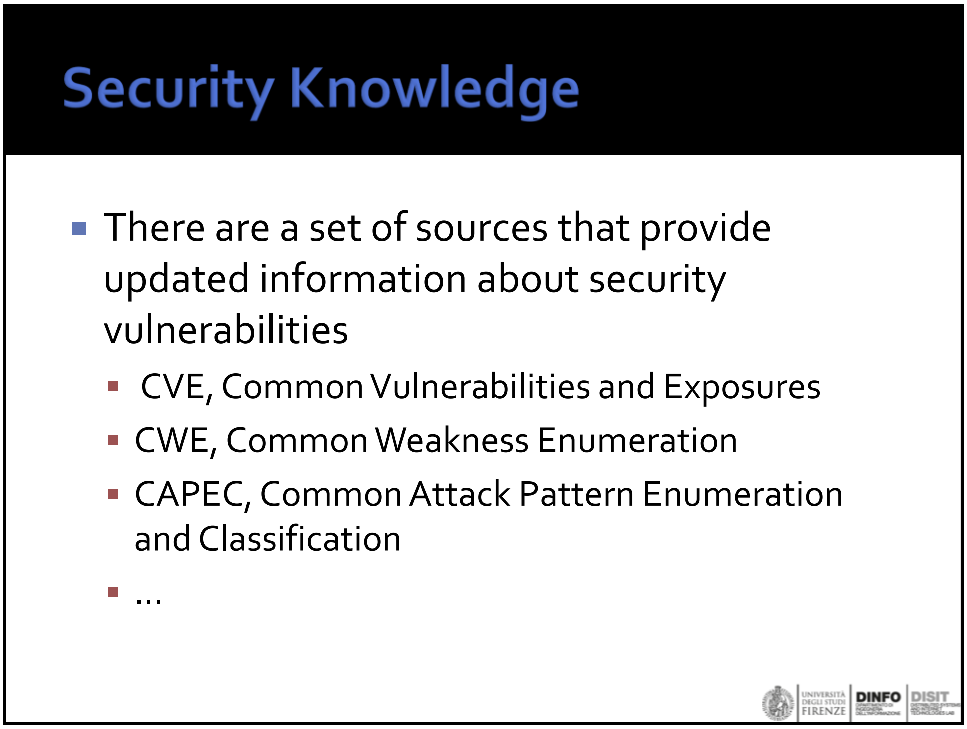
Security Knowledge



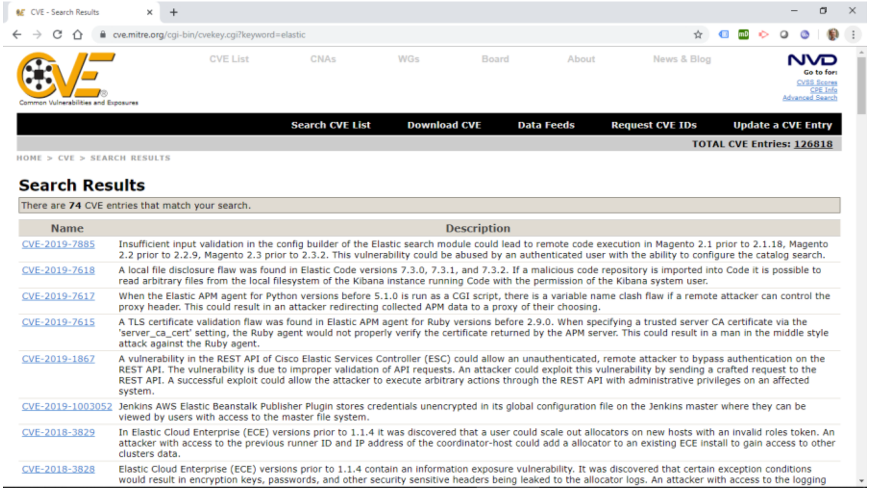
Security Knowledge

- There are a set of sources that provide updated information about security vulnerabilities
  - CVE, Common Vulnerabilities and Exposures
  - CWE, Common Weakness Enumeration
  - CAPEC, Common Attack Pattern Enumeration and Classification
  - …

