

Spécialité : SSIR-2D

Rapport de Projet de owasp



Realisé par : Nasri wissem

Encadrant académique : oussema riahi

Année universitaire : 2023 2024

Table des matières :

1.Modélisation des menaces

 1.1.Modélisation des menaces avec « Microsoft Threat Modeling Tool 2016 »

2.Tests statiques de sécurité des applications (SAST)

 2.1.Analyse du code c à l'aide de l'outil « flaw finder »

 2.2.Analyse du code Python à l'aide de l'outil « Bandit »

3.Tests dynamiques de la sécurité des applications (DAST)

 3.1Analyse du code de l'application « ghost » de la machine vulnérable « OWASP_Broken_Web_Apps_VM_1.2 » à l'aide de l'outil « Zaproxy »

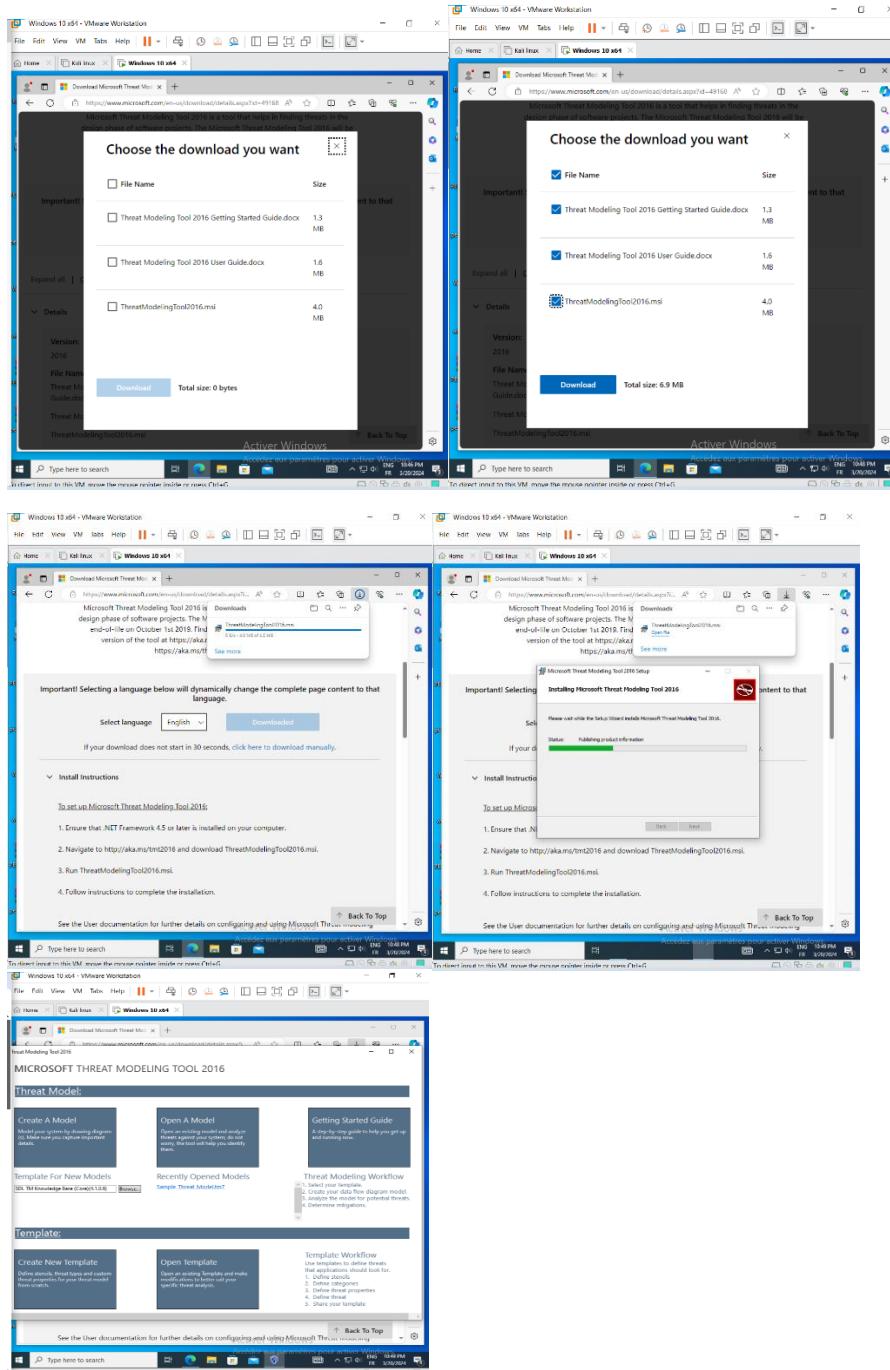
4.Analyses (SAST vs DAST)

