Microsoft Security Servicing Commitments

Our commitment to protecting customers from vulnerabilities in our products, services, and devices includes providing security updates that address these vulnerabilities when they are discovered. We also want to ensure we are transparent with our customers in our approach. This document helps to describe the criteria the Microsoft Security Response Center (MSRC) uses to determine whether a reported vulnerability will be addressed through servicing, or in the next version of a product. For vulnerabilities in products, this servicing takes the form of a security update, most commonly released as security updates on Update Tuesday. The purpose of this document is to clarify the commitments as they pertain to Windows.

Security Servicing Criteria

The criteria used by Microsoft when evaluating whether or not to provide a security update for a reported vulnerability involves answering two key questions:

1. Does the vulnerability violate a promise made by a security boundary or a security feature that Microsoft has committed to defending?

2. Does the severity of the vulnerability meet the bar for servicing?

If the answer to both questions is yes, then the vulnerability will be addressed through a security update that applies to all affected and supported offerings. If the answer to either question is no, then by default the vulnerability will be considered for the next version or release of an offering but will not be addressed through a security update, though in some cases an exception may be made.

Security boundaries and features with servicing commitments

Microsoft's products, services, and devices rely on promises made by a number of security boundaries and security features in order to achieve our security goals.

Security boundaries

A security boundary provides a logical separation between the code and data of security domains with different levels of trust. For example, the separation between kernel mode and user mode is a classic and straightforward security boundary. Microsoft software depends on multiple security boundaries in order to isolate devices on the network, virtual machines, and applications on a device. The following table summarizes the security boundaries that Microsoft has defined.

Security boundary	Security promise	Servicing commitment	Bug Bounty
Network boundary	An unauthorized network endpoint cannot access or tamper with the code and data on a customer's device.	Yes	Yes
Kernel boundary	A non-administrative user mode process cannot access or tamper with kernel code and data.	Yes	Yes

Process boundary	An unauthorized user mode process cannot access or tamper with the code and data of another process.	Yes	Yes
AppContainer sandbox boundary	An AppContainer-based sandbox process cannot access or tamper with code and data outside of the sandbox based on the container capabilities	Yes	Yes
Session boundary	A user logon session cannot access or tamper with another user logon session without being authorized.	Yes	Yes
Web browser boundary	An unauthorized website cannot violate the same-origin policy, nor can it access or tamper with the native code and data of the Microsoft Edge web browser sandbox.	Yes	Yes
Virtual machine boundary	An unauthorized Hyper-V guest virtual machine cannot access or tamper with the code and data of another guest virtual machine.	Yes	Yes
Virtual Secure Mode boundary	Data and code within a VSM trustlet or enclave cannot be accessed or tampered with by code executing outside of the VSM trustlet or enclave.	Yes	Yes

Security features

A security feature provides protection against one or more threats. In some cases, a security feature may make a promise related to the threat they are protecting against and there are not expected to be any by design limitations that prohibit delivering on that promise. The following table summarizes the security features that Microsoft has defined that make a promise that has a servicing commitment.

Category	Security feature	Security promise	Servicing commitment	Bug Bounty
Device security	BitLocker	Data that is encrypted on disk cannot be obtained when the device is turned off	Yes	Yes
	Secure Boot	Only authorized code can run in the pre-OS, including OS loaders, as defined by the UEFI firmware policy.	Yes	Yes
Platform security	Windows Defender System Guard (WDSG)	Improperly signed binaries cannot execute or load in accordance with the Application Control policy for the system. Bypasses leveraging applications which are permitted by the policy are not in scope.	Yes	Yes

Application security	Windows Defender Application Control (WDAC)	Only executable code, including scripts run by enlightened Windows script hosts, that conforms to the device's policy can run. Bypasses leveraging applications which are permitted by the policy are not in scope.	Yes	Yes
Identity and access control	Windows Hello / Biometrics	An attacker cannot spoof, phish, or breach NGC credentials to impersonate a user.	Yes	Yes
	Windows Resource Access Control	An identity (user, group) cannot access or tamper with a resource (file, named pipe, etc.) unless explicitly authorized to do so	Yes	Yes
Cryptography API: Next Generation (CNG)	Platform Cryptography	Algorithms are implemented to specification (e.g. NIST) and do not leak sensitive data.	Yes	Yes
Health attestation	Host Guardian Service (HGS)	Assess the identity and health of a caller issuing or withholding health claims necessary for downstream cryptographic operations.	Yes	Yes
Authentication Protocols	Authentication Protocols	Protocols are implemented to specification and an attacker cannot tamper with, reveal sensitive data, or impersonate users gaining elevated privileges.	Yes	Yes

Defense-in-depth security features

In some cases, a security feature may provide protection against a threat without making a promise. These security features are typically referred to as defense-in-depth features or mitigations because they provide additional security but may have by design limitations that prevent them from making a promise. A bypass for a defense-in-depth security feature does not pose a direct risk because an attacker must also have found a vulnerability that affects a security boundary, or they must rely on social engineering to achieve the initial stage of a device compromise.

