**TỔNG QUAN**

* Nhóm nghiên cứu đã tổng hợp các quy chuẩn API sử dụng để tương tác giữa HOST-IPS và IPS-manager, tất cả các trao đổi trong hệ thống đều tuân theo chuẩn JSON
* Nhằm phục vụ cho quá trình xác thực và định danh, nhóm BKCS yêu cầu IPS-manager thiết kế một bảng người dung nhằm phục vụ mục địch này, yêu cầu tooig thiểu như phải có 3 trường dưới đây:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tên tài khoản** | **Mật khẩu** | **Khóa** |
| **Ví dụ** | Bkcs | Bkcs2020 | Ev28ygJqngOhWd77miVgTe6FJDFRgnFE |

* Trong đó, khóa là một chuỗi 32 ký tự bất kỳ, được sử dụng để xác thực API trong suốt quá trình hoạt động của máy, quản trị viện sau khi đăng nhập IPS-manager sẽ lấy được key và có nhiệm vụ copy key vào máy trạm để định danh và xác thực máy trạm. Tất cả các API trong hệ thống đều thuộc dạng POST, với tổi thiểu trường “key” nhằm nhiệm vụ xác thực hệ thống
* Nhóm nghiên cứu đã cài đặt hoàn thiện mô hình lên server 112.137.130.53 để bên đại học công nghệ nghiên cứu

**DANH SÁCH CHI TIẾT CÁC API**

# API cập nhật các thông báo quét virus lên Manager IPS

* Đường dẫn: **/virus-scan**
* Phương thức: **POST**
* Header: ***Accept: application/json***
* Nội dung:

**{**

[

{

"path": "<path>",

"time": "<time>"

},...

],

‘key’: <client\_key>

}

Trong đó:

* path: Đường dẫn tới thư mục thực hiện quét virus
* time: Thời gian quét
* client\_key: khóa định danh người dùng

# API cập nhật các thông báo theo dõi tệp tin lên Manager IPS

* Đường dẫn: **/moniter-update**
* Phương thức: **POST**
* Header: ***Accept: application/json***
* Nội dung:

{

"alert\_list":[

[<id>, <time>, <msg>, <path>]

,..

.],

“key”: < client\_key>

}

- Đối tượng với thuộc tính alert\_list chứa danh sách các thông báo sau khi quét. Mỗi phần tử trong danh sách là một mảng gồm:

|  |  |
| --- | --- |
| **Tên** | **Ý nghĩa** |
| path | Định danh của thông báo được lưu trong CSDL Host IPS |
| time | Thời gian xảy ra sự kiện với file theo định dạng  **Y-M-D h:m:s** |
| id | Thông điệp sự kiện:   * + - "File created." – File được tạo mới     - "File changed." – File đã bị chỉnh sửa     - "File access." – File đã bị truy cập     - "File deleted." – File đã bị xóa     - "Folder deleted." – Thư mục theo dõi đã bị xóa |
| client\_key | Đường dẫn tuyệt đối đến file xảy ra sự kiện. |
| key | Định danh máy trạm |

# API cập nhật các thông báo kiểm tra toàn vẹn lêm Manager IPS

* Đường dẫn: **/integrity-update**
* Phương thức: **POST**
* Header: ***Accept: application/json***
* Nội dung:

**{**

"alert\_list"**:[**

**[<**id**>,** **<**time**>,** **<**msg**>,** **<**path**>]**

**,...**

**],**

“key”: <client\_key>

**}**

Đối tượng với thuộc tính **alert\_list** chứa danh sách các thông báo sau khi quét. Mỗi phần tử trong danh sách là một mảng gồm:

|  |  |
| --- | --- |
| **Tên** | **Ý nghĩa** |
| id | Định danh của thông báo được lưu trong CSDL Host IPS |
| time | Thời gian xảy ra sự kiện với file theo định dạng  **Y-M-D h:m:s** |
| msg | Thông điệp sự kiện:   * + - "The new file add to folder." – File được thêm mới vào thư mục     - "File is changed." – File bị thay đổi nội dung     - "File is deleted." – File bị xóa bỏ     - "Registry value change. " – Giá trị của khóa registry bị thay đổi     - "Registry value add. " – Khóa registry thêm giá trị mới     - "Registry value deleted." – Giá trị của khóa registry bị xóa     - "Registry key add. " – Khóa registry được thêm mới     - "Registry key deleted." – Khóa registry bị xóa |
| path | Đường dẫn tuyệt đối đến đối tượng được kiểm tra toàn vẹn |
| client\_key | Định danh khóa |