CVE

- CVE is a list of entries—each containing an identification number, a description, and at least one public reference—for **publicly known cybersecurity vulnerabilities**.
- CVE Entries are used in numerous cybersecurity products and services from around the world, including the U.S. National Vulnerability Database (NVD).

https://cve.mitre.org/



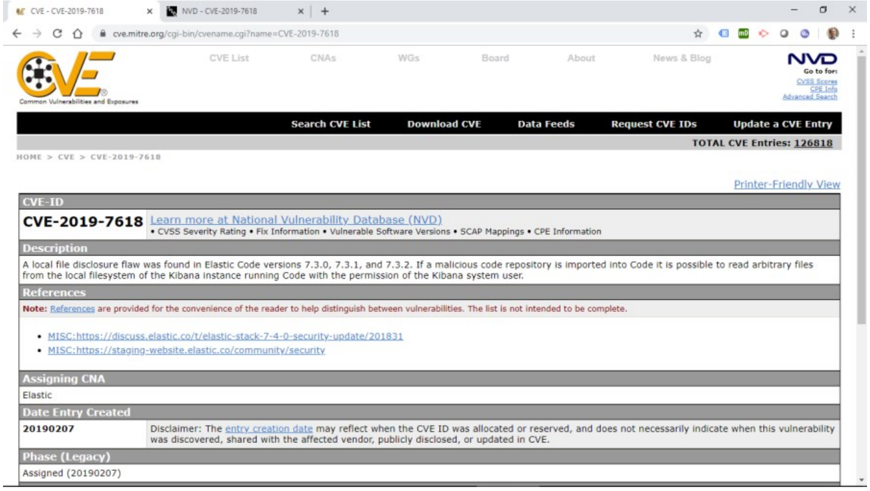
TOTAL CVE Entries: 126818

Search Results

There are **74** CVE entries that match your search.

Name	Description
CVE-2019-7885	Insufficient input validation in the config builder of the Elastic search module could lead to remote code execution in Magento 2.1 prior to 2.1.18, Magento 2.2 prior to 2.2.9, Magento 2.3 prior to 2.3.2. This vulnerability could be abused by an authenticated user with the ability to configure the catalog search.
CVE-2019-7618	A local file disclosure flaw was found in Elastic Code versions 7.3.0, 7.3.1, and 7.3.2. If a malicious code repository is imported into Code it is possible to read arbitrary files from the local filesystem of the Kibana instance running Code with the permission of the Kibana system user.
CVE-2019-7617	When the Elastic APM agent for Python versions before 5.1.0 is run as a CGI script, there is a variable name clash flaw if a remote attacker can control the proxy header. This could result in an attacker redirecting collected APM data to a proxy of their choosing.
CVE-2019-7615	A TLS certificate validation flaw was found in Elastic APM agent for Ruby versions before 2.9.0. When specifying a trusted server CA certificate via the 'server_ca_cert' setting, the Ruby agent would not properly verify the certificate returned by the APM server. This could result in a man in the middle style attack against the Ruby agent.
CVE-2019-1867	A vulnerability in the REST API of Cisco Elastic Services Controller (ESC) could allow an unauthenticated, remote attacker to bypass authentication on the REST API. The vulnerability is due to improper validation of API requests. An attacker could exploit this vulnerability by sending a crafted request to the REST API. A successful exploit could allow the attacker to execute arbitrary actions through the REST API with administrative privileges on an affected system.
CVE-2019-1003052	Jenkins AWS Elastic Beanstalk Publisher Plugin stores credentials unencrypted in its global configuration file on the Jenkins master where they can be viewed by users with access to the master file system.
CVE-2018-3829	In Elastic Cloud Enterprise (ECE) versions prior to 1.1.4 it was discovered that a user could scale out allocators on new hosts with an invalid roles token. An attacker with access to the previous runner ID and IP address of the coordinator-host could add a allocator to an existing ECE install to gain access to other clusters data.
CVE-2018-3828	Elastic Cloud Enterprise (ECE) versions prior to 1.1.4 contain an information exposure vulnerability. It was discovered that certain exception conditions would result in encryption keys, passwords, and other security sensitive headers being leaked to the allocator logs. An attacker with access to the logging

