Technologies used in the Antimalware Engine

Award-winning antimalware technologies

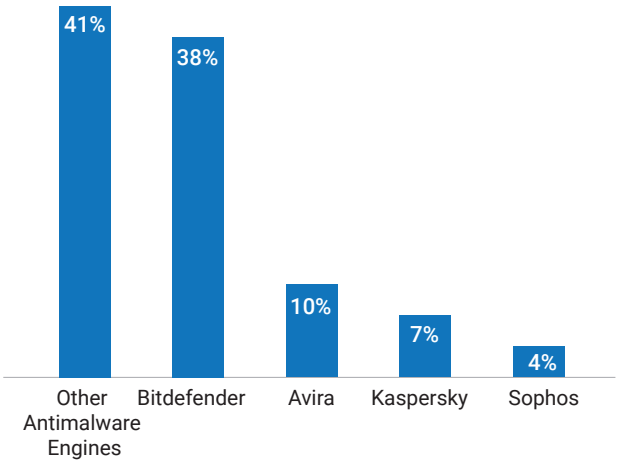
In the current cybersecurity landscape, threat actors are always probing and constantly switching tactics, making companies susceptible to malware incidents and outbreaks, business disruption and data breaches. Bitdefender's award-winning antimalware engine protects against the full range of cyber threats. It implements multiple technologies and detection methods to ensure industry leading detection accuracy and performance for known and unknown threats that leverage zero-day scenarios.

The Antimalware engine is available for integration in multiple SDK "flavors" designed to meet the specific requirements of our partners, and can be easily implemented at the endpoint, network, perimeter, gateway, and on cloud-based platforms. It can also be complemented by a wide range of other Bitdefender technologies in order to harden security, protect against additional threat vectors, or respond to additional market demand.

- Protection against known and unknown malware, including viruses, Trojans, worms, ransomware, advanced persistent threats, spyware, adware, etc.
- 99.9% detection consistently demonstrated in 3rd party tests
- High speed scanning, full multi-threading architecture
- Small footprint, low memory and processing requirements
- Fast and easy integration into partner applications and services
- Multiple SDKs available, optimized for various use cases
- Support for multiple OSes (Windows, Linux, Mac)
- Used by the Bitdefender consumer and business products
- Multiple awards from independent test organizations such as AV-Comparatives, AV-Test

Bitdefender's antimalware engine is used by market-leading network, endpoint and cloud security vendors. 11 of the 30 security solution vendors participating in AV-Comparatives' tests integrate the Bitdefender antimalware technologies. This testifies to the advanced detection and protection capabilities of the Bitdefender engine.

AV Comparatives - Third-party Antimalware Engine Providers



Key engine components and technologies

File format analyzers and parsers

- The Bitdefender antimalware engine supports a large number of file formats, from various executables to Microsoft Office documents, from Flash files and MP4 videos, etc. For each file format there is an analyzer in the engine, which says whether the format can be handled. There is also a parser which is responsible for extracting the relevant data for the engine.
- Unlike regular file format parsers, the Bitdefender engine only needs to extract the data that is relevant for malware detection; for example, for PDF files or Microsoft Office documents, only the part which contains scripts needs to be extracted.
- File format parsers must correctly deal with damaged or altered file formats, including handling the new, unsupported versions of the file format. Parsers are carefully designed so they attempt to extract the necessary data even from slightly damaged files (because some applications would open those files). They are also designed to handle completely damaged files, which would likely crash the original application.

Un-archivers and archivers

- The Bitdefender engine scans inside the most common type of archives and packed files.
- The engine can unpack a large number of archive formats, including all modern archivers and a number of old archives which are no longer supported. The latter are still necessary as many third-party test organizations and individual researchers have test collection malware using these older archives.

- In engine terminology, an archive is a file which can contain one or more other files. This definition includes not only traditional archives such as ZIP or RAR, but also disk images like ISO or DMG, install packages such as InstallShield or MSI, and even multipart MIME documents.
- As with file parsers, the Bitdefender engine can also handle damaged archives, and will try its best to unpack the file even from a damaged archive, so that it can be scanned. The Bitdefender antimalware archive algorithms are built around the concept of "in-depth scanning", which means they can be configured to scan embedded archives down to any depth. The engine can extract an archive embedded in another archive (embedded in another archive...) as long as there is enough disk/memory to store the unpacked objects.
- The Bitdefender engine can also pack (i.e. create) a number of archives, as this is necessary for the disinfection or deletion of files inside the archives.