# Xây dựng API cho mô-đun quét lỗ hổng hệ thống

* Mô tả: Xây dựng API gửi thông tin gia cố hệ thống
* Đường dẫn**: /checkservice**
* Phương thức: **POST**
* Header: Accept: **application/json**
* Nội dung:

{

[ ‘time:’: '< Thời gian quét>',

‘name’: '<Tên máy trạm thực hiện quét>',

‘Type’ ':'<Kiểu quét>',

‘content’: '<Nội dung thực hiện>',

],

‘key’: '<khóa định danh>'

}

# IPS manager: Cập nhật luật tường lửa

* Mô tả: Định kỳ Host IPS gửi truy vấn lên IPS manager để cập nhật các luật tường lửa mới nhất.
* Đường dẫn**: /** **/get/update\_rule\_firewall**
* Phương thức: **POST**
* Header: **Accept: application/json**
* Nội dung gửi { <key>: “Khóa định danh” }
* Nội dung trả về

|  |
| --- |
| [      {          "Rule Name": "< Rule Name>",          "Direction": "< Direction>",          "LocalIP": <Local IP>,          "RemoteIP": "<Remote IP>",          "LocalPort": "<LocaPort>",          "RemotePort": "< Remote Port>",          "Action": "<Action>"      }, …  ] |

Trong đó:

|  |  |
| --- | --- |
| **Tên biến** | **Nội dung** |
| Local IP | Địa chỉ nguồn của gói tin:   * Phải tuần theo dạng xxx.xxx.xxx.xxx (VD: 127.0.0.1 ) * Nếu để trông phải có giá trị ANY |
| Remote IP | Địa chỉ đích của gói tin:   * Phải tuần theo dạng xxx.xxx.xxx.xxx (VD: 127.0.0.1 ) * Nếu để trông phải có giá trị ANY |
| Local Port | Cổng dịch vụ nguồn của gói tin, có thể là một danh sách liên tiếp, ví dụ: (80-88), nếu để trông phải có ANY |
| Remote Port | Cổng dịch vụ nguồn của gói tin, có thể là một danh sách liên tiếp, ví dụ: (80-88), nếu để trông phải có ANY |
| Rule Name | Tham số tên luật, là chuỗi kí tự bất kì |
| Direction | Luật áp dụng cho gói tin đi vào|ra, có 2 kiểu {in, out} |

# IPS manager: Cập nhật thông tin ứng dụng

* Mô tả: Định kỳ Host IPS gửi truy vấn lên IPS manager để cập nhật các thông tin ứng dụng mới nhất.
* Mô tả: Định kỳ Host IPS gửi truy vấn lên IPS manager để cập nhật các luật tường lửa mới nhất.
* Đường dẫn: /**update\_app\_window** và /**update\_app\_linux**
* Phương thức: **POST**
* Header: Accept: **application/json**
* Nội dung gửi { <key>: “Khóa định danh” }
* Nội dung trả về:

{

[

    {

        "Name": "< Rule Name>",

        "Url ": "< url >",

        "Star ": < star IP>,

        "Version ": "< version>",

        " Cve\_list\_version ": "< cve\_list\_version >",

        " Rate ": "< rate>"

    }, …

],

key’:” Ev28ygJqngOhWd77miVgTe6FJDFRgnFE”

}

Trong đó:

|  |  |
| --- | --- |
| name | Tên ứng dụng |
| url | Đường dẫn tải ứng dụng |
| star | Mức đánh giá xếp hạng ứng dụng |
| version | Phiên bản mới nhất của ứng dụng |
| cve\_list\_version | URL trang công bỗ lỗ hổng bảo mật ứng dụng |
| rate | Đánh giá ứng dụng |

# API mô-đun cập nhật luật clamAV

# API lấy danh sách luật ứng với máy trạm

* Mô tả: Xây dựng API lấy danh sách luật ứng với máy trạm
* Đường dẫn**: get/clamlist**
* Phương thức: **POST**
* Header: Accept: **application/json**
* Nội dung gửi { <key>: “Khóa định danh” }
* Nội dung trả về:

{

[ “<Tên máy trạm 1>”: {“< Tên bộ luật 1>”, “<Tên bộ luật 2>”},

“<Tên máy trạm 1>”: { “<Tên bộ luật 2>”},

…]

‘key’ : <Khóa định danh>

}

Ví dụ:

{

[ “<bkcsPC1>”: {“< Trojan>”, “<DGA>”},

“<bkcsPC2>”: { “<DGA>”}],

‘key’:” Ev28ygJqngOhWd77miVgTe6FJDFRgnFE”

}

# API tải danh sách các gói luật

* Mô tả: Xây dựng API lấy danh sách gói luật
* Đường dẫn**: get/clam/<Tên gói luật>**
* Phương thức: **GET**
* Nội dung: Tải bộ luật được đóng gói dưới dạng ZIP

Lưu ý:

* Tên bộ luật được đặt tùy ý.
* Nội dung luật phải theo đúng cú pháp chuẩn của ClamAV, có thể xem tại:

<https://www.clamav.net/documents/creating-signatures-for-clamav>

* Nội dung luật phải đi kèm với định dạng tương ứng, bao gồm:

+ ndb, ndu: Quy định tập luật mở rộng

+ ldb, ldu, idb: Tập luật logic

+ cbc: Tập luật dạng bytecode

+ info: Tập kiểm tra thông tin cơ bản

* Tên tệp nén phải trùng với tên thư mục chứa luật

# API mô-đun cập nhật luật Modsecurity

# API lấy danh sách luật ứng với máy trạm

Mô tả: Xây dựng API lấy danh sách luật ứng với máy trạm

* Đường dẫn**: get/modlist**
* Phương thức: **POST**
* Header: Accept: **application/json**
* Nội dung gửi { <key>: “Khóa định danh” }
* Nội dung trả về:

{

[ “<Tên máy trạm 1>”: {“< Tên bộ luật 1>”, “<Tên bộ luật 2>”},

“<Tên máy trạm 1>”: { “<Tên bộ luật 2>”}

…],

‘key’ : <Khóa định danh>

}

Ví dụ:

{

[ “<bkcsPC1>”: {“< trustware>”, “<CRS>”},

“<bkcsPC2>”: { “<trustway>”}],

‘key’:” Ev28ygJqngOhWd77miVgTe6FJDFRgnFE”

}

# API tải danh sách các gói luật

* Mô tả: Xây dựng API lấy danh sách gói luật
* Đường dẫn**: get/clam/<Tên gói luật>**
* Phương thức: **GET**
* Nội dung: Tải bộ luật được đóng gói dưới dạng ZIP

Lưu ý:

* Tên bộ luật được đặt tùy ý.
* Khuyến khích dung các bộ luật có sẵn để tránh xung đột
* Các bộ luật tự do nên có ID 7 chữ số, phải tuân theo đúng định dạng luật của Modsecurity

**DANH SÁCH FILE THỰC THI TRONG HỆ THỐNG**

# Tường lửa Window

* File thực thi : window\_firewall
* Mô tả: Thêm, xem, sửa, xóa luật tường lửa Window.
* Tham số đầu vào:

|  |  |
| --- | --- |
| **Cú pháp khi gọi script** | **Nội dung** |
| --protocol -p | Giao thức của gói tin. |
| --src-ip | Địa chỉ nguồn của gói tin. |
| --dst-ip | Địa chỉ đích của gói tin. |
| --src-port | Cổng dịch vụ nguồn của gói tin. |
| --dst-port | Cổng dịch vụ đích của gói tin. |
| -h, --help | Hiển thị trợ giúp |
| -L, --list-rule | Lấy danh sách luật thoe cấu trúc JSON |
| -A, --append-rule | Thêm luật, theo sau cần các tham số như tên, địa chỉ cổng để xác định luật cần thêm. |
| -D, --delete-rule | Xóa luật, theo sau cần các tham số như tên, địa chỉ cổng để xác định luật cần thay đổi. |
| --enable-rule | Bật luật, theo sau cần các tham số như tên, địa chỉ cổng để xác định luật cần thay đổi. |
| --disable-rule | Tắt luật, theo sau cần các tham số như tên, địa chỉ cổng để xác định luật cần thay đổi. |
| --rule-name, -n | Tham số tên luật |
| --direction {in,out}, -d {in,out} | Luật áp dụng cho gói tin đi vào|ra |
| --rule-target, -J {block,allow,bybass} | Hành động với gói tin khớp với luật :  Block: Chặn gói tin đi vào hệ thống  Allow: Cho gói tin vào hệ thống  Bybass: Cho đi qua |
| --program | Tham số đường dẫn ứng dụng.  Luật áp dụng cho ứng dụng. |

* Ví dụ:

+ Thêm luật tên TEST1 chặn tất cả gói tin UDP từ 0.0.0.0:70 đến1.1.1.1:80:

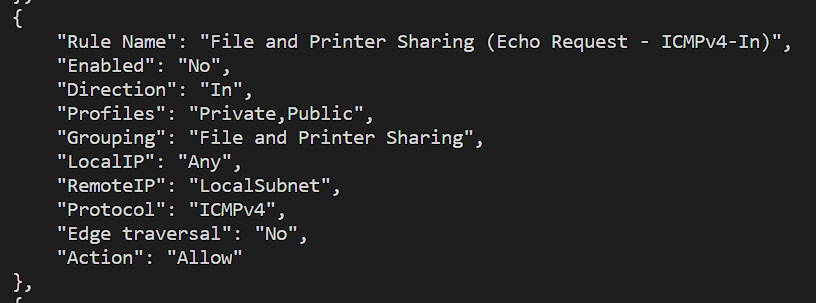
|  |
| --- |
| -A --protocol udp -d out -n TEST1 -J block --src-ip  0.0.0.0 --dst-ip 1.1.1.1 --src-port 70 --dst-port 80 |

+ Tắt luật tên block8888:

|  |
| --- |
| -n block8888 --disable |

+ Lấy danh sách luật:

|  |
| --- |
| -L |



# Tường lửa Linux

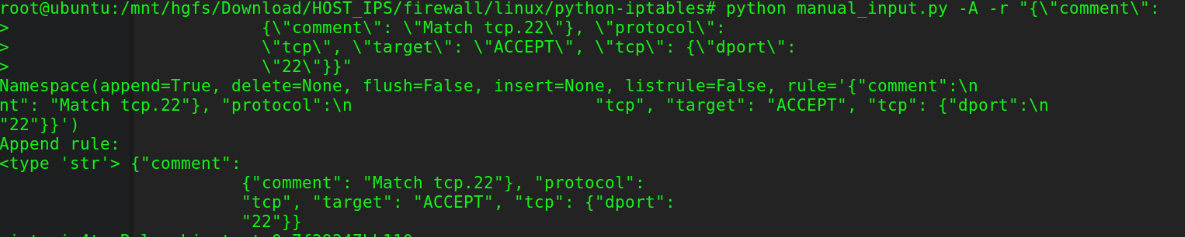
|  |  |
| --- | --- |
| Tên tệp thực thi | Chức năng |
| manual\_input | Chức năng thêm luật bằng tay vào bảng INPUT |
| manual\_output | Chức năng thêm luật bằng tay vào bảng OUTPUT |
| blacklist\_input | Chức năng thêm luật theo chính sách blacklist vào bảng INPUT |
| blacklist\_output | Chức năng thêm luật theo chính sách blacklist vào bảng OUTPUT |
| whitelist\_input | Chức năng thêm luật theo chính sách whitelist vào bảng INPUT |
| whitelist\_output | Chức năng thêm luật theo chính sách whitelist vào bảng OUTPUT |

# 10.1 Thêm luật bằng tay

Người dùng nhập vào các thông tin cho mẫu có sẵn, chương trình sẽ tạo ra luật tương ứng.

File thực thi : manual\_<input|output>

* Thêm luật vào bảng INPUT hay OUTPUT tương ứng với chặn các gói tin đầu vào và đầu ra.
* Tham số -A :Vị trí thêm luật: Thêm vào đầu bảng luật (luật có ưu tiên cao nhân, gói tin đi qua đầu tiên), luật ở cuối bảng (độ ưu tiên thấp nhất) và thêm vào 1 vị trí xác định (vị trí thứ n trong đó n nhỏ hơn số luật hiện có).
* Tham số -r : Nội dung luật dưới dạng cấu trúc dữ liệu JSON. Luật dưới dạng JSON có thể lấy từ mô-đun xem luật. Các chương trình bên tương tác với người dùng có thể bố sung các luật chặn hay cho qua với nhiều thông số gói tin phức tạp khác như trạng thái, header packet ..vv…. Ngoài ra mô-đun này có thể cho người dùng nhập nhiều luật cùng một lúc hay dùng để sao lưu và phục hồi tập luật.