CVE - example



CVE-ID	
CVE-2019-7618	Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

A local file disclosure flaw was found in Elastic Code versions 7.3.0, 7.3.1, and 7.3.2. If a malicious code repository is imported into Code it is possible to read arbitrary files from the local filesystem of the Kibana instance running Code with the permission of the Kibana system user.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- MISC:https://discuss.elastic.co/t/elastic-stack-7-4-0-security-update/201831
- MISC:https://staging-website.elastic.co/community/security

Assigning CNA

Elastic

Date Entry Created

20190207	Disclaimer: The entry creation date may reflect when the CVE ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

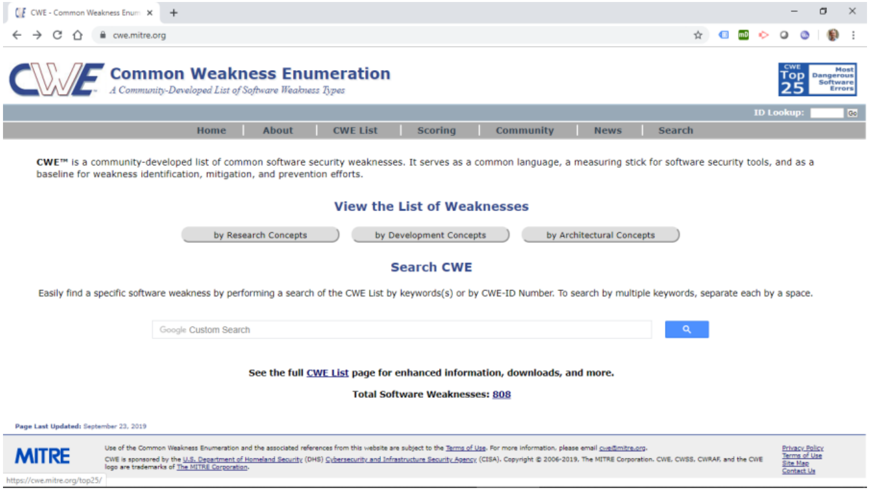
Phase (Legacy)

Assigned (20190207)

CWE

- **CWE** is a community-developed list of common software security weaknesses. It serves as a common language, a measuring stick for software security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

https://cwe.mitre.org/



CWE - Common Weakness Enum
cwe.mitre.org
CWE Top 25 Most Dangerous Software Errors
ID Lookup: Go
Home | About | CWE List | Scoring | Community | News | Search

Common Weakness Enumeration

A Community-Developed List of Software Weakness Types

CWE™ is a community-developed list of common software security weaknesses. It serves as a common language, a measuring stick for software security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

View the List of Weaknesses

by Research Concepts | by Development Concepts | by Architectural Concepts

Search CWE

Easily find a specific software weakness by performing a search of the CWE List by keywords(s) or by CWE-ID Number. To search by multiple keywords, separate each by a space.

Google Custom Search

See the full CWE List page for enhanced information, downloads, and more.

