441ecaeb3b1de4e32be5a59c467c597509a66944ee						archive
En: **6.66336**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

KMSpico

We have analyzed 6 samples in total and 5 were flagged containing at least one possible malware finding.

KMSPIC~1.EXE						
88f11abdd3e82c4ff30c0b67d4af73e10df6f83d6cbe0ce4f94fc2b2ebc013b8						archive
En: **7.99642**	VT: **51/71**	Com: **21/29**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

UNINST~1.CMD						
0552a48861a2c9825d51eeb0197a959dc85e4e960fb00cee89ccc4806eaadba8						archive
En: **4.42915**	VT: **1/39**	Com: **6/19**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AUTO(R~1.CMD						
7339a4cc48220a161fcc737ed26e99e5678a4d1faa3f7e2686c46b5a5d234828						archive
En: **5.12596**	VT: **12/60**	Com: **3/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AutoPico.exe						
470a75fe3da2ddf9d27fb3f9c96e6c665506ea7ba26ab89f0c89606f678ae4a2						archive
En: **6.32282**	VT: **50/71**	Com: **6/6**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

KMSELDI.exe						
91539a89fb6531ad4e52e8b19bfe02ec4cbb22393bc0058cc15f56d926017ac7						archive
En: **6.75512**	VT: **51/71**	Com: **5/12**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

REMOVE~1.CMD						
b991f07494e570a55efa8f93108a277ff110a38a8e4bc9bd47e894e872063ec1						archive
En: **4.57128**	VT: **0/62**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Topaz Video

We have analyzed 5 samples in total and 3 were flagged containing at least one possible malware finding.

Topaz Video AI Pro v7.1.1.exe						
651eedf0ebfdd5dda63fca0d9b75f0954dc54069ee79f77477a15220e8eadc67						archive
En: **7.99984**	VT: **0/71**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

afterpatch.exe						
38b74567533fe1545f9d417cd4ef61f879e14908a3185bb7fc7db89207afdea6						archive
En: **7.53962**	VT: **1/57**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

TopazVideo-1.5.0.msi						
3f94f7e3b8e3c45d7859eb2d166288c6edc714e7f4db08829abea643e019df0c						archive
En: **7.99523**	VT: **0/57**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

version.dll						
ca2806755c3e1fdc02179735cb91c01449989c727afb9eeb930d5f62b90ed165						archive
En: **5.65877**	VT: **15/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

dxgi.dll						
c60a213b795a0d662f55dac9a34b2123a8df53c359cbd8a4365f2d45b7f932ea						archive
En: **5.63499**	VT: **20/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Avira System SpeedUp

We have analyzed 2 samples in total and 1 was flagged containing at least one possible malware finding.

Avira System Speedup Crack.exe						
2e0ae36d94bc3bf0557e8234b1e82c1bf5d6d4510cfe69126bc50c757e4b3dbe						archive
En: **6.37527**	VT: **37/72**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

avira_en_asu80_137606563-1549701162__ws.exe						
3cd7ee9260ec23c688605b65dfabf4f0f0c32d871fc5f650aa49dc09cb52ed18						archive
En: **7.97854**	VT: **0/68**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe Illustrator

We have analyzed 15 samples in total and 8 were flagged containing at least one possible malware finding.

Set-up.exe						
24d2666c00ecd02350af0d70c8a9b71ed2bf0ce2553e61506fc1cbba0a9156b3						archive
En: **6.43392**	VT: **2/70**	Com: **17/24**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Illustrator.exe						
b2959d2713c20cabd2b9fc3cae6c671fbc77bcd60de1a1bdc7d905c811a7bf8a						archive
En: **6.38403**	VT: **0/72**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

dvaappsupport.dll						
6b3873e797f8dba15a722a732ee10fe87180a113146cbbd40720e6a3de96f1cd						archive
En: **6.45565**	VT: **0/72**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
603c4e18b36c46325121c35128da7ba0a94b673ff30437f9b33a69e9ce5110e7						archive
En: **6.43391**	VT: **40/70**	Com: **7/5**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

GenP-v3.6.9.exe						
cf3bf1973ed8a75fc816781cd39dd19f8c2c72f27a50d7506ada4e7c87b5913a						archive
En: **7.81775**	VT: **50/71**	Com: **2/8**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePIM.dll						
7917708a9f343067e01242423eaa73ae981e20ce07ace6274fdcc71ee2b03b51						archive
En: **6.68496**	VT: **0/70**	Com: **2/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe	
830210afefdd2210cfda5ce0d898b56ff196b940acde7f22d06aadae850ee78b	archive

