Microsoft Vulnerability Severity Classification for Windows

Last Updated: May 13th, 2021

Summary:
The information listed in this bug bar is used by the Microsoft Security Response Center (MSRC) to triage bugs and determine bug severity in terms of security.

When a vulnerability in one class (e.g. EOP) can be combined with By-Design behavior to achieve higher class vulnerability (e.g. RCE), the vulnerability is rated at the higher class.

The ratings are derived from MSRC advisory rating classifications. Several views are provided into this information with a goal of making it easier to use, and each view contains the same core information. The bug bar describes different severities for *client software* (defined as software that runs locally on a single computer or software that accesses shared resources provided by a server over a network) and *server software* (Computers configured to run software that await and fulfill requests from client processes running on other computers).

Server – Severity Pivot

The server bar is usually not appropriate when user interaction is part of the exploitation process. If a Critical vulnerability exists only on our server products and is exploited in a way that requires user interaction and results in the compromise of the server, the severity may be reduced from Critical to Important in accordance with the definition of extensive user interaction presented at the start of the client severity pivot.

Critical	Summary: (Server) – “Network Worms or *unavoidable* cases where the server is compromised” • Elevation of Privilege – The ability to either execute arbitrary code OR obtain more privilege than authorized o Remote Anonymous User ▪ Examples • Unauthorized File System Access – Arbitrary writing to file system • Execution of Arbitrary code • SQL Injection (that allows code execution) • Exploitable memory corruption issuesin remote anonymously callable code o Guest virtual machine ▪ Examples

	• In a virtualized environment, a vulnerability allows the guest VM to cause arbitrary code execution in the host machine, effectively defeating the virtual machine boundary.
Important	Summary: (Server) – “Non-default critical scenarios or cases where mitigations exist that can help *prevent* critical scenarios.” • Denial of Service ○ Must be “easy to exploit” by sending a small quantity of data or executing another simple and reliable attack ○ Anonymous ▪ Persistent DoS • Example: sending a single malicious TCP packet results in a system crash • Example: sending a small number of packets that causes a service failure ▪ Temporary DoS with amplification • Example: sending a small number of packets that causes the system to be unusable for a period of time. Example: A web server (like IIS) being down for a minute or longer. • Example: *a single remote client* consuming all available resources (sessions, memory, etc.) on a server by establishing sessions and keeping them open ○ Authenticated ▪ Persistent DoS ***against a high value asset*** • Example: sending a small number of packets that causes a service failure for a ***high value asset*** in server roles (ex: Certificate Server, Kerberos server, Domain Controller). For example, when a domain authenticated user can DoS a Domain Controller. • Elevation of Privilege – The ability to either execute arbitrary code OR obtain more privilege than authorized ○ Remote Authenticated User ○ Local Authenticated User (Terminal Server) ○ Examples ▪ Unauthorized File System Access - Arbitrary writing to file system ▪ Execution of Arbitrary code ▪ Exploitable memory corruption issues in code that can be accessed by remote or local authenticated users that are not administrators. (Administrator scenarios do not have security concerns by definition, but are still reliability issues)

- Information Disclosure (Targeted)
    - Any cases where the attacker can bypass a security boundary to read information on the system that was not intended or designed to be exposed
    - Examples
        - Personally Identifiable Information (PII) disclosure.
            - Disclosure of PII (examples: email addresses, phone numbers, credit card info)
            - Attacker can collect PII without user consent or in a covert fashion
        - Unintentional read access to memory contents in kernel space from a non-administrative user mode process
        - In a virtualized environment, a vulnerability allows the guest VM to obtain current or previous memory contents of the host or another virtual machine, effectively defeating the virtual machine boundary.
- Spoofing
    - An entity (computer, server, user, process) is able to masquerade ***as a specific entity*** (user or computer) of his/her choice.
    - Examples
        - Web server uses client certificate authentication (SSL) improperly to allow an attacker to be identified as any user of his/her choice
        - New protocol is designed to provide remote client authentication, but flaw exists in the protocol that allows a malicious remote user to be seen as a different user of his/her choice
- Tampering
    - Modification of any "***high value asset***" data in ***a common or default scenario*** where the modification persists after restarting the affected software.
    - Permanent or persistent modification of any user or system data used ***in a common or default scenario.***
    - Examples
        - Modification of application data files or databases in a common or default scenario e.g. Authenticated SQL Injection
        - Proxy cache poisoning in a common or default scenario
        - Modification of OS or application settings without user consent in a common or default scenario
- Security features: Breaking or bypassing any security feature provided