Total Software Weaknesses: 808

Page Last Updated: September 23, 2019

MITRE

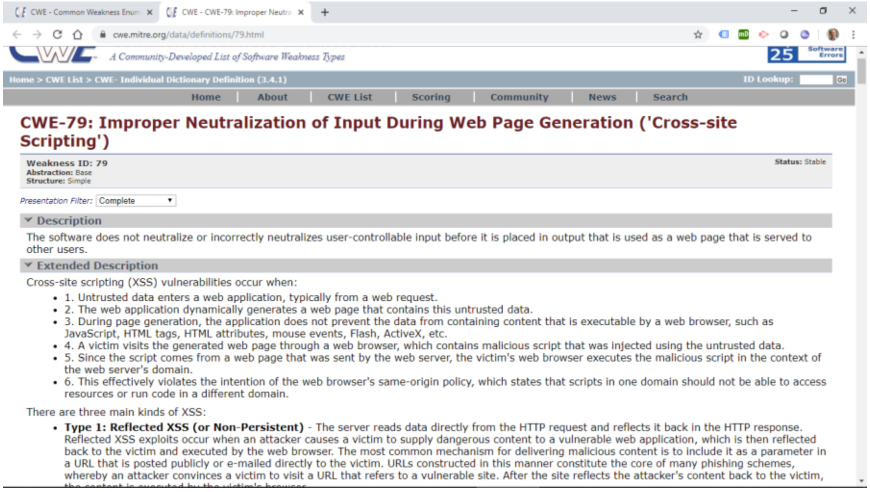
Use of the Common Weakness Enumeration and the associated references from this website are subject to the Terms of Use. For more information, please email cwe@mitre.org.

CWE is sponsored by the U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA). Copyright © 2006-2019, The MITRE Corporation. CWE, CWSS, CWRAF, and the CWE logo are trademarks of The MITRE Corporation.

Privacy Policy
Terms of Use
Site Map
Contact Us

https://cwe.mitre.org/top25/

CWE - XSS



CWE - Common Weakness Enum
CWE - CWE-79: Improper Neutra
cwe.mitre.org/data/definitions/79.html
25 Software Errors
ID Lookup: Go
Home | About | CWE List | Scoring | Community | News | Search

A Community-Developed List of Software Weakness Types

Home > CWE List > CWE- Individual Dictionary Definition (3.4.1)

CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

Weakness ID: 79
Abstraction: Base
Structure: Simple

Status: Stable

Presentation Filter: Complete

Description

The software does not neutralize or incorrectly neutralizes user-controllable input before it is placed in output that is used as a web page that is served to other users.

Extended Description

Cross-site scripting (XSS) vulnerabilities occur when:

- 1. Untrusted data enters a web application, typically from a web request.
- 2. The web application dynamically generates a web page that contains this untrusted data.
- 3. During page generation, the application does not prevent the data from containing content that is executable by a web browser, such as JavaScript, HTML tags, HTML attributes, mouse events, Flash, ActiveX, etc.
- 4. A victim visits the generated web page through a web browser, which contains malicious script that was injected using the untrusted data.
- 5. Since the script comes from a web page that was sent by the web server, the victim's web browser executes the malicious script in the context of the web server's domain.
- 6. This effectively violates the intention of the web browser's same-origin policy, which states that scripts in one domain should not be able to access resources or run code in a different domain.

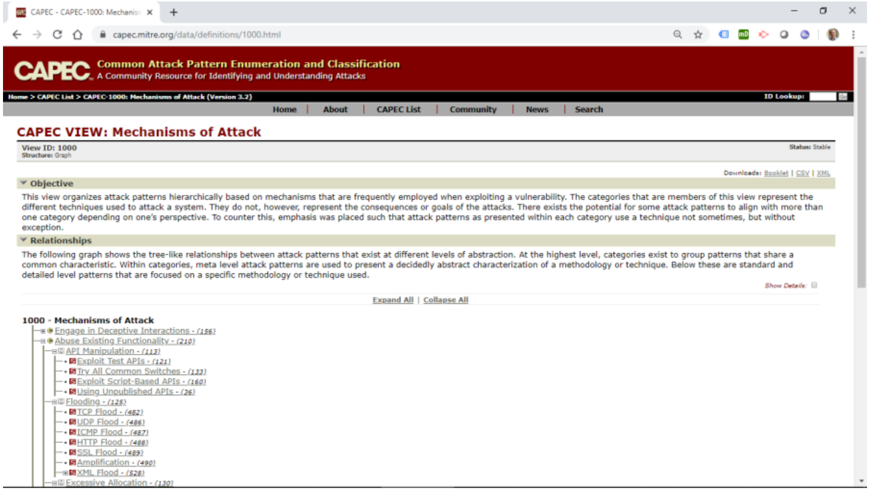
There are three main kinds of XSS:

- **Type 1: Reflected XSS (or Non-Persistent)** - The server reads data directly from the HTTP request and reflects it back in the HTTP response. Reflected XSS exploits occur when an attacker causes a victim to supply dangerous content to a vulnerable web application, which is then reflected back to the victim and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or e-mailed directly to the victim. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces a victim to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the victim, the content is executed by the victim's browser.