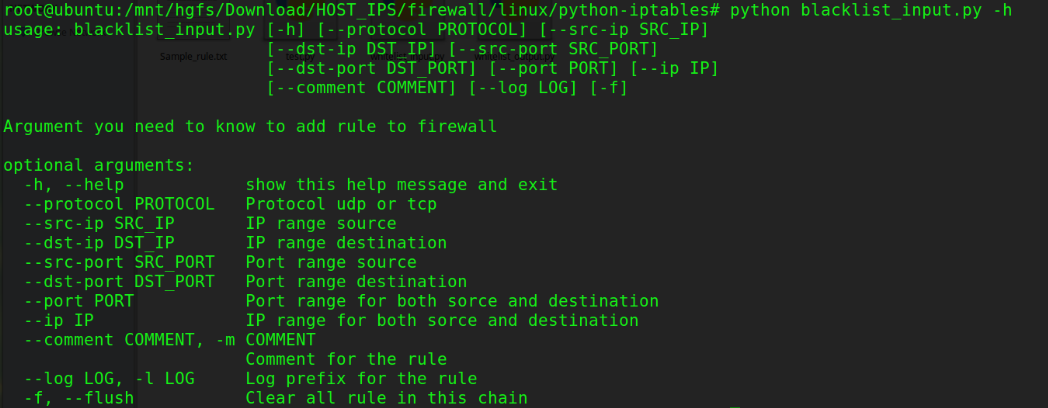
# 10.2 Chức năng thêm luật theo chính sách

- File thực thi : <whitelist|blacklist>\_<input|output>

- Tham số đầu vào:

|  |  |
| --- | --- |
| Cú pháp khi gọi script | Nội dung |
| --protocol -p | Giao thức của gói tin. |
| --src-ip | Địa chỉ nguồn của gói tin. |
| --dst-ip | Địa chỉ đích của gói tin. |
| --ip | Địa chỉ IP nằm trong địa chỉ nguồn hoặc địa chỉ đích hoặc cả 2 trong gói tin. |
| --src-port | Cổng dịch vụ nguồn của gói tin. |
| --dst-port | Cổng dịch vụ đích của gói tin. |
| --port | Cổng nguồn hoặc cổng đích hoặc cả hai. |

* Tùy chọn --comment : thêm ghi chú vào trong luật để người dùng tiện quản lý.
* Tùy chọn --log : log lại các gói tin được luật xử lý. khi chọn tham số này, người dùng sẽ bổ sung thêm 1 xâu tiền tố cho luật ghi ghi chép hoạt động trong log.



# 10.3 Khởi tạo bảng luật theo chính sách:

- File thực thi : <whitelist|blacklist>\_<input|output> -F

Whitelist : danh sách các luật cho gói tin qua, mặc định chặn.

Blacklist : danh sách các luật chặn gói tin qua, mặc định cho qua.

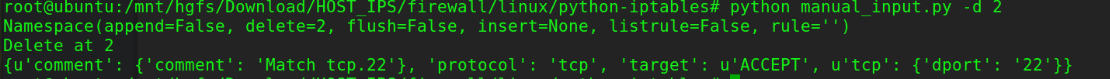
Input|Output: Các luật áp dụng cho các luồng gói tin vào|ra.

# 10.4 Xóa luật cụ thể

- File thực thi : <manual>\_<input|output>

Tham số : -d <vị trí luật cần xóa>

Input|Output: Các luật áp dụng cho các luồng gói tin vào|ra.



# 10.5 Xóa toàn bộ luật

- File thực thi : manual\_<input|output>

- Tham số : -F

Input|Output: Các luật áp dụng cho các luồng gói tin vào|ra.

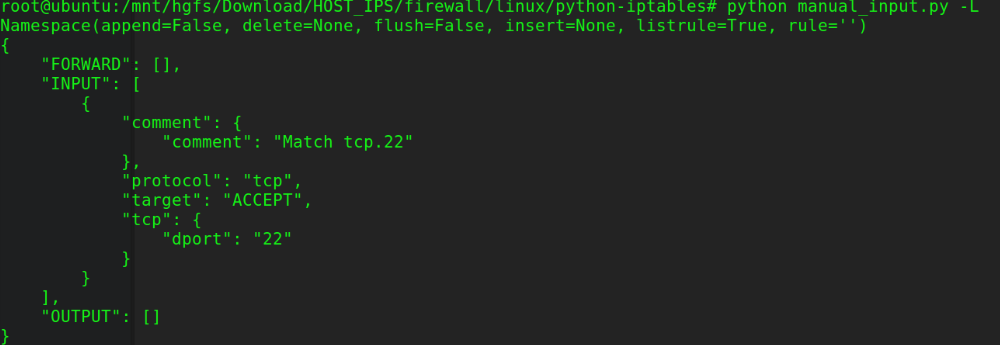


# 10.6 Lấy danh sách luật đang chạy

- File thực thi : manual\_<input|output>

- Tham số : -L

Input|Output: Các luật áp dụng cho các luồng gói tin vào|ra.



