Malware Definition

Malware, or “malicious software,” is an umbrella term that describes any malicious program or code that is harmful to systems.

Hostile, intrusive, and intentionally nasty, malware seeks to invade, damage, or disable computers, computer systems, networks, tablets, and mobile devices, often by taking partial control over a device’s operations. Like the human flu, it interferes with normal functioning.

The motives behind malware vary. Malware can be about making money off you, sabotaging your ability to get work done, making a political statement, or just bragging rights. Although malware cannot damage the physical hardware of systems or network equipment (with one known exception—see the Google Android section below), it can steal, encrypt, or delete your data, alter or hijack core computer functions, and spy on your computer activity without your knowledge or permission.

Malware Types:

- Adware is unwanted software designed to throw advertisements up on your screen, most often within a web browser. Typically, it uses an underhanded method to either disguise itself as legitimate, or piggyback on another program to trick you into installing it on your PC, tablet, or mobile device.
- Spyware is malware that secretly observes the computer user’s activities without permission and reports it to the software’s author.
- A virus is malware that attaches to another program and, when executed—usually inadvertently by the user—replicates itself by modifying other computer programs and infecting them with its own bits of code.

- Worms are a type of malware similar to viruses. Like viruses, worms are self-replic
- ating. The big difference is that worms can spread across systems on their own, whereas viruses need some sort of action from a user in order to initiate the infection.
- A Trojan, or Trojan horse, is one of the most dangerous malware types. It usually represents itself as something useful in order to trick you. Once it's on your system, the attackers behind the Trojan gain unauthorized access to the affected computer. From there, Trojans can be used to steal financial information or install other forms of malware, often ransomware.
- Ransomware is a form of malware that locks you out of your device and/or encrypts your files, then forces you to pay a ransom to regain access. Ransomware has been called the cybercriminal's weapon of choice because it demands a quick, profitable payment in hard-to-trace cryptocurrency. The code behind ransomware is easy to obtain through online criminal marketplaces and defending against it is very difficult. While ransomware attacks on individual consumers are down at the moment, attacks on businesses are up 365 percent for 2019. As an example, the Ryuk ransomware specifically targets high-profile organizations that are more likely to pay out large ransoms.
- Rootkit is a form of malware that provides the attacker with administrator privileges on the infected system, also known as "root" access. Typically, it is also designed to stay hidden from the user, other software on the system, and the operating system itself.
- A keylogger is malware that records all the user's keystrokes on the keyboard, typically storing the gathered information and sending it to the

attacker, who is seeking sensitive information like usernames, passwords, or credit card details.

- Malicious cryptomining, also sometimes called drive-by mining or cryptojacking, is an increasingly prevalent malware usually installed by a Trojan. It allows someone else to use your computer to mine cryptocurrency like Bitcoin or Monero. So instead of letting you cash in on your own computer's horsepower, the cryptominers send the collected coins into their own account and not yours. Essentially, a malicious cryptominer is stealing your resources to make money.
- Exploits are a type of malware that takes advantage of bugs and vulnerabilities in a system in order to give the attacker access to your system. While there, the attacker might steal your data or drop some form of malware. A zero-day exploit refers to a software vulnerability for which there is currently no available defense or fix.

Malware Analysis Definition

Malware analysis is the study of the unique features, objectives, sources, and potential effects of harmful software and code, such as spyware, viruses, malvertising, and ransomware. It analyzes malware code to understand how it varies from other kinds.

Below is a malware analysis guide to help you better understand this unique cybersecurity methodology.

Benefits of Malware Analysis

Malware analysis provides several significant benefits. For example, it enables organizations to perform the following malware analysis steps:

1. Figure out how much damage an intrusion caused
2. Identify who may have installed malware inside the system
3. Determine the attack's level of sophistication
4. Pinpoint the exact vulnerability the malware exploited to access your system

Types of Malware Analysis

There are several types of malware analysis. You can use one or a combination before or after an attack, depending on the situation your organization faces.

Static Malware Analysis

Static malware analysis looks for files that may harm your system without actively running the malware code, making it a safe tool for exposing malicious libraries or packaged files. Static malware analysis can uncover clues regarding the nature of the malware, such as filenames, hashes, IP addresses, domains, and file header data. The malware can be observed using a variety of tools, such as network analyzers.