CAPEC – Common Attack Pattern Enumeration and Classification

- CAPEC helps by providing a comprehensive dictionary of known patterns of attack employed by adversaries to exploit known weaknesses in cyber-enabled capabilities.
- Understanding how the adversary operates is essential to effective cyber security.
- It can be used by analysts, developers, testers, and educators to advance community understanding and enhance defenses.

https://capec.mitre.org/


CAPEC - CAPEC-1000: Mechanis
capec.mitre.org/data/definitions/1000.html

CAPEC Common Attack Pattern Enumeration and Classification
A Community Resource for Identifying and Understanding Attacks

Home > CAPEC List > CAPEC-1000: Mechanisms of Attack (Version 3.2)

ID Lookup:

Home | About | CAPEC List | Community | News | Search

CAPEC VIEW: Mechanisms of Attack

View ID: 1000
Structure: Graph

Status: Stable

Downloads: Booklet | CSV | XML

Objective

This view organizes attack patterns hierarchically based on mechanisms that are frequently employed when exploiting a vulnerability. The categories that are members of this view represent the different techniques used to attack a system. They do not, however, represent the consequences or goals of the attacks. There exists the potential for some attack patterns to align with more than one category depending on one's perspective. To counter this, emphasis was placed such that attack patterns as presented within each category use a technique not sometimes, but without exception.

Relationships

The following graph shows the tree-like relationships between attack patterns that exist at different levels of abstraction. At the highest level, categories exist to group patterns that share a common characteristic. Within categories, meta level attack patterns are used to present a decidedly abstract characterization of a methodology or technique. Below these are standard and detailed level patterns that are focused on a specific methodology or technique used.

Show Details: □

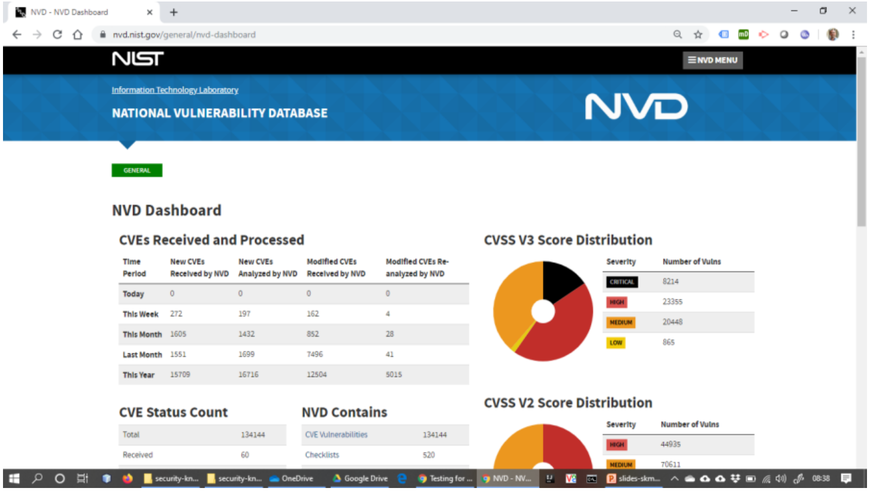
Expand All | Collapse All

1000 - Mechanisms of Attack
- Engage in Deceptive Interactions - (156)
- Abuse Existing Functionality - (210)
  - API Manipulation - (113)
    - Exploit Test APIs - (121)
    - Try All Common Switches - (133)
    - Exploit Script-Based APIs - (160)
    - Using Unpublished APIs - (36)
  - Flooding - (125)
    - TCP Flood - (482)
    - UDP Flood - (486)
    - ICMP Flood - (487)
    - HTTP Flood - (488)
    - SSL Flood - (489)
    - Amplification - (490)
    - XML Flood - (528)
  - Excessive Allocation - (130)