	○ Examples ▪ Disabling or bypassing Windows Defender Application Guard without informing user or gaining consent ▪ Disabling or bypassing Secure Boot without informing user or gaining consent ▪ Windows Hello bypass ▪ Bitlocker bypass, ex: not encrypting part of the drive
Moderate	• Denial of Service ○ Anonymous ▪ Temporary DoS without amplification in a default/common install • Example: ***multiple remote clients*** consuming all available resources (sessions, memory, etc.) on a server by establishing sessions and keeping them open ○ Authenticated ▪ Persistent DoS • Example: logged in Exchange user can send a specific mail message and crash the Exchange Server, and the crash is **not** due to a Write AV, exploitable read AV, or integer overflow ▪ Temporary DoS with amplification in a default/common install • Example: an ordinary SQL Server user executes a stored procedure installed by some product and consumes 100% of the CPU for a few minutes • Information Disclosure (Targeted) ○ Cases where the attacker can easily read information on the system ***from specific locations***, including system information, that was not intended/designed to be exposed ○ Examples ▪ Targeted disclosure of anonymous data ▪ Targeted disclosure of the existence of a file ▪ Targeted disclosure of file version number • Spoofing ○ An entity (computer, server, user, process) is able to masquerade as a different, random entity that cannot be specifically selected. ○ Examples

	▪ Client properly authenticates to server, but server hands back a session from another random user who happens to be connected to the server at the same time. ▪ MS04-002 (HTTP/NTLM & Exchange) • Tampering o Permanent or persistent modification of any user or system data ***in a specific scenario.*** o Examples ▪ Modification of application data files or databases ***in a specific scenario*** ▪ Proxy cache poisoning ***in a specific scenario*** ▪ Modification of OS/application settings without user consent ***in a specific scenario*** o Temporary modification of data ***in a common or default scenario*** that does not persist after restarting the OS/application/session. • Security Assurances: A security assurance is either a security feature or another product feature/function that customers expect to offer security protection. Communications have messaged (explicitly or implicitly) that customers can rely on the integrity of the feature, and that's what makes it a security assurance. Security advisories may be released for a shortcoming in a security assurance that undermines the customer's reliance or trust. o Examples: ▪ Processes running with normal "user" privileges cannot gain "admin" privileges unless admin password/credentials have been provided via intentionally authorized methods. ▪ Internet-based JavaScript running in Microsoft Edge or Internet Explorer cannot control anything the host operating system unless the user has explicitly changed the default browser security settings.
Low	• Information Disclosure (Untargeted) o Runtime information ▪ Example: leak of non-sensitive memory • Tampering o Temporary modification of data ***in a specific scenario*** that does not persist after restarting the OS/application/session.

Client – Severity Pivot

Extensive user action is explained as follows:

- “User interaction” usually happens in client scenarios.
- Normal, simple user actions like clicking links, previewing mail (including attachments), viewing local folders or file shares, opening a file (without any warning dialog) are not “extensive user interaction.”
- Extensive: User manually navigating to particular web site (ex: typing in URL) or clicking through one or more decision dialogs
- **NOT** Extensive: User clicking through email links.