En: **5.65442**	VT: **27/72**	Com: **1/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
0f8ccdf2162915f2d056702e9df32c65e4ed2e0c0d5d5ea606aa8803b4ef5d9b						archive
En: **6.58015**	VT: **0/65**	Com: **3/22**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

amtlib.dll						
65897650b5f6d39c3c55cb7a0fdbf5896dcff815878fce52ec5c86ccd4e006a0						archive
En: **6.36718**	VT: **0/71**	Com: **0/7**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39						archive
En: **6.43080**	VT: **0/69**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39						archive
En: **6.43080**	VT: **0/71**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
603c4e18b36c46325121c35128da7ba0a94b673ff30437f9b33a69e9ce5110e7						archive

En: **6.43391**	VT: **42/71**	Com: **7/5**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

GenP-v4.0.2.exe						
cc2f424dd53409dc7c23db7f3e2298e66138ecec272e87336438346b9b422c35						archive
En: **7.28099**	VT: **23/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe Lightroom

We have analyzed 7 samples in total and 3 were flagged containing at least one possible malware finding.

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39						archive
En: **6.43080**	VT: **0/71**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39						archive
En: **6.43080**	VT: **0/71**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

GenP-v3.8.0.exe						
6721aaaa5b8a0bbc1bc68797e6b5f32b65129896c4e2e082f995446f7a9b5f8d						archive
En: **7.81807**	VT: **44/71**	Com: **1/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe exe firewall block windows.bat						
169fb0435fb83e23f7f5e46ca6aeea4410fc293c32c92e770c05e1155cf80e04						archive
En: **4.86673**	VT: **0/60**	Com: **1/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe	
3d20655679c8829a6baad001851905927ef1b826e3eea594b7be3f8331211e39	archive

En: **6.43080**	VT: **0/71**	Com: **8/14**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Adobe Premier

We have analyzed 6 samples in total and 2 were flagged containing at least one possible malware finding.

autoplay.exe						
3b9dabd99dc58a5242616cb6d1d876bca3046119a9b150c7d7868bf02202ea82						archive
En: **6.19240**	VT: **1/71**	Com: **17/86**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
37bafe751e9307c119b84d7247f7c1d6b5c63810f4ad67dfc8c1a6d1479bf4b2						archive
En: **6.43391**	VT: **35/65**	Com: **5/13**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
0f8ccdf2162915f2d056702e9df32c65e4ed2e0c0d5d5ea606aa8803b4ef5d9b						archive
En: **6.58015**	VT: **0/65**	Com: **3/22**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

amtlib.dll						
fa0d3b8e40d1500eb6ce1075b0698b6110fec80cbfab780932dd01f15e44f363						archive
En: **6.36722**	VT: **0/72**	Com: **5/25**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Set-up.exe						
16caee252577d59377e31c25225b0a6568764cc96074a566aca543ddf2eb1175						archive
En: **6.36887**	VT: **0/70**	Com: **8/11**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

AdobePIM.dll						
75277ea2d73f3c8344960173f9658d3b335f84d3842441d4257ae239938b49ea						archive
En: **6.66747**	VT: **0/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Devinci Resolve

We have analyzed 5 samples in total and 2 were flagged containing at least one possible malware finding.

Install Resolve 20.0.exe						
eceaa41e1a54b94913a61e4eb555d77e0d8d238e11d7627d32a36cb48c85f9ed						archive
En: **6.81702**	VT: **0/71**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Resolve.exe						
37a05ff82b833318e97634fac4cbb8c150a23720d16029b6be6dba81c2c28bf9						archive
En: **6.97131**	VT: **1/64**	Com: **2/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

DaVinci Remote Monitor.exe						
6633efca2eb07e6c1ae654b85d7be678728c0fc75e8b7f7d801ae4d66f746f57						archive
En: **5.91998**	VT: **1/69**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

fusionsystem.dll						
8052f5edc73695c5e6adab06daa0eb1be85b892dc8a873d6c9311bc8c0fb9b59						archive
En: **6.81869**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

fraunhoferdcp.dll						
0e35a63b9efbd745d6ee2829d8f274854843026d61be3021389d26ba7d1f9313						archive
En: **6.80201**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Sketchup

We have analyzed 12 samples in total and 3 were flagged containing at least one possible malware finding.

