Seguridad - OWASP

Seguridad

- Conocer entorno de mi aplicación
  - Lenguajes y Frameworks utilizados
    - Python, Flask
    - Javascript, Jquery, React, Highcharts, etc
  - Dependencia de bibliotecas
    - Validaciones
    - Procesamiento imágenes
    - Cálculos
  - Almacenamiento
    - MySQL
    - Archivos

Seguridad

- Flask: Consideraciones de seguridad
  - XSS: Cross-Site Scripting
    - Inyección de HTML y Javascript
    - Desarrolladores **debemos** escapar y limpiar valores para que no tengan elementos HTML
    - Jinja2 escapa automáticamente todos los valores, se debe tener cuidado
      - Generar HTML sin ayuda de Jinja2
      - Cuando se desee descargar archivos HTML, usar cabecera `Content-Disposition: attachment`
      - Usar comillas en atributos de HTML:

```
<input value="{{ value }}">
```

Seguridad

- Flask: XSS
  - Si no cuidamos la inyección en atributos, un atacante podría inyectar:

    `onmouseover=alert(document.cookie)`

    - Un atacante preparado podría
      - Enviar las cookies a otro destino en lugar de hacer `alert`
      - Usar CSS para llenar toda la página y el usuario al mover el mouse genera el ataque
  - Jinja no protege el atributo `href` del tag a, pues puede contener Javascript:

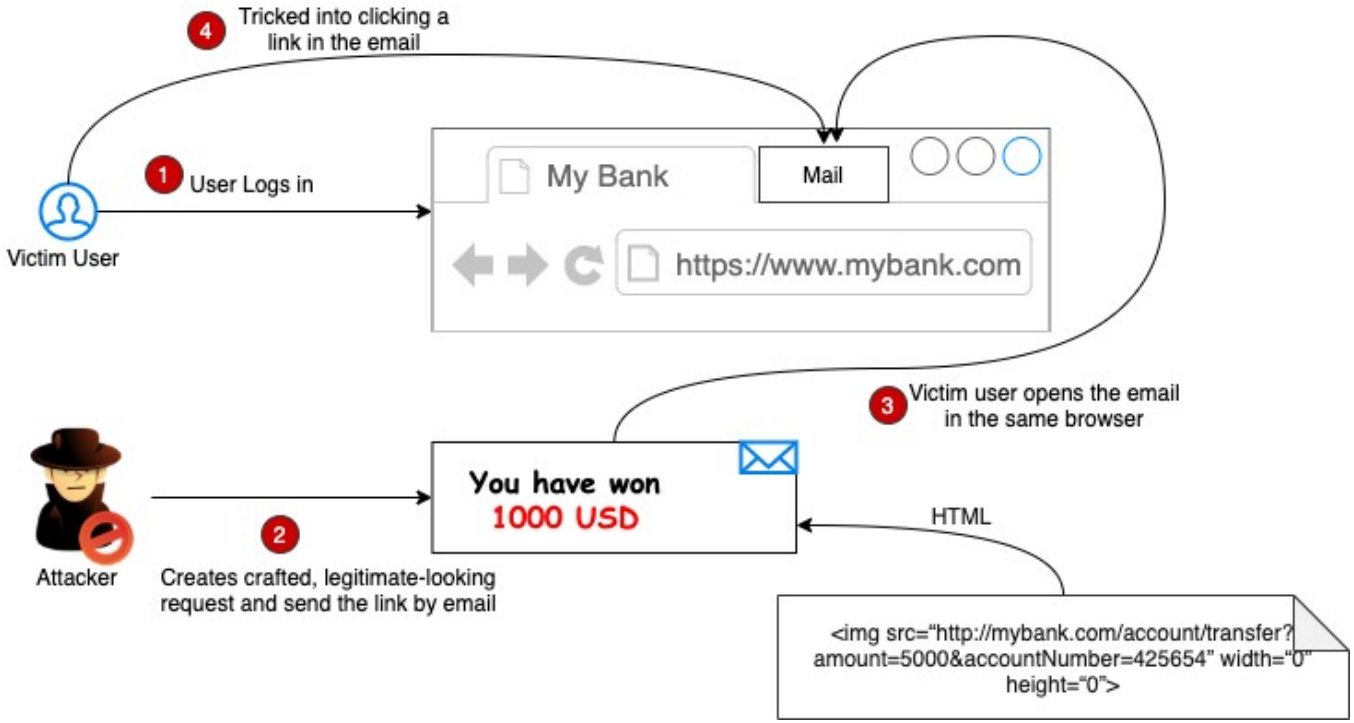
```
<a href="{{ value }}">click here</a>
<a href="javascript:alert('unsafe');">click here</a>
```

