Exploitability Metrics

Attack Vector (AV)

This metric reflects the context by which vulnerability exploitation is possible. This metric value (and consequently the Base score) will be larger the more remote (logically, and physically) an attacker can be in order to exploit the vulnerable component. The assumption is that the number of potential attackers for a vulnerability that could be exploited from across the internet is larger than the number of potential attackers that could exploit a vulnerability requiring physical access to a device, and therefore warrants a greater score.

Metric Value	Description
Network (N)	A vulnerability exploitable with network access means the vulnerable component is bound to the network stack and the attacker's path is through OSI layer 3 (the network layer). Such a vulnerability is often termed "remotely exploitable" and can be thought of as an attack being exploitable one or more network hops away (e.g. across layer 3 boundaries from routers). An example of a network attack is an attacker causing a denial of service (DoS) by sending a specially crafted TCP packet from across the public internet (e.g. CVE-2004- 0230).
Adjacent Network (A)	A vulnerability exploitable with adjacent network access means the vulnerable component is bound to the network stack, however the attack is limited to the same shared physical (e.g. Bluetooth, IEEE 802.11), or logical (e.g. local IP subnet) network, and cannot be performed across an OSI layer 3 boundary (e.g. a router). An example of an Adjacent attack would be an ARP (IPv4) or neighbor discovery (IPv6) flood leading to a denial of service on the local LAN segment. See alsoCVE-2013-6014.
Local (L)	A vulnerability exploitable with local access means that the vulnerable component is not bound to the network stack, and the attacker's path is via read/write/execute capabilities. In some cases, the attacker may be logged in locally in order to exploit the vulnerability, otherwise, she may rely on User Interaction to execute a malicious file.
Physical (P)	A vulnerability exploitable with physical access requires the attacker to physically touch or manipulate the vulnerable component. Physical interaction may be brief (e.g. evil maid attack1) or persistent. Example of such an attack is a cold boot attack which allows an attacker to get access to disk encryption keys after gaining physical access to the system, or peripheral attacks such as Firewire/USB Direct Memory Access attacks.

Attack Complexity (AC)

This metric describes the conditions beyond the attacker's control that must exist in order to exploit the vulnerability. As described below, such conditions may require the collection of more information about the target, the presence of certain system configuration settings, or computational exceptions. Importantly, the assessment of this metric excludes any requirements for user interaction in order to exploit the vulnerability (such conditions are captured in the User Interaction metric). This metric value is largest the less complex is an attack.

Metric Value	New Description
Low	Specialized access conditions or extenuating circumstances do not exist. An attacker can expect repeatable success against the vulnerable component.

High (H)	A successful attack depends on conditions beyond the attacker's control. That is, a successful attack cannot be accomplished at will, but requires the attacker to invest in some measurable amount of effort in preparation or execution against the vulnerable component before a successful attack can be expected.2 For example, a successful attack may depend on an attacker overcoming any of the following conditions: • The attacker must conduct target-specific reconnaissance. For example, on target configuration settings, sequence numbers, shared secrets, etc. • The attacker must prepare the target environment to improve exploit reliability. For example, repeated exploitation to win a race condition, or overcoming advanced exploit mitigation techniques. • The attacker injects herself into the logical network path between the target and the resource requested by the victim in order to read and/or modify network communications (e.g. man in the middle attack).

Privileges Required (PR)

This metric describes the level of privileges an attacker must possess before successfully exploiting the vulnerability. This metric value is increasing as fewer privileges are required.

Metric Value	Description
None (N)	The attacker is unauthorized prior to attack, and therefore does not require any access to settings or files to carry out an attack.
Low (L)	The attacker is authorized with (i.e. requires) privileges that provide basic user capabilities that could normally affect only settings and files owned by a user. Alternatively, an attacker with Low privileges may have the ability to cause an impact only to non-sensitive resources.
High (H)	The attacker is authorized with (i.e. requires) privileges that provide significant (e.g. administrative) control over the vulnerable component that could affect component-wide settings and files.

User Interaction (UI)

This metric captures the requirement for a user, other than the attacker, to participate in the successful compromise the vulnerable component. This metric determines whether the vulnerability can be exploited solely at the will of the attacker, or whether a separate user (or user-initiated process) must participate in some manner. This metric value is greatest when no user interaction is required.

Metric Value	Description
None (N)	The vulnerable system can be exploited without interaction from any user.
Required (R)	Successful exploitation of this vulnerability requires a user to take some action before the vulnerability can be exploited. For example, a successful exploit may only be possible during the installation of an application by a system administrator.