Activator.exe						
d759b44ef4b0e86c75b73383138d578029997f5458f287049f6c3e7d8b5852b9						archive
En: **5.09148**	VT: **46/70**	Com: **2/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

sup_2016_en_x64.exe						
04804d735d6b79318525936c320ef10154a5c8078770c838b0ef4045b45cad9b						archive
En: **7.99956**	VT: **0/65**	Com: **4/3**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Style Builder.exe						
b596ada5c6ad5933a9fe12f1e4525a1d721e97af14a6679cd1979f2ffe3745f1						archive
En: **6.27891**	VT: **0/72**	Com: **0/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

SketchUp.exe

fc24bb0724c9a02b12c0ec2302a71b7f382f70ca406d8e6011c9005c3d1e3d1c						archive
En: **6.59055**	VT: **0/72**	Com: **0/5**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

LayOut.exe						
b6ded0dffb3f318a4c999328cdbfc03904662ab03ee6b0cc0dca2f33fb72f347						archive
En: **6.52084**	VT: **0/72**	Com: **0/4**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

sup_2016_en_x32.exe						
9b4ec2984a6d4f8f9d2249ee55509330dd982dd4abe2b4f0875ab275e38a0e4e						archive
En: **7.99949**	VT: **0/68**	Com: **4/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Style Builder.exe	
2ae6954f7b42c3f4a0adcc62d6dae9108350ae159573a084c1aab1cc466d68f2	archive

En: **6.50046**	VT: **0/64**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

SketchUp.exe						
14e3f87bb16939735856cf817c9025f2946f5b58c5a9ce4e8f9dc97b93d1c62d						archive
En: **6.88886**	VT: **0/69**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

SketchUp.exe						
cc86f436414162bf02f3716fbd02f9a8571b3b6d35a792a725b3be4144b94da8						archive
En: **7.92788**	VT: **40/69**	Com: **3/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

LayOut.exe						
08098464f47a904898d197acf8dcdba12f3f982932c5ef224b8e27b9e64ec6a6						archive
En: **6.75962**	VT: **0/72**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

SketchUp.exe						
895b87a533c98f9f3ca034e1b4941bb400b81c07899b1c2bd0db0d3aab0305e2						archive
En: **7.94232**	VT: **40/71**	Com: **2/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

LayOut.exe						
f556d58c2e491be73229e9cff0f9f014a751363f1d3b79518e6bba61aba16182						archive
En: **6.74477**	VT: **0/63**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Solidworks

We have analyzed 13 samples in total and 3 were flagged containing at least one possible malware finding.

sw_d.exe						
59f60b9680f8d1220229ed0b4402e578f2da5f29ecf7152809a9428140399524						archive
En: **6.66199**	VT: **0/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

installs.exe						
5357844c0f6ca3154ca7f1ea552410738c9bfe92cdc81bfdfdf47f3c06da25ad						archive
En: **6.47147**	VT: **1/70**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

lmgrd.exe						
a2b03dbb2846c06c601256f3845a53151837251f6491264a69ec17cb4fdd0d5c						archive
En: **6.59199**	VT: **0/72**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

lmtools.exe						
0a36527d86c66c4758f4116e8f762ed04e11ed1791ff2b97f616d3c68b2a3be6						archive
En: **6.35644**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

lmutil.exe						
4ae9792e4b186d90b53f16d5aa16105bbec4967844e446b5214525d1b125c40e						archive
En: **6.33305**	VT: **1/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

server_install.bat						
49f922495fe79151e8b1c7e5bfdca11573eb4ac958888301e7d9126a1f000459						archive
En: **4.79433**	VT: **0/60**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

server_remove.bat	
8b0b95e49330c8fc517061ab420e0da66555a20a270ad88835eea05a591af04e	archive