Executable unpackers

Runtime packers are used to minimize the download size of an executable file, but in many cases, they are also employed to add an obfuscation layer over the program code. Once initiated, the executable runs a decoder loop that opens the packed section and then transfers execution (in memory) to the unpacked section. Seeing a runtime-packer in use is not enough evidence to classify a program as malicious, because benign legitimate applications also use packers. Nonetheless, the presence of a packer can raise a yellow flag in an anti-malware engine. When other suspicious behaviors are encountered, the combination results in a heuristic classification as malware.

As most malware binaries are packed, this is an extremely important feature of the antimalware engine.

- The Bitdefender engine can unpack a large variety of packed executables by using either emulation or generic unpackers.
- The engine will try to use generic unpackers for the packers it recognizes (such as UPX). For the packers it does not recognize or for which it has no generic unpacker, it will use emulation. This is extremely useful as it means that if malware writers start to use a new packer, technology licensing partners do not need to wait until Bitdefender creates an unpacker for it.

Emulation

Emulation is the antimalware engine's ultimate weapon against polymorphic malware and for accurate proactive detections in general. Polymorphic malware changes the encryption method and key, with which it encrypts the original virus code, with every replication step, but the original virus code/body stays the same. Bypassing any encryption and peeking under the surface allows our generic detection to spot any polymorphic malware family.

Emulation means the engine simulates a virtual computer – its CPU, memory, operating system API and resources—and simulates execution of a suspect file in this virtual environment. The suspect file's code is disassembled, just as an operating system would do it, but the instructions have effect only on software-simulated, safe data structures. This disassembly helps us gain a high level of insight into the suspect file, allowing for an accurate threat prediction.

An emulator has to cope with the many features of the simulated computer and operating system, such as processes and threads, files, and anti-debugging and anti-emulation techniques.

The Bitdefender antimalware engine uses emulation to deal with obfuscated binaries.

- It checks files by running them in a virtual environment inside the Bitdefender engine, designed to emulate the behavior of an actual computer. If any specific file exhibits suspicious, malware-like activity, the engine reports the file as malicious. If not, the file is declared clean and the process is allowed to run.
- The output of the emulation also powers malware detections, notably the heuristics-based detection.
- The engine has a full machine code emulator for x86/x64 platforms, as well as a full emulator for JavaScript and VBScript languages. These emulators are confined and have no access to the actual file system or Internet, and thus will not leak the information.

Heuristics-based detection

- Heuristic detection is responsible for the detection of previously unknown zero-day malware. It generally provides outstanding proactive detection capabilities against new malware variants, new malware families and against unknown vulnerabilities and exploits.
- Heuristics are a form of proactive detection that closes the window during which computers are vulnerable. Rather than relying on signatures or binary or code fingerprints, heuristic detection relies on complex algorithms that specify actual patterns and behaviors, which may indicate that an application is malicious. This works because malicious programs inevitably attempt to

perform actions in a context that legitimate applications do not. Examples of suspicious behavior would include attempting to drop files or disguise processes injecting or executing code in another process's memory space. Because heuristic detection looks for behavioral characteristics rather than relying on simple pattern-matching, they are able to detect and block new and emerging threats for which a signature or fingerprint has yet to be released.

- The majority of heuristic detections, including the Bitdefender B-HAVE technology – Bitdefender terminology for heuristic-based detection – temporarily delay applications from starting while the code is executed in a virtual environment that is completely isolated – or sandboxed - from the real computer. If no suspicious behavior is observed, the computer is instructed to start the application normally. If suspicious behavior is observed, the program execution is blocked. The entire process happens in milliseconds and so has practically no impact on either the user experience or perceived performance. In other words, heuristic detection is the logic that analyzes the output of generic signatures (static) and emulator (dynamic) code analysis when it is not clear whether the binary is malware. Files are first checked against the Bitdefender Signature Database (a database of malware "fingerprints") that is updated at an hourly rate. If the file's content matches one of the signatures, the product automatically tries to disinfect the threat. If this action fails, the file is moved into quarantine. If no signature is matched, the file is sent to B-HAVE/ the heuristic engine to be checked.
- Heuristic detection is thus based on the output of the emulator, and is effective in detecting malware which have not been previously seen. Retrospective tests by AV-Comparatives are based on exclusively using such detection.

Generic detection

Generic detection is useful for discovering all samples (including those unknown) of a known malware family.

It is a type of detection used by the antimalware engine for identifying files with malicious characteristics. Generic detection extracts the key characteristics of one or a few samples of a malware family or exploits, and creates a "one size fits all" detection rule that will catch as many variants of the same family or exploits for the same vulnerability, as possible.