NIST - National Vulnerabilities Database (VND)

- Collects and classifies the CVE vulnerabilities
- CVEs are related with CWEs and with the products (CPE)
- Metrics are defined to measure the vulnerability dangerousness,
  - each vulnerability is classified in different aspects, see CVSS calculators
  - https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator
- provides machine readable JSON data of vulnerabilities description

NVD - dashboard



NVD Dashboard

CVEs Received and Processed

Time Period	New CVEs Received by NVD	New CVEs Analyzed by NVD	Modified CVEs Received by NVD	Modified CVEs Re-analyzed by NVD
Today	0	0	0	0
This Week	272	197	162	4
This Month	1605	1432	852	28
Last Month	1551	1699	7496	41
This Year	15709	16716	12504	5015

CVE Status Count

Total	134144
Received	60

NVD Contains

CVE Vulnerabilities	134144
Checklists	520

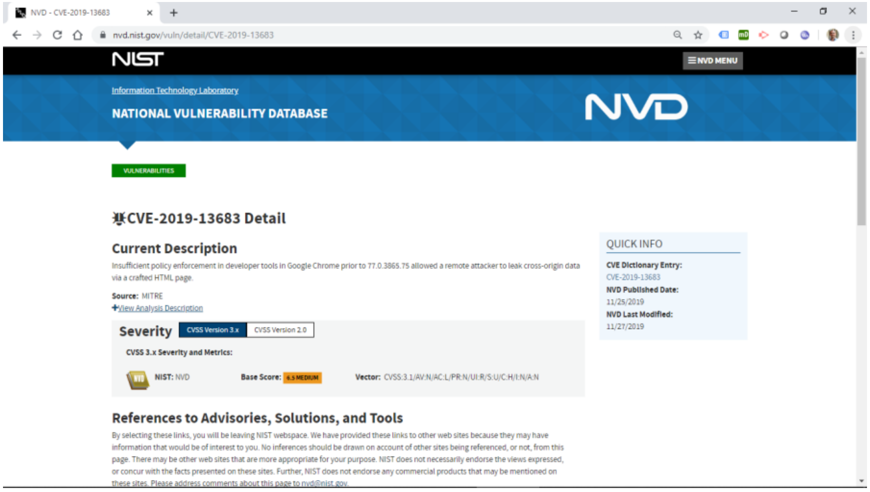
CVSS V3 Score Distribution

Severity	Number of Vulns
CRITICAL	8214
HIGH	23355
MEDIUM	20448
LOW	865

CVSS V2 Score Distribution

Severity	Number of Vulns
HIGH	44935
MEDIUM	70611

Example


NVD - CVE-2019-13683
nvd.nist.gov/vuln/detail/CVE-2019-13683
NIST
☰ NVD MENU
Information Technology Laboratory
NATIONAL VULNERABILITY DATABASE
NVD
VULNERABILITIES

CVE-2019-13683 Detail

Current Description

Insufficient policy enforcement in developer tools in Google Chrome prior to 77.0.3865.75 allowed a remote attacker to leak cross-origin data via a crafted HTML page.