En: **4.33656**	VT: **0/62**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

netapi32.dll						
8cf7e5ce33d7eff087f2891de80fb62324991716d4be86c540dbff90145792f5						archive
En: **6.63166**	VT: **0/72**	Com: **1/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

sw_d.exe						
b6235d88c4f52d764a3defc78e29f440bf70a26450a858a621acf6d35cd218c9						archive
En: **6.55994**	VT: **0/69**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

installs.exe						
5357844c0f6ca3154ca7f1ea552410738c9bfe92cdc81bfdfdf47f3c06da25ad						archive
En: **6.47147**	VT: **1/70**	Com: **1/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

lmgrd.exe						
f3288e15ba8cd5f8307a92b88c68191039bb2e61566c76463509ce2c289b9a8a						archive
En: **6.66928**	VT: **0/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

lmtools.exe						
0a36527d86c66c4758f4116e8f762ed04e11ed1791ff2b97f616d3c68b2a3be6						archive
En: **6.35644**	VT: **0/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

setup.exe						
2bf193185e520713964cf0710b882bd5ce7fc10c4a3ee8026858c51dba13d687						archive
En: **4.84185**	VT: **0/69**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

VirtualDJ

We have analyzed 10 samples in total and 7 were flagged containing at least one possible malware finding.

virtualdj_pc_v8.0.2048.msi						
c91801f6c01935aba883f3d23fabe25f44b338fc3e30444ca7cde11b4c1dd166						archive
En: **7.98536**	VT: **0/62**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

virtualdj8.exe						
70efecf6b6026553e50fb289a6d15539b03cdebdbef0a62364b4136209fd2105						archive
En: **7.26646**	VT: **1/70**	Com: **6/5**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

install_virtualdj_pc_v8.0.2265.msi						
a1764f122eaca5c22f956b2d8c393f40335a083aeea7d54641bce8533b660e57						archive
En: **7.98615**	VT: **0/61**	Com: **0/2**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

virtualdj8.exe						
c7e3976b4ff730eadb93ff0646bda31110664cbc020fa6c7efb924c9d9253070						archive
En: **7.28457**	VT: **1/69**	Com: **0/3**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup Virtual DJ v8.5.9295.exe						
121c5d7cb927d04b04a496fa0cf17705454010f590df6c36a07414e3ca59cc41						archive
En: **7.82573**	VT: **0/65**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup Turbo Activate Activator v1.0.0.exe						
bb0200a1deae607a2630ab2f0052ec25fcefbc3b2d8676aee356ddf6379ba732						archive
En: **7.32387**	VT: **3/59**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

R2R System v1.4.0.exe	
ba6954683a36669cc92f0545892ae686fc7cc8a223fccff416baf40ea0dceda2	archive

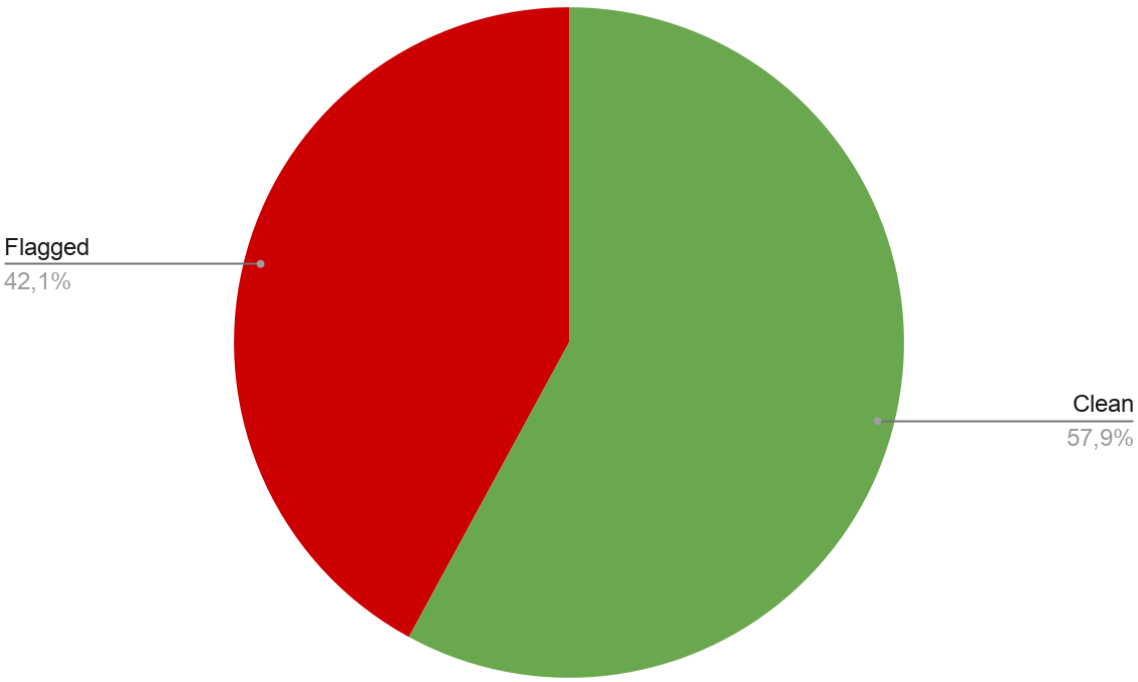
En: **7.99017**	VT: **1/71**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