Scope (S)

An important property captured by CVSS v3.0 is the ability for a vulnerability in one software component to impact resources beyond its means, or privileges. This consequence is represented by the metric Authorization Scope, or simply Scope.

Formally, Scope refers to the collection of privileges defined by a computing authority (e.g. an application, an operating system, or a sandbox environment) when granting access to computing resources (e.g. files, CPU, memory, etc). These privileges are assigned based on some method of identification and authorization. In some cases, the authorization may be simple or loosely controlled based upon predefined rules or standards. For example, in the case of Ethernet traffic sent to a network switch, the switch accepts traffic that arrives on its ports and is an authority that controls the traffic flow to other switch ports.

When the vulnerability of a software component governed by one authorization scope is able to affect resources governed by another authorization scope, a Scope change has occurred. Intuitively, one may think of a scope change as breaking out of a sandbox, and an example would be a vulnerability in a virtual machine that enables an attacker to delete files of the host OS (perhaps even its own VM). In this example, there are two separate authorization authorities: one that defines and enforces privileges for the virtual machine and its users, and one that defines and enforces privileges for the host system within which the virtual machine runs.

A scope change would not occur, for example, with a vulnerability in Microsoft Word that allows an attacker to compromise all system files of the host OS, because the same authority enforces privileges of the user's instance of Word, and the host's system files.

The Base score is greater when a scope change has occurred.

Metric Value	Description
Unchanged (U)	An exploited vulnerability can only affect resources managed by the same authority. In this case the vulnerable component and the impacted component are the same.
Changed (C)	An exploited vulnerability can affect resources beyond the authorization privileges intended by the vulnerable component. In this case the vulnerable component and the impacted component are different.

Impact Metrics

Confidentiality Impact (C)

This metric measures the impact to the confidentiality of the information resources managed by a software component due to a successfully exploited vulnerability. Confidentiality refers to limiting information access and disclosure to only authorized users, as well as preventing access by, or disclosure to, unauthorized ones. This metric value is increasing in the degree of loss to the impacted component.

Metric Value	Description
High (H)	There is total loss of confidentiality, resulting in all resources within the impacted component being divulged to the attacker. Alternatively, access to only some restricted information is obtained, but the disclosed information presents a direct, serious impact. For example, an attacker steals the administrator's password, or private encryption keys of a web server.
Low (L)	There is some loss of confidentiality. Access to some restricted information is obtained, but the attacker does not have control over what information is obtained, or the amount or kind of loss is constrained. The information disclosure does not cause a direct, serious loss to the impacted component.
None (N)	There is no loss of confidentiality within the impacted component.

Integrity Impact (I)

This metric measures the impact to integrity of a successfully exploited vulnerability. Integrity refers to the trustworthiness and veracity of information. This metric value is increasing in the consequence to the impacted component.

Metric Value	Description
High (H)	There is a total loss of integrity, or a complete loss of protection. For example, the attacker is able to modify any/all files protected by the impacted component. Alternatively, only some files can be modified, but malicious modification would present a direct, serious consequence to the impacted component.
Low (L)	Modification of data is possible, but the attacker does not have control over the consequence of a modification, or the amount of modification is constrained. The data modification does not have a direct, serious impact on the impacted component.
None (N)	There is no loss of integrity within the impacted component.

Availability Impact (A)

This metric measures the impact to the availability of the impacted component resulting from a successfully exploited vulnerability. While the Confidentiality and Integrity impact metrics apply to the loss of confidentiality or integrity of data (e.g., information, files) used by the impacted component, this metric refers to the loss of availability of the impacted component itself, such as a networked service (e.g., web, database, email). Since availability refers to the accessibility of information resources, attacks that consume network bandwidth, processor cycles, or disk space all impact the availability of an impacted component. This metric value is increasing in consequence to the impacted component.

Metric Value	Description
High (H)	There is total loss of availability, resulting in the attacker being able to fully deny access to resources in the impacted component; this loss is either sustained (while the attacker continues to deliver the attack) or persistent (the condition persists even after the attack has completed). Alternatively, the attacker has the ability to deny some availability, but the loss of availability presents a direct, serious consequence to the impacted component (e.g., the attacker cannot disrupt existing connections, but can prevent new connections; the attacker can repeatedly exploit a vulnerability that, in each instance of a successful attack, leaks a only small amount of memory, but after repeated exploitation causes a service to become completely unavailable).
Low (L)	There is reduced performance or interruptions in resource availability. Even if repeated exploitation of the vulnerability is possible, the attacker does not have the ability to completely deny service to legitimate users. The resources in the impacted component are either partially available all of the time, or fully available only some of the time, but overall there is no direct, serious consequence to the impacted component.
None (N)	There is no impact to availability within the impacted component.