Clarification: Note that the effect of “extensive user interaction” is not “one level reduction in severity,” but is and has been “a reduction in severity in certain circumstances” where the phrase “extensive user interaction” appears in the bug bar. The intent is to help differentiate fast spreading and wormable from those where, because the user interacts, the attack is slowed down. This bug bar does not allow us to reduce Elevation of Privilege below important because of user interaction.

Critical	Summary: (Client) – “Network Worms, or *unavoidable* common browsing/use scenarios where client is compromised ***without*** warnings or prompts. • Elevation of Privilege (Remote) – The ability to either execute arbitrary code OR obtain more privilege than intended • Examples ○ Unauthorized File System Access – Writing to file system ○ Execution of Arbitrary code – ***without*** extensive user action ○ Exploitable memory corruption issues in remotely callable code (***without*** extensive user action) • Guest virtual machine ○ In a virtualized environment, a vulnerability allows the guest VM to cause arbitrary code execution in the host machine, effectively defeating the virtual machine boundary.
Important	Summary: (Client) – “Common browsing/use scenarios where client is compromised ***with*** warnings or prompts, or via extensive actions without prompts.” Note that this does not discriminate over the quality/usability of a prompt and likelihood a user might click through the prompt, but just that a prompt of some form exists. • Elevation of Privilege ○ (Remote)

        - Execution of Arbitrary code – ***with*** extensive user action
        - All Write AVs (Access Violations), all kernel-mode Read AVs (Access Violations), other exploitable read AVs, or integer overflows in ***remote*** callable code (***with*** extensive user action)
        - Windows Store and Mobile Applications
            - Execution of arbitrary code outside the restricted app container context without user interaction.
            - Use of capabilities without informing the user.
                - Example: Use of location capability without informing the user
                - Example: Use of SMS capability without informing the user.

    - (Local)
        - Local low privilege user can elevate his/her rights to those of another user, administrator, and/or local system
        - All Write AVs (Access Violations), all kernel-mode Read AVs (Access Violations), exploitable integer overflows and any crashes classified as Probably Exploitable or Exploitable by !Exploitable in ***local*** callable code
- Information Disclosure (Targeted)
    - Any cases where the attacker can bypass a security boundary to read information on the system that was not intended or designed to be exposed
    - Examples
        - Unauthorized File System Access - Reading from file system
        - Unintentional read access to memory contents in kernel space from a user mode process
        - In an environment where a client is connecting to a server, a web browser connecting to a webserver for example, a vulnerability allows an attacker to collect information that facilitates predicting addressing of the memory layout. In turn the attacker could use this information to deliver tailored exploits to bypass memory protection technologies such as DEP and ASLR for an additional RCE vulnerability.
        - In a virtualized environment, a vulnerability allows the guest VM to obtain current or previous memory contents of the host or another virtual machine, effectively defeating the virtual machine boundary.
        - Disclosure of Personally Identifiable Information (PII).
            - Disclosure of PII (example: email addresses, phone numbers)

- Denial of Service
  - System Corruption DoS that requires re-installation of the system and/or components
    - Example: Visiting a web page causes registry corruption that makes the machine un-bootable
  - Drive-by DoS
    - Criteria:
      - Un-authenticated System DoS
      - Default exposure
      - No user interaction
      - No Audit and punish trail
      - Example: Drive-by Bluetooth System DoS

- Spoofing
  - Ability for an attacker to present UI that is different from but visually identical to UI which users ***must rely on to make valid trust decisions*** in a ***default/common scenario***. A trust decision is defined as any time the user takes an action believing some information is being presented by a particular entity, either the system or some specific local or remote source.
  - Examples
    - Displaying a different URL in the browser's address bar from the URL of the site that browser is actually displaying in a ***default/common scenario***
    - Displaying a window over the browser's address bar that looks identical to an address bar but displays bogus data in a ***default/common scenario***
    - Displaying a different file name in a "Do you want to run this program?" dialog box than that of the file that will actually be loaded in a ***default/common scenario***
    - Display a "fake" login prompt to gather user or account credentials
- Tampering
  - Permanent or persistent modification of any user data or data used to make trust decisions in a common or default scenario
  - Examples
    - Web browser cache poisoning