- Generic detection is responsible for the detection of all strains of malware belonging to a specific malware family or subfamily.
- It is very useful in detecting polymorphic or metamorphic malware, as those are impossible to handle based on signatures only.
- Unlike single-file detections which can only identify unique files, generic detection can look for broadly similar code or behavioral patterns in dozens or even hundreds of suspect programs or files, to efficiently determine their potential for causing harm.

Signature-based detection

Conventional detection relies on signatures. Antimalware signatures are code snippets extracted from malware samples and used by antimalware programs to perform pattern-matching. The problem with this method is that it takes time to produce the signature: antimalware vendors need to obtain a sample of the malware, develop a signature, and then push that signature to users – and this leads to the creation of a vulnerability window.

Signature-based detection is also used by the Bitdefender engine for detecting all known malware samples. It uses key aspects of an examined file to create a static fingerprint of known malware. A major limitation of this approach is that, by itself, this method is unable to flag malicious files for which signatures are not yet available. Having this in mind, cyber criminals frequently mutate their creations to retain malicious functionality by changing the file's signature. However, despite the bad reputation that signatures may sometimes receive, they are still very useful for detection of specific, non-mutating threads. As used by Bitdefender, the "signature" is a combination of small hashes on certain parts of malware code (it is not a full hash on the file), and thus it is capable to detect even completely mutated files, as long as the actual malware code part remains unmodified.

There are several reasons why signatures are useful:

- A new signature can be created and added to the engine very quickly – in a matter of minutes. A generic detection would require a malware researcher to study the malware family, write the code, and test it – a process which takes at least a few hours. By providing the signature, users will be protected much faster.
- Applying signatures is a faster process compared to running the detection code. Applying 10,000 signatures is significantly faster than applying 100 generic detections, which results in increased performance for users.
- Signatures are compact; adding a large set of them will only increase the engine size insignificantly, thus keeping the total database size under control.
- As they are based on malware content, signatures are very precise and accurate in terms of detection. They almost never produce false positives, which results in a better-quality security product.

In conclusion, while Bitdefender is truly a next-generation engine and relies on a large number of technologies for malware detection, signatures are and will continue to remain a part of the engine, and will produce a solid bulk of detections.

Machine-learning algorithms

Machine-learning algorithms significantly improve detection time for modern threats, as they can analyze large amounts of data significantly faster than any human would. When trained to accurately detect various types of malware behavior, machine-learning algorithms yield a high detection rate, both for known and unknown samples. Incorporating machine learning into both static (file-based) and dynamic (behavior-based) malware analysis significantly accelerates reactions against new malware samples, offering protection even from previously unknown threats – APTs, zero-day attacks, and ransomware. Bitdefender has been training algorithms for years - Perceptrons, Neural Networks, Centroids, Binary Decision Tree and Deep Learning, to name a few. Some are specialized on specific malware families, some on new malicious files, and some are built to minimize the number of false positives. They complement each other, as well as traditional heuristic and signature-based detection. For instance, Neural Networks are some of the most popular implementations of machine learning algorithms that are designed to increase malware detection rates using repeated training sessions on popular malware categories. Allowing these algorithms to extract features from existing malware samples or families enables them to learn to predict future malware based on shared similar features.

Cloud-based detection

Bitdefender's antimalware engine combines powerful local filters with cloud-based updates powered by the Bitdefender security cloud, offers access to Bitdefender's global antimalware intelligence. Powered by the Bitdefender Global Protective Network (GPN), the engine uses a broad, cloud database. The database is fuelled by the 500 million threat sensors deployed through a global network of datacenters. Cloud-based detection offers up-to-the-minute protection and visibility across the global threat landscape and provide global users with real-time protection as well as an additional layer of mitigation against false positives.

Performance

Based on the users' experience with anti-malware products, especially the ones preloaded on a machine newly acquired, it shows that they slow down a system significantly. However, this is not true for all antimalware vendors. The Bitdefender detection logic has been designed with real-life usage scenarios in mind, where ensuring a satisfactory user-experience is just as important as protection. In addition, optional modules such as SmartScan implement intelligent optimization algorithms that speed up the scanning process without compromising on the malware detection rate, maintaining a high effectiveness against threats in a performance-effective manner. As you see in this graph, the drop in performance with Bitdefender is kept under 10% - with all security options enabled.

About Bitdefender Technology Licensing. Bitdefender provides end-to-end cyber security solutions and advanced threat protection to more than 500 million users in more than 150 countries. Since 2001, Bitdefender has consistently produced award-winning business and consumer security technology, and has become a provider of choice for leading Independent Software Vendors (ISVs), hardware vendors, service providers and enterprise organizations looking to integrate security technologies into their products and services. Today, Bitdefender has over 150 technology partners worldwide. More information is available at www.bitdefender.com/oem