Seguridad

- Flask: XSS
  - Revisar definiciones para cabecera HTTP Content-Security-Policy
    - https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy
- Cross-Site Request Forgery (CSRF)
  - Falsificación de solicitudes entre sitios
  - Obliga al usuario a ejecutar acciones no deseadas en una aplicación web en la cual está autenticado
  - Usando un enlace por email o chat un atacante puede engañar a un usuario para ejecutar acciones

Seguridad

- Flask: CSRF



Seguridad

- Flask: CSRF
  - Si la información de autenticación se almacena en cookies:
    - Se enviará en cada solicitud entre cliente y servidor
      - Esto incluye solicitudes generadas por sitios de terceros
    - Podríamos tener una URL para eliminar información de un usuario que se llama por método POST:

      http://example.com/user/delete
    - Un atacante podría crear una página que envía una solicitud POST con Javascript
      - Un usuario autenticado en example.com, podría cargar esa página y su perfil podría ser eliminado

Seguridad

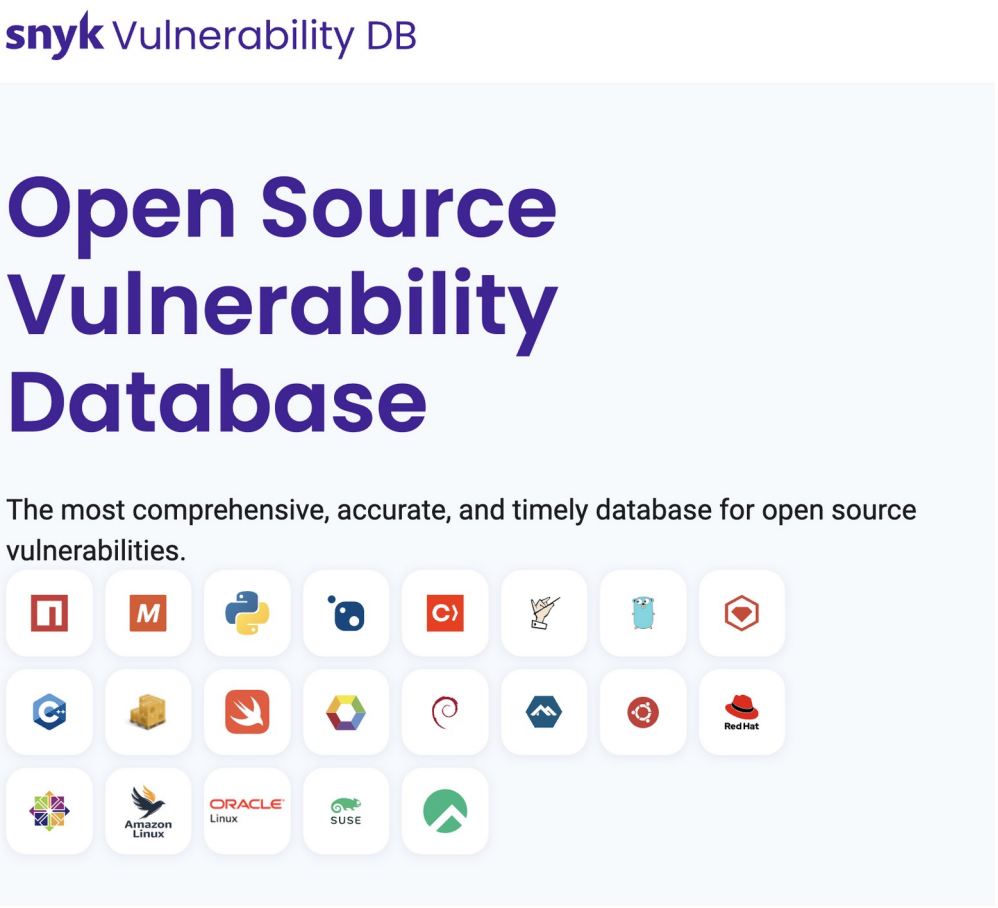
- Flask: CSRF
  - ¿Cómo prevenir?
    - Para cada solicitud que modifica contenido en el servidor se debería usar un “one-time” token
      - Se debe almacenar en una cookie y transmitirlo con la solicitud al servidor
      - El servidor lleva registro de los token generado y compara con el que recibe en la nueva solicitud
        - » Si son iguales, realizar el procesamiento
    - Flask no provee un mecanismo, se puede usar:

      https://flask-wtf.readthedocs.io/en/1.0.x/
- Otras recomendaciones para Flask:

  https://flask.palletsprojects.com/en/2.3.x/security/

Seguridad

- Revisión de vulnerabilidades conocidas:
  - https://security.snyk.io/



Seguridad

- Catálogo de vulnerabilidades
  - https://cve.mitre.org/

CVE List CNAs WGs Board About News & Blog

NVD
Go to for:
CVSS Scores
CPE Info

Search CVE List | Downloads | Data Feeds | Update a CVE Record | Request CVE IDs

TOTAL CVE Records: 202620

NOTICE: Transition to the all-new CVE website at WWW.CVE.ORG and CVE Record Format JSON are underway.

NOTICE: Changes are coming to CVE List Content Downloads in 2023.

The mission of the CVE® Program is to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities.

CVE News

News has moved to the new CVE website.

Go to new News page >>

CVE Podcast

Podcasts have moved to the new CVE website.

Go to new Podcast page >>

CVE Blog

Blogs are moving to the new CVE website.

Go to new Blogs page >>

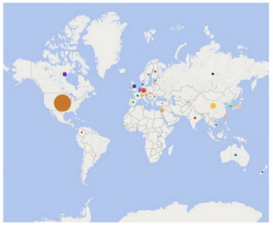
Become a CNA

CVE Numbering Authorities, or "CNAs," are essential to the CVE Program's success and every CVE Record is added to the CVE List by a CNA.

Join today!