# Danh sách ứng dụng Window

* File thực thi: windowApplication
* Mô tả: Liệt kê ứng dụng đang có trong hệ điều hành Window đầu ra dạng JSON.
* Dữ liệu đầu ra:

|  |
| --- |
| {          "application\_name": "Microsoft Visual C++ 2015-2019 Redistributable (x64) - 14.20.27508",          "current\_version": "14.20.27508.1",          "publisher": "Microsoft Corporation",          "icon": "C:\\ProgramData\\Package Cache\\{7b178cda-9740-4701-a92a-f168d213b343}\\VC\_redist.x64.exe,0",          "install\_information": {              "uninstall\_string": "\"C:\\ProgramData\\Package Cache\\{7b178cda-9740-4701-a92a-f168d213b343}\\VC\_redist.x64.exe\"  /uninstall",              "size": 23726,              "install\_location": "undefined",              "install\_date": "undefined",              "install\_source": "undefined"          }      } |

|  |  |
| --- | --- |
| Tên trường dữ liệu | Ý nghĩa |
| application\_name | Tên ứng dụng |
| current\_version | Phiên bản hiện tại |
| install\_location | Vị trí cài đặt ứng dụng |
| publisher | Đơn vị phát hành ứng dụng |
| uninstall\_string | Địa chỉ tệp gỡ bỏ ứng dụng |
| install\_date | Thời điểm cài ứng dụng |

# Danh sách ứng dụng Linux

* File thực thi: linux\_app
* Mô tả: Liệt kê ứng dụng đang có trong hệ điều hành Linux với đầu ra dạng JSON.
* Dữ liệu đầu ra:

|  |
| --- |
| {"description": "XSL stylesheets for the yelp help browser", "application\_name": "yelp-xsl", "architect": "all", "version": "3.18.1-1"}  {"description": "Display graphical dialog boxes from shell scripts", "application\_name": "zenity", "architect": "amd64", "version": "3.18.1.1-1ubuntu2"}  {"description": "Display graphical dialog boxes from shell scripts (common files)", "application\_name": "zenity-common", "architect": "all", "version": "3.18.1.1-1ubuntu2"}  {"description": "Archiver for .zip files", "application\_name": "zip", "architect": "amd64", "version": "3.0-11"}  {"description": "compression library - runtime", "application\_name": "zlib1g:amd64", "architect": "amd64", "version": "1:1.2.8.dfsg-2ubuntu4.3"} |

|  |  |
| --- | --- |
| Tên khóa | Giá trị |
| application\_name | Tên ứng dụng |
| version | Phiên bản hiện tại của ứng dụng |
| description | Mô tả ứng dụng |
| architect | Kiến trúc ứng dụng |

# Kiểm tra cập nhật ứng dụng Linux

* File thực thi: linux\_app –upgradeable
* Mô tả: Xuất ra danh sách các ứng dụng linux có thể cập nhật với phiên bản mới nhất

# Kiểm tra lỗ hổng bảo mật

* File thực thi: getcve -u <url ứng dụng> -v <phiên bản ứng dụng cần kiểm tra>
* Dữ liệu đầu ra:

|  |
| --- |
| {          "index": "3",          "cve\_id": "CVE-2019-5787",          "cve\_url": "https://www.cvedetails.com/cve/CVE-2019-5787/",          "cwe\_id": "416",          "cwe\_url": "https://www.cvedetails.co//www.cvedetails.com/cwe-details/416/cwe.html",          "vulnerability\_type": "",          "publish\_date": "2019-05-23",          "update\_date": "2019-06-28",          "cve\_score": "9.3",          "access": "Remote",          "complexcity": "Medium",          "cve\_summary": "Use-after-garbage-collection in Blink in Google Chrome prior to 73.0.3683.75 allowed a remote attacker to potentially exploit heap corruption via a crafted HTML page."      } |

|  |  |
| --- | --- |
| Tên trường | Ý nghĩa |
| cve\_id | Mã id của lỗ hổng CVE |
| cve\_url | Đường dẫn đến web chi tiết lỗ hổng |
| vulnerability\_type | Chủng loại lỗ hổng bảo mật |
| publish date | Ngày công bố lỗ hổng |
| cve\_score | Điểm đánh giá múc độ nguy hiểm của lỗ hổng theo thang 10 |
| access | Khả năng truy cập hệ thống nếu sử dụng lỗ hổng |
| cve\_summary | Mô tả lỗ hổng |