Source: MITRE

+View Analysis Description

Severity CVSS Version 3.x | CVSS Version 2.0

CVSS 3.x Severity and Metrics:

NIST: NVD **Base Score:** 6.5 MEDIUM **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N

QUICK INFO

CVE Dictionary Entry:
CVE-2019-13683
NVD Published Date:
11/25/2019
NVD Last Modified:
11/27/2019

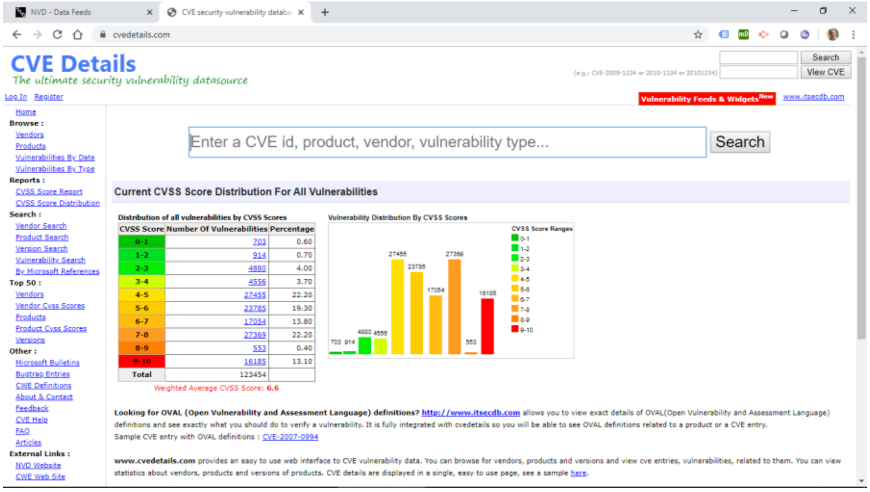
References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your purpose. NIST does not necessarily endorse the views expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commercial products that may be mentioned on these sites. Please address comments about this page to nvd@nist.gov.

CVE details

- **www.cvedetails.com** provides an easy to use web interface to CVE vulnerability data. You can browse for vendors, products and versions and view cve entries, vulnerabilities, related to them. You can view statistics about vendors, products and versions of products
- CVE vulnerability data are taken from National Vulnerability Database (NVD) xml feeds provided by NIST.
- Additional data from several sources like exploits from www.exploit-db.com, vendor statements and additional vendor supplied data, Metasploit modules are also published in addition to NVD CVE data.
- Vulnerabilities are classified by cvedetails.com using keyword matching and cwe numbers if possible, but they are mostly based on keywords.
- Unless otherwise stated CVSS scores listed on this site are "CVSS Base Scores" provided in NVD feeds. Vulnerability data are updated daily using NVD feeds.

https://www.cvedetails.com/



CVE Details

The ultimate security vulnerability datasource

(e.g.: CVE-2009-1234 or 2010-1234 or 20101234)

Enter a CVE id, product, vendor, vulnerability type...

Current CVSS Score Distribution For All Vulnerabilities

Distribution of all vulnerabilities by CVSS Scores

CVSS Score	Number Of Vulnerabilities	Percentage
0-1	703	0.60
1-2	914	0.70
2-3	4880	4.00
3-4	4556	3.70
4-5	27455	22.20
5-6	23785	19.30
6-7	17054	13.80
7-8	27369	22.20
8-9	553	0.40
9-10	16185	13.10
Total	123454	

Weighted Average CVSS Score: **6.6**

Looking for OVAL (Open Vulnerability and Assessment Language) definitions? http://www.itsecdb.com allows you to view exact details of OVAL(Open Vulnerability and Assessment Language) definitions and see exactly what you should do to verify a vulnerability. It is fully integrated with cvedetails so you will be able to see OVAL definitions related to a product or a CVE entry.
Sample CVE entry with OVAL definitions : CVE-2007-0994

www.cvedetails.com provides an easy to use web interface to CVE vulnerability data. You can browse for vendors, products and versions and view cve entries, vulnerabilities, related to them. You can view statistics about vendors, products and versions of products. CVE details are displayed in a single, easy to use page, see a sample here.

E-Commerce security

- Which are the security measures to be taken when handling payments?
- credit cards information are a very sensitive personal data
- There are some guidelines to be followed
- For example on OWASP
  - https://www.owasp.org/images/f/f7/Security_of_Payment_cards.doc
  - https://cheatsheetseries.owasp.org/cheatsheets/Transaction_Authorization_Cheat_Sheet.html