1. Modélisation des menaces

Modélisation des menaces Le modèle de menace est un processus par lequel des menaces potentielles, telles que les vulnérabilités structurelles peuvent être identifiées, énumérées et classées par ordre de priorité. Lorsqu'elle est effectuée correctement, la modélisation des menaces peut fournir une ligne de vue claire sur un projet logiciel, aidant à justifier les efforts de sécurité. Le processus de modélisation des menaces aide une organisation à documenter les menaces de sécurité connues pour une application et à prendre des décisions rationnelles sur la manière de les traiter

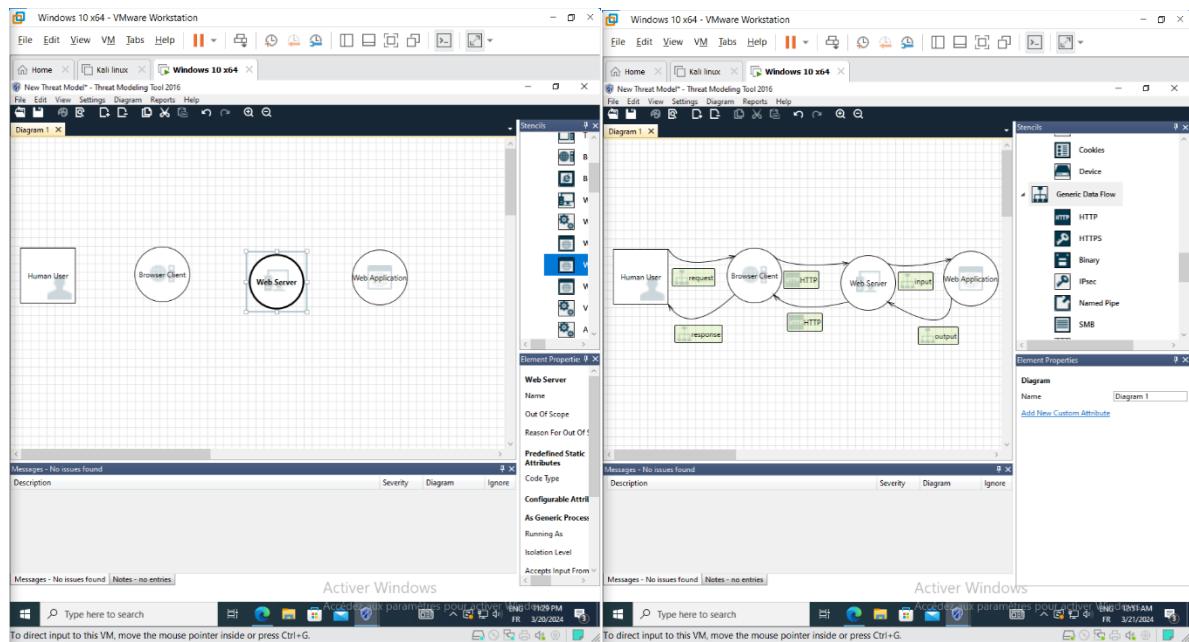