- Business benefits
- No fee or contract
- Few requirements
- Easy to join

Go to new CVE website



Learn How to Become a CNA >>>

Watch CNA Onboarding Videos >>

Newest CVE Records

Tweets from @CVEnew

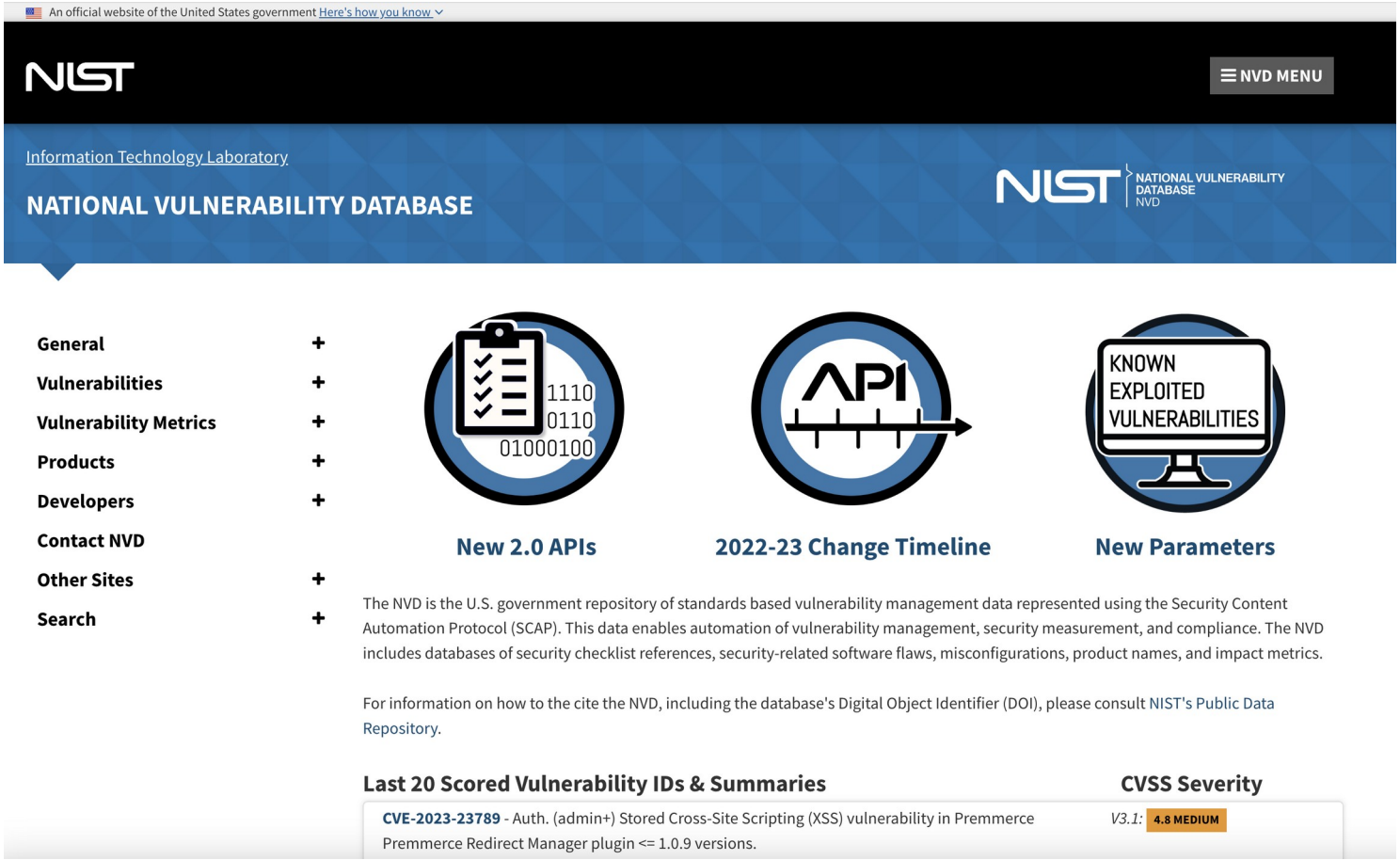
CVE
@CVEnew · 2h

CVE-2023-2700 A vulnerability was found in libvirt. This security flaw ouccers due to repeatedly querying an SR-IOV PCI device's capabilities that exposes a memory leak caused by a failure to free the virPCIVirtualFunction array within the parent ... cve.mitre.org/cgi-bin/cvenam...

Follow @CVEnew >>

Seguridad

- “National Vulnerability Database”
  - https://nvd.nist.gov/



OWASP

- Open Web Application Security Project
  - Proyecto de código abierto
  - Metodología de seguridad de auditoría web
    - Abierta y colaborativa
    - Analiza la seguridad de aplicaciones web
    - Referente en auditorías de seguridad

    https://owasp.org/
    https://owasp.org/Top10/

Pruebas seguridad

- Pruebas iniciales:
  - Lighthouse:

    https://chrome.google.com/webstore/detail/lighthouse/blipmdconlkpinefehnmjammfjpmpbjk

- Herramientas para evaluar vulnerabilidades:
  - https://owasp.org/www-community/Vulnerability_Scanning_Tools

    - OWASP Zed Attack Proxy (ZAP)
      - https://www.zaproxy.org/
    - nikto2
      - https://cirt.net/nikto2