The following table summarizes the defense-in-depth security features that Microsoft has defined which do not have a servicing commitment. Any vulnerability or bypass that affects these security features will not be serviced by default, but it may be addressed in a future version or release. Many of these features are being continuously improved across each product release and are also covered by active bug bounty programs.

Category	Security feature	Security goal	Servicing commitment	Bug Bounty
User safety	User Account Control (UAC)	Prevent unwanted system-wide changes (files, registry, etc) without administrator consent	No	No
	AppLocker	Prevent unauthorized applications from executing	No	No
	Controlled Folder Access	Protect access and modification to controlled folders from apps that may be malicious	No	No
	Mark of the Web (MOTW)	Prevent active content download from the web from elevating privileges when viewed locally.	No	No
Exploit mitigations	Data Execution Prevention (DEP)	An attacker cannot execute code from non-executable memory such as heaps and stacks	No	Yes
	Address Space Layout Randomization (ASLR)	The layout of the process virtual address space is not predictable to an attacker	No	Yes
	Kernel Address Space Layout Randomization (KASLR)	The layout of the kernel virtual address space is not predictable to an attacker	No	No
	Arbitrary Code Guard (ACG)	An ACG-enabled process cannot modify code pages or allocate new private code pages	No	Yes
	Code Integrity Guard (CIG)	A CIG-enabled process cannot directly load an improperly signed executable image (DLL)	No	Yes
	Control Flow Guard (CFG)	CFG protected code can only make indirect calls to valid indirect call targets	No	Yes
	Child Process Restriction	A child process cannot be created when this restriction is enabled	No	Yes
	SafeSEH/SEHOP	The integrity of the exception handler chain cannot be subverted	No	Yes
	Heap randomization and metadata protection	The integrity of heap metadata cannot be subverted and the layout of heap allocations is not predictable to an attacker	No	Yes
	Windows Defender	Allow apps to enable additional defense-in-depth exploit	No	No

	Exploit Guard (WDEG)	mitigation features that make it more difficult to exploit vulnerabilities		
Platform lockdown	Protected Process Light (PPL)	Prevent non-administrative non-PPL processes from accessing or tampering with code and data in a PPL process via open process functions	No	No
	Shielded Virtual Machines	Helps to protect a VM's secrets and its data against malicious fabric admins or malware running on the host from both runtime and offline attacks	No	No

Severity of vulnerabilities

The second dimension that Microsoft uses to evaluate whether or not a reported vulnerability should be serviced is based on the severity of the vulnerability. The severity of a vulnerability is determined using a version of the SDL Bug Bar which maps the properties of the vulnerability (impact, scenario, etc.) to its severity – an example of the SDL Bug Bar can be found here. MSRC defines five severity levels: Critical, Important, Moderate, Low, and None. If a vulnerability is rated as Critical or Important, and the vulnerability applies to a security boundary or security feature that has a servicing commitment, then the vulnerability will be addressed through a security update. The following table provides a summarized definition of the criteria used to determine the severity for reported vulnerabilities.

Products

CRITICAL	**Remote Code Execution**: Any vulnerability which could allow an attacker to execute malicious code on a system *without user interaction.* Examples: • Code execution on Hyper-V host from Guest Virtual Machine • Chakra Remote Code Execution • SMB Remote Code Execution
IMPORTANT	**Elevation of Privilege**: A vulnerability which allows a low privileged user to bypass controls and operate as a higher privileged user. Examples: • Win32k Use-after-Free • App container escapes • Windows Defender Application Guard (WDAG) escapes • SOP Bypass vulnerabilities

	- Escalation from non-administrative user to SYSTEM privileges **Information Disclosure**: A vulnerability which allows an attacker to obtain access to data which should be protected during normal operation. Examples: - Uninitialized kernel memory disclosure to user mode - Kernel pool pointer disclosure **Remote Code Execution:** Any vulnerability which could allow execution of malicious code *but requires user interaction*. Examples: - Heap buffer overrun via a document file **Denial of Service**: A vulnerability which allows an attacker to disrupt the system and interrupt or halt normal operations. Examples: - Remotely triggerable resource exhaustion issues - Remotely triggerable issues that result in a reboot **Security Feature Bypass**: A vulnerability which allows an attacker to circumvent controls or features designed to protect users. Examples: - WDSG bypasses - BitLocker bypasses
MODERATE	**Denial of Service**: A vulnerability which allows an attacker to disrupt the system and interrupt or halt normal operations. **Examples:**
	- Locally triggerable with no remote vector - Remotely triggerable but requiring many attackers (DDoS)