Dynamic Malware Analysis

Dynamic malware analysis uses a sandbox, which is a secure, isolated, virtual environment where you can run suspected dangerous code. Security professionals can closely monitor the malware in the sandbox without worrying about infecting the rest of the system or network, allowing them to gather more information about the malware.

Hybrid Malware Analysis

Hybrid malware analysis combines both static and dynamic techniques. For example, if malicious code makes changes to a computer's memory, dynamic analysis can detect that activity. Then, static analysis can determine exactly what changes were made.

4 Stages of Malware Analysis

The four steps of malware analysis are:

1. Static properties analysis
2. Interactive behavior analysis
3. Fully automated analysis
4. Manual code reversing

You can break down the malware analysis process into four stages:

Static Properties Analysis

Static properties refer to strings of code embedded inside the malware file, hashes, header details, and metadata. Static properties analysis provides a quick and easy way to gather helpful information about malware because the malware does not have to be executed for you to study it.

Interactive Behavior Analysis

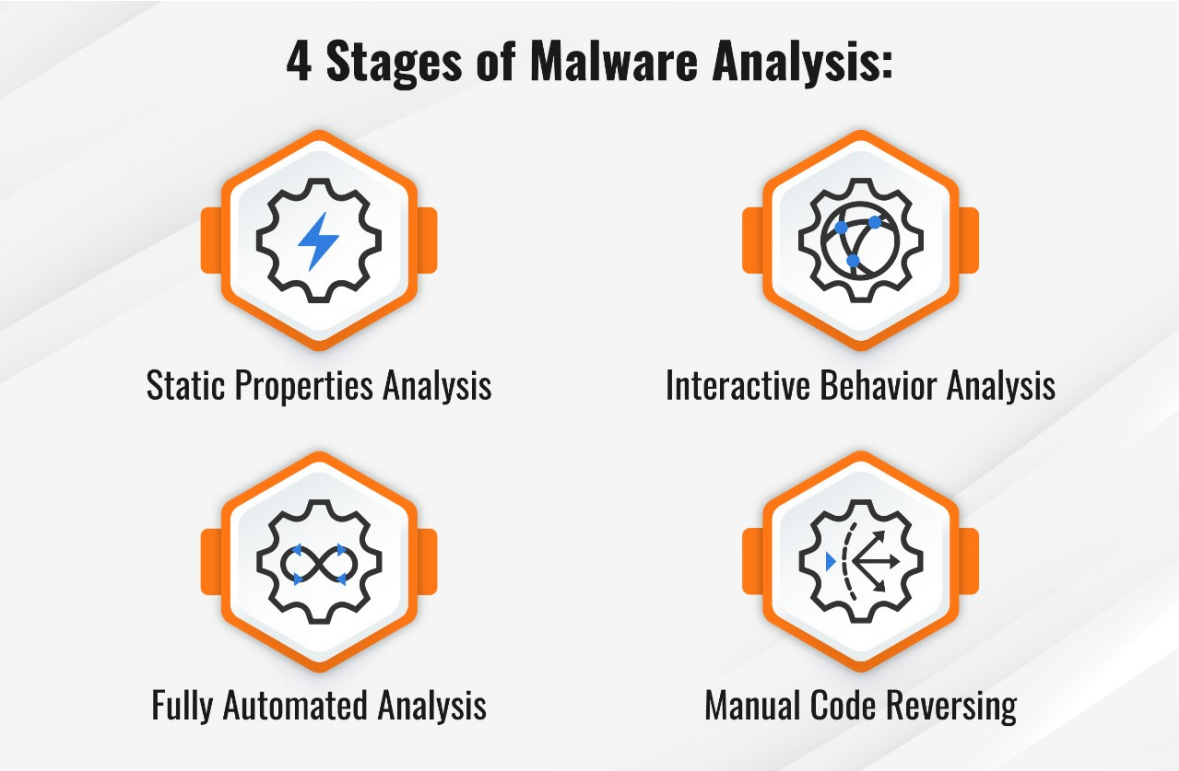
Interactive behavior analysis involves a security analyst interacting with malware running in a lab, making observations regarding its behavior. In this way, you can better understand how malware uses different elements of a computer system, such as its memory.

