

# CVE API Documentation

## Overview

This API provides endpoints for retrieving lists of CVEs (Common Vulnerabilities and Exposures) and details of specific CVEs from the National Vulnerability Database (NVD). The API allows retrieving, filtering, sorting, and pagination of the CVE data. The endpoints are built using Node.js without any external libraries.

## Base URL for EndPoints

<http://localhost:3001>

## Environment Variables(.env)

**BASE\_API:** <https://services.nvd.nist.gov/rest/json/cves/2.0> (The base URL for the NVD API)

**PORT:** 3001

## Endpoints

### 1. Get List of CVEs

URL: /cves/list

Method: GET

Description: Retrieves a paginated list of CVEs with optional filtering and sorting.

Query Parameters:

1. page (integer, optional): The page number for pagination. Defaults to 1.
2. itemsPerPage (integer, optional): The number of items per page. Defaults to 10.
3. filterType (string, optional): The type of filter to apply. Can be one of the following:
  - cveld: Filter by CVE ID.
  - Modified Year: Filter by the last modified date (year).
  - CVSSv2: Filter by CVSSv2 metrics.
  - CVSSv3: Filter by CVSSv3 metrics.
  - filterValue (string, optional): The value for the filter.
  - sortOrder (string, optional): The sort order of the results. Can be asc for ascending or desc for descending. Defaults to asc.

Response:

200 OK: Returns a JSON object with the list of CVEs, total number of CVEs, and total pages.

500 Internal Server Error: If there is an error fetching the data.

Example Request:

http

GET /cves/list?page=1&itemsPerPage=10&filterType=cveId&filterValue=CVE-1999-2001&sortOrder=asc

Example Response:

Json:

```
{
  "cveLength": 1,
  "cves": [
    {
      "id": "CVE-1999-0095",
      "identifier": "cve@mitre.org",
      "published": "10-1-1988 4:00:00 AM",
      "lastModified": "6-11-2019 8:29:00 PM",
      "status": "Modified"
    }
  ],
  "totalPages": 1
}
```

## 2. Get CVE Details

URL: /cves

Method: GET

Description: Retrieves detailed information about a specific CVE.

Query Parameters:

1. cveId (string, required): The CVE ID for which details are to be retrieved.

Response:

200 OK: Returns a JSON object with the details of the specified CVE.

400 Bad Request: If the cveld parameter is missing.

500 Internal Server Error: If there is an error fetching the CVE details.

Example Request:

http

GET /cves?cveld=CVE-1995-0095

Example Response:

Json:

```
{
  "id": "CVE-1999-0095",
  "description": "The debug command in Sendmail is enabled, allowing attackers to execute
commands as root.",
  "cvssMetricV2": {
    "version": "2.0",
    "vectorString": "AV:N/AC:L/Au:N/C:C/I:C/A:C",
    "accessVector": "NETWORK",
    "accessComplexity": "LOW",
    "authentication": "NONE",
    "confidentialityImpact": "COMPLETE",
    "integrityImpact": "COMPLETE",
    "availabilityImpact": "COMPLETE",
    "baseScore": 10
  },
  "metrics": {
    "cvssMetricV2": [
      {
        "source": "nvd@nist.gov",
        "type": "Primary",
        "cvssData": {
```

```
"version": "2.0",
"vectorString": "AV:N/AC:L/Au:N/C:C/I:C/A:C",
"accessVector": "NETWORK",
"accessComplexity": "LOW",
"authentication": "NONE",
"confidentialityImpact": "COMPLETE",
"integrityImpact": "COMPLETE",
"availabilityImpact": "COMPLETE",
"baseScore": 10
},
"baseSeverity": "HIGH",
"exploitabilityScore": 10,
"impactScore": 10,
"acInsufInfo": false,
"obtainAllPrivilege": true,
"obtainUserPrivilege": false,
"obtainOtherPrivilege": false,
"userInteractionRequired": false
}
]
},
"configurations": [
{
  "nodes": [
    {
      "operator": "OR",
      "negate": false,
      "cpeMatch": [
        {
          "vulnerable": true,
          "criteria": "cpe:2.3:a:eric_allman:sendmail:5.58:*:*:*:*:*:*:*"
```

```

        "matchCriteriaId": "1D07F493-9C8D-44A4-8652-F28B46CBA27C"
      }
    ]
  }
]
}
]
}

```

## Functions

### 1.formatDateTime(dateTimeString)

Description: Formats a datetime string into a human readable format.

Parameters:

dateTimeString (string): The datetime string to format.

Returns: A formatted datetime string in the format DDMMYYYY HH:MM:SS.

Example:

```

const formattedDateTime = formatDateTime("20210701T12:34:56Z");
// Returns "01072021 12:34:56"

```

### 2.getCvesList(req, res, parsedUrl)

Description: Handles the /cves/list endpoint to retrieve a list of CVEs with optional filtering, sorting, and pagination.

### 3. getCveDetails(req, res, parsedUrl)

Description: Handles the /cvedetails endpoint to retrieve detailed information about a specific CVE.

Response: Sends a JSON response with the details of the specified CVE.

Error Handling:

If there is an error fetching data from the NVD API, a 500 Internal Server Error response is sent with a message "Error fetching data" or "Error fetching CVE details".

If the cveId parameter is missing in the /cvedetails endpoint, a 400 Bad Request response is sent with a message "CVE ID is required".

Parameters for `getCvesList` and `getCveDetails`:

`req` (object): The HTTP request object.

`res` (object): The HTTP response object.

`parsedUrl` (object): The parsed URL object.

## Usage

1. Ensure the environment variables are set correctly, particularly `BASE_API`.
2. Start the server using `npm start`.
3. Use the endpoints to retrieve CVE data with the appropriate query parameters.