R2RCERTEST.exe						
c7142fc351a4d2ee9d1e7e1a394d7b323f966f5f6e082feadb812525058d4319						archive
En: **5.21723**	VT: **4/70**	Com: **0/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup Network Block Runtime v2.3.0.exe						
899e9f8ecea98a8a058d3651d91b5c5d518378d4e3f9ad5957b5ae215fb4038c						archive
En: **7.48746**	VT: **2/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Setup INM810 – inMusic Activator v1.1.0.exe						
bbd9a562319f1c8ebb8236f2341d3d1ef7534de7fdb224fb8d8bfc599b33ab5c						archive
En: **7.15034**	VT: **5/70**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Data Analysis



The pie chart illustrates that in all of our samples **42.1%** were flagged by VirusTotal by at least one virus engine.

KMS_VL_ALL_AIO.exe						
2ce96dd0e86edbad2d62af8ccd66247fcbaa928ffd47efff08db131254ce7e74						archive
En: **6.71926**	VT: **52/71**	Com: **2/1**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

Sample **KMS_VL_ALL_AIO.exe** is the one returning the highest score of all of the analyzed samples

TechTools.NET – Adobe Photoshop CC 2015 FULL Portable.exe						
3c5018a01c8001d297df7ccfb1bf4e1ee256c22a94442b325e301e57aaa5d274						archive
En: **8.00000**	VT: **1/53**	Com: **0/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

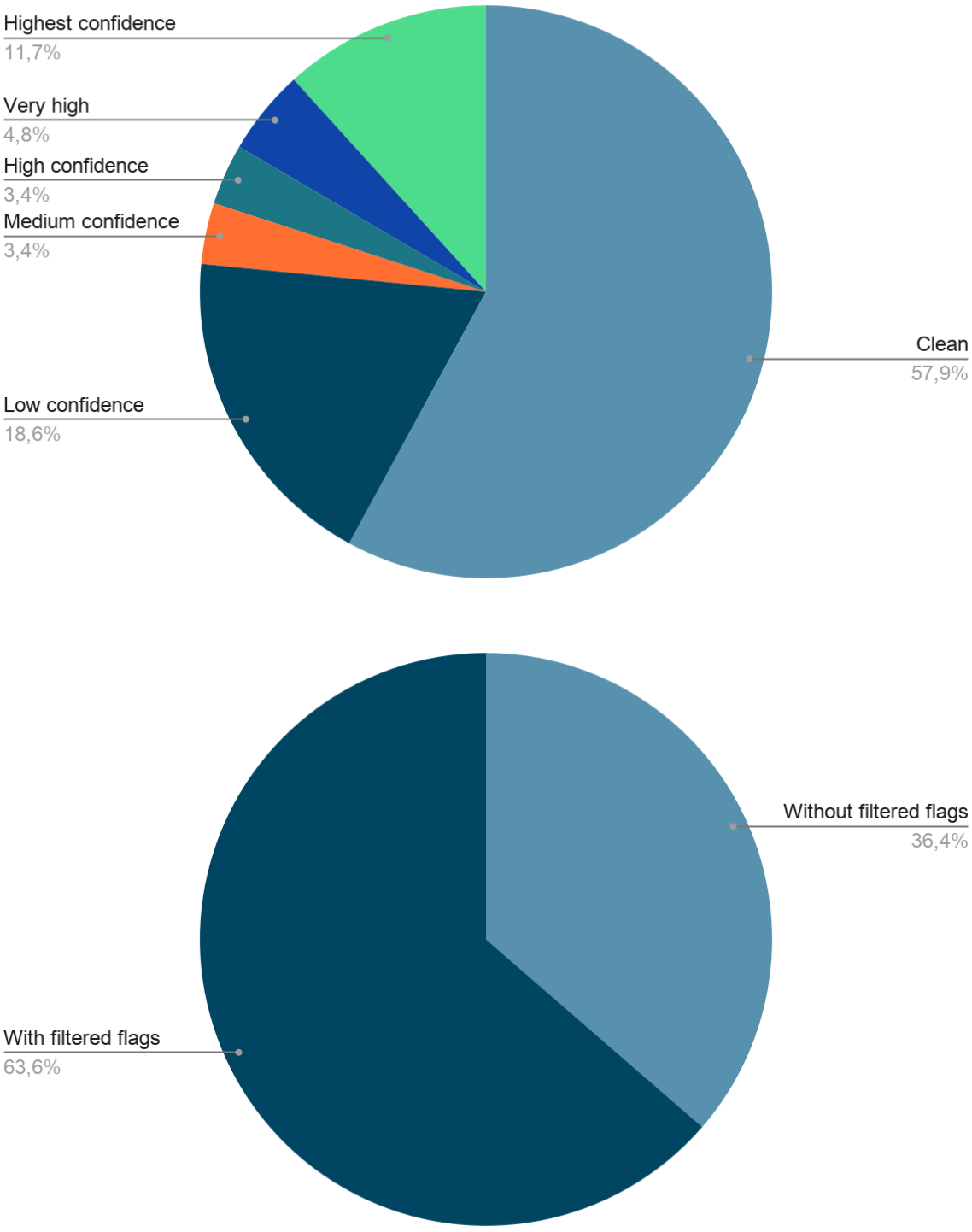
Sample **TechTools.NET – Adobe Photoshop CC 2015 FULL Portable.exe** is the one with the highest recorded entropy of all of the analyzed samples.