Fully Automated Analysis

Fully automated analysis scans suspected malware files using automated tools, focusing on what the malware can do once inside your system. After the analysis, you get a report outlining the potential damage to assets connected to your network.

Manual Code Reversing

Manual code reversing breaks down the code used to build the malware to learn how it works and what it is capable of doing. This is a time-consuming process that requires significant skill. However, when used correctly, manual code reversing can reveal valuable information about the malware.



Malware Analysis Use Cases

Malware analysis can be used in a variety of cybersecurity situations, such as:

Incident Response

For remediation and recovery to be successful, incident response teams must move quickly, and this is where malware analysis is especially useful. By giving incident responders applicable information for ongoing and upcoming incidents, malware analysis enables them to contain and prevent attacks.

Malware Research and Detection

To best safeguard your organization, identifying malicious code and understanding how it differs from benevolent code is extremely important. For example, by knowing which sites transmit malicious code, you can blacklist websites that propagate threats.

Indicator of Compromise (IOC) Extraction

With malware analysis, you can extract indicators of compromise (IOCs) to better understand how malware can attack your system. An IOC is data indicating that a system breach or attack has occurred. You can use this data to understand how your system reacts to attacks, making it easier to detect attacks in the future.

Threat Hunting

Threat hunters use malware analysis to identify previously unknown cyberthreats. For example, if you set up a honey trap, which is designed to attract malware and confine it to a homeless area of your network, you can

study how the malware behaves and potentially discover a new threat. Using malware analysis in this way may reveal threats that can get past your defenses.

Threat Alerts and Triage

Malware analysis enables IT teams to better understand how threats work and then use this information to react faster. The right malware analysis tool can send you alerts, prioritizing them according to severity. This way, instead of wasting time tracking down false positives, your security team can focus their energies on the threats that really matter.

Tools for Malware Analysis

Several malware analysis tools are available on the market, and here are some of the most well-known:

Process Hacker

Process Hacker enables analysts to understand the processes that are running on any given device on the network. This can be very useful as you allow malware to execute because you can watch the processes it impacts. With this information, you can determine how different computers react when malware is introduced to your system.

Fiddler

Fiddler can observe and study malicious traffic because it serves as a proxy, accepting and managing network traffic. Running Fiddler enables malware analysts to study the code and locate the hardcoded malicious sites that will be used to download the malware.

Limon

Limon is a controlled sandbox environment for studying malware that attacks Linux systems, enabling IT teams to monitor how the malware behaves and determine what it was designed to do.

PeStudio

PeStudio identifies potentially suspicious files by analyzing what is happening on your system. After it identifies malicious files, it quarantines them and assigns each a hash. You can then use each hash to access the malware and run it in a safe environment to learn how it behaves.

Ghidra

Ghidra disassembles malware instead of merely identifying it. It then takes whatever it finds in the malware code and translates it into something a human can read. In this way, it shows you what the malware designer might have been thinking while writing the malicious code.

Cuckoo Sandbox

Cuckoo Sandbox studies malware in a safe sandbox environment, recording its activity and then generating a report. This provides IT teams with data outlining how the malware attempts to impact your system.

CrowdStrike Falcon Insight

CrowdStrike Falcon automatically analyzes malware by combining CrowdStrike's threat intelligence with a sandbox environment. By comparing the malware's behavior in the sandbox to information from CrowdStrike's

threat intelligence, Falcon Insight can determine whether the malware already exists or is new to the threat landscape.

Frequently Asked Questions about Malware Analysis

How can malware be analyzed?

Malware can be analyzed using three different methods: static malware analysis, dynamic malware analysis, and hybrid malware analysis

What is the goal of malware analysis?

The goal of malware analysis is to better understand how malware operates so you can use that information to detect and stop threats.

Why is malware analysis important?

Malware analysis provides several significant benefits. For example, it enables organizations to:

1. Figure out how much damage an intrusion caused
2. Identify who may have installed malware inside the system
3. Determine the attack's level of sophistication
4. Pinpoint the exact vulnerability the malware exploited to access your system