	▪ Modification of significant OS/application settings without user consent ▪ Modification of user data ▪ Writing of arbitrary data outside of the app container context without user interaction • Security features: Breaking or bypassing any security feature provided ○ Examples ▪ Disabling or bypassing Windows Defender Application Guard without informing user or gaining consent ▪ Disabling or bypassing Secure Boot without informing user or gaining consent ▪ Windows Hello bypass ▪ Bitlocker bypass, ex: not encrypting part of the drive
Moderate	• Denial of Service ○ Permanent or persistent DoS – Requires cold reboot or causes system crash ▪ Example: Opening a Word document causes the machine to crash ▪ Example: Browsing the Internet causes machine to crash ▪ Example: Launching a Windows Store app causes machine to crash • Information Disclosure (Targeted) ○ Cases where the attacker can read information on the system ***from known locations***, including system information, that was not intended/designed to be exposed ○ Examples ▪ Targeted Existence of file ▪ Targeted File version number • Information Disclosure (Unencrypted connection) - Windows Store Applications ○ Case where the attacker can read information from the unencrypted connection ○ Examples

        - The application is revealing user’s personal information – email address, name and surname, insurance number, medical information, national identification or any other data that can be used to identify the user.
        - The application is revealing user’s data – GPS coordinates, translator search, search queries or any other data that can be used to identify user preferences.
        - The application is revealing internal IP addresses and the device data (ID, name or other).

- Information Disclosure (Third party) - Windows Store Applications
    - Case where the information is sent to the third-party server
    - Examples
        - The application is sending trackable information such as: user’s email address, user’s GPS coordinates, device data (ID, name or other) or internal IP.

- Spoofing
    - Ability for attacker to present UI that is different from but visually identical to UI that users ***are accustomed to trust*** in ***a specific scenario***. “Accustomed to trust” is defined as anything a user is familiar with based on normal interaction with the OS/application but does not typically think of as a “trust decision”.
    - Examples
        - Displaying an email attachment with a file extension that is different from the file’s actual extension
    - Windows Store Applications

        The application displays web content downloaded from an external server.

        Ability for an attacker to present UI that is different from but visually identical to UI which users must rely on to make valid trust decisions in a default/common scenario.

        - Example
            - Displaying the fake login dialogbox. The user can be tricked into entering their account credentials.

        The application is loading any data from the local network IP address. Local address can be easy spoofed especially on the public Wi-Fi networks.

	• Security Assurances: A security assurance is either a security feature or another product feature/function that customers expect to offer security protection. Communications have messaged (explicitly or implicitly) that customers can rely on the integrity of the feature, and that's what makes it a security assurance. Security advisories will be released for a shortcoming in a security assurance that undermines the customer's reliance or trust. ○ Examples: ▪ Processes running with normal "user" privileges cannot gain "admin" privileges unless admin password/credentials have been provided via intentionally authorized methods. ▪ Internet-based JavaScript running in Microsoft Edge or Internet Explorer cannot control anything the host operating system unless the user has explicitly changed the default IE security settings.
Low	• Denial of Service ○ Temporary DoS – Requires restart of application. ▪ Example: opening a HTML document causes Microsoft Edge or Internet Explorer to AV and crash ▪ Example: opening a jpeg file causes a Windows Store photo viewer app to crash • Spoofing ○ Ability for an attacker to present UI that is different from but visually identical to UI ***where that UI serves as a single part of a larger attack scenario***. ○ Examples ▪ User has to go a "bad" web site, click on a button in spoofed dialog box, and is then susceptible to a vulnerability based on a different browser bug • Tampering ○ Temporary modification of any data that does not persist after restarting the OS/application. • Information Disclosure (Untargeted) ○ Examples ▪ Leak of non-sensitive heap memory