Filtered results

As we've stated we are using VirusTotal and other factors to mass analyze samples, this approach is unfortunately never going to be as precise as a dynamic analysis including manual decompilation of the analyzed sample to fully understand what happens under the hood. Nevertheless we feel this approach can still reveal many interesting facts about the security of downloading pirated content using the BitTorrent network as well as educating users on how to navigate this landscape and become more vigilant, even when downloading and running software from trusted sources.

We split our samples into 6 categories.

- **Highest confidence:** 31 or more flags with false positives flags filtered out
- **Very High confidence:** 21 to 31 flags with false positives flags filtered out
- **High confidence:** 10 to 20 flags with false positives flags filtered out
- **Medium confidence:** 4 to 9 flags with false positives flags filtered out
- **Low confidence:** 1 to 3 flags with false positives flags filtered out
- **Clean:** no flags at all



This chart shows that **36.4%** flagged samples with confidence of Medium or higher have no false positive flags.

We have looked at all of the results from all the various engines and our samples were ranked with 486 distinct categories.
These are examples of flags we found in our analyzed samples, the first list notes some of the most aggressive ones from our list.

- **Trojan.Win64.Tnega:** The "Tnega" family is essentially a malicious Swiss Army knife. Once it tricks its way onto a machine, it can perform a variety of commands dictated by a remote hacker. This includes downloading ransomware, turning the PC into a spam bot, or silently logging keystrokes to steal bank passwords.
- **Trojan:Win32/Kepavll!rfn:** The `!rfn` part stands for "reputation or behavior-based detection." Instead of looking for a known file signature, the antivirus caught this program acting suspiciously (like trying to secretly inject code into legitimate Windows processes). Its primary job is to act as an inside man, opening a backdoor to download much heavier malware later.
- **Trojan:Unknow/Wacatac.B9nj:** Wacatac is a broad catch-all name Microsoft Defender uses when a file exhibits classic, dangerous malware behavior (such as trying to modify critical system boot files or connecting to known malicious web domains). It is highly aggressive and usually signals a direct threat to system stability.

This second list contains some of the flags which are likely false positives and have been filtered out from our analysis.

- **PUA.Win64.AdobePatch.A:** This is explicitly a "hacktool" designed to bypass Adobe's licensing verification. Because it forcibly alters the code of another program to make it free, security companies flag it as a threat. While it violates copyright terms, the tool itself is doing what the user downloaded it to do.
- **Hack.Win32.Patcher.cl:** A "patcher" modifies the compiled binary code of an existing file on the computer. To an antivirus, a program editing another program's code looks exactly like a virus injecting payload.
- **exe.hacktool.crack:** These files are designed to inject code into another program's memory or permanently modify its `.exe` file. For example, a crack might replace the code that checks *"Did the user log in?"* with a line that always says *"Yes, log them in."*

Activator.exe						
d759b44ef4b0e86c75b73383138d578029997f5458f287049f6c3e7d8b5852b9						archive
En: **5.09148**	VT: **46/70**	Com: **2/0**	Shell: **Yes**	Net: **Yes**	Crypto: **Yes**	Reg: **Yes**

This sample was one of the highest rated in our dataset, it scored 46 out of 70 and only 5 were rated as hacktools. For this sample VirusTotal offers results for many sandboxed environments and we can see it employs techniques such as Software packing[3], Sandbox evasion[4] or File obfuscation[5], techniques often used by malware. We have strong reasons to believe that even if some of the antivirus engines return false positives and the file is from a verified uploader a simple activator should not be receiving a community score of **-12**, do **200+** network requests and drop **95** files.

[3] Software packing compresses or encrypts an entire executable file, wrapping it inside a new, uncompressed "wrapper" script. When the program runs, the wrapper unpacks the original code directly into the computer's memory, bypassing static antivirus scanners that only scan files sitting on the hard drive.

[4] Sandbox evasion is a collection of tricks code uses to detect if it is being watched (like checking for human-like mouse movements, specific registry keys, or time delays)—if it realizes it's in a sandbox, it plays dead and acts harmless.

[5] File obfuscation is the practice of making source code or binaries completely unreadable to humans and security analysts without changing how the program actually functions. It achieves this by scrambling variable names, adding useless "junk" code, or altering execution paths, turning clear logic into an undecipherable maze.

Conclusion

Our study highlights the severe risks of downloading pirated content, revealing that staggering **42.1%** of the tested samples were potentially malicious. While **11.7%** were rated as with the highest of confidence. The lower confidence samples were purely rated as such because they received lower ratings overall. **36.4%** of samples with confidence Medium or higher had no false positives flags, meaning it is very likely they were packing an actual malware and **58.3%** of those were in the Highest confidence category.

The study also demonstrated the limitations of relying solely on traditional desktop antivirus software when interacting with pirated content. Software cracks and activators frequently employ techniques similar to malware behavior, which contributes to false positives and conditions users to ignore security warnings entirely. Threat actors exploit this expectation by embedding genuine malware inside tools that users already anticipate will trigger antivirus alerts.

To efficiently process a large volume of samples, the research relied on static analysis techniques, including SHA-256 hashing, entropy measurement, heuristic string extraction, and VirusTotal aggregation. While dynamic analysis and reverse engineering can provide deeper insight into individual payloads, the selected methodology allowed scalable identification of broad distribution trends and common indicators of compromise across the BitTorrent ecosystem.

Ultimately, this study demonstrates that downloading executable content from public torrent networks presents a substantial security risk. **Even when files appear legitimate or originate from reputable uploaders, the combination of obfuscation techniques, multi-engine detections, and suspicious behavioral indicators demonstrates that pirated software ecosystems continue to be an effective vector for malware distribution. Furthermore, checking torrents via the tracker community does not reliably filter out malware.**

Works Cited

bitsight. “New Research Reveals 43 Percent of BitTorrent Applications on Corporate Networks Contain Malicious Software.” *bitsight*, 2015, https://www.bitsight.com/press-releases/bitsight-announces-file-sharing-risk-analysis-module. Accessed 26 5 2026.

Choo, Euijin, et al. “A Large Scale Study and Classification of VirusTotal Reports on Phishing and Malware URLs.” *arXiv*, 2022, https://arxiv.org/pdf/2205.13155. Accessed 26 5 2026.

Huntress. “What Is a False Positive Virus?” *huntress*, 2025, https://www.huntress.com/cybersecurity-101/topic/false-positive-viruses. Accessed 26 5 2026.

Kryczka, Michal, et al. “TorrentGuard: stopping scam and malware distribution in the BitTorrent ecosystem.” *arXiv*, 2012, https://arxiv.org/pdf/1105.3671. Accessed 26 5 2026.

resec. “The Antivirus Multi-Scan Tradeoff.” *resec*, 2022, https://resec.co/the-antivirus-multi-scan-tradeoff. Accessed 26 5 2026.

TorrentFreak. “Torrent Sites Ban Popular Uploader ‘CracksNow’ for Sharing Ransomware.” *TorrentFreak*, 2019, https://torrentfreak.com/torrent-sites-ban-popular-uploader-cracksnow-for-sharing-ransomware-190217/. Accessed 18 5 2026.