P

Polymorphic Malware: How changing code evades simple signatures

What it is Polymorphic malware changes its observable code or encrypted representation as it spreads or runs while preserving the same core behavior. The technique can be used by viruses, worms, Trojans, ransomware, and other malware...

PUBLISHER
Gridinsoft Support

UPDATED
Jul 17, 2026

READ TIME
2 min

What it is

Polymorphic malware changes its observable code or encrypted representation as it spreads or runs while preserving the same core behavior. The technique can be used by viruses, worms, Trojans, ransomware, and other malware. Its purpose is to make simple hash and byte-pattern signatures less reliable.

How it works

A typical polymorphic engine encrypts the main payload with a different key and varies the small decryptor or wrapper for each generation. The resulting files look different at the binary level but unpack to related functionality. Metamorphic malware goes further by rewriting the body of its

code rather than mainly changing encryption and packaging.

Key points

- Polymorphism does not make malware invisible; behavior, unpacked memory, relationships, and network activity can still reveal it.
- A changing hash means defenders need family-level and behavior-based detection rather than endless blocklists.
- Packers are also used by legitimate software, so packing alone is not proof of malicious intent.

Defensive approach

- Use layered detection that combines reputation, emulation, memory analysis, behavior, and network telemetry.
- Investigate the parent process and delivery chain, not only the final changing file.
- Restrict script and macro execution and prevent users from launching untrusted downloads.
- Keep endpoint engines updated so new unpacking and family logic is available.

Further reading

For a longer technical profile, see the Gridinsoft comparison of polymorphic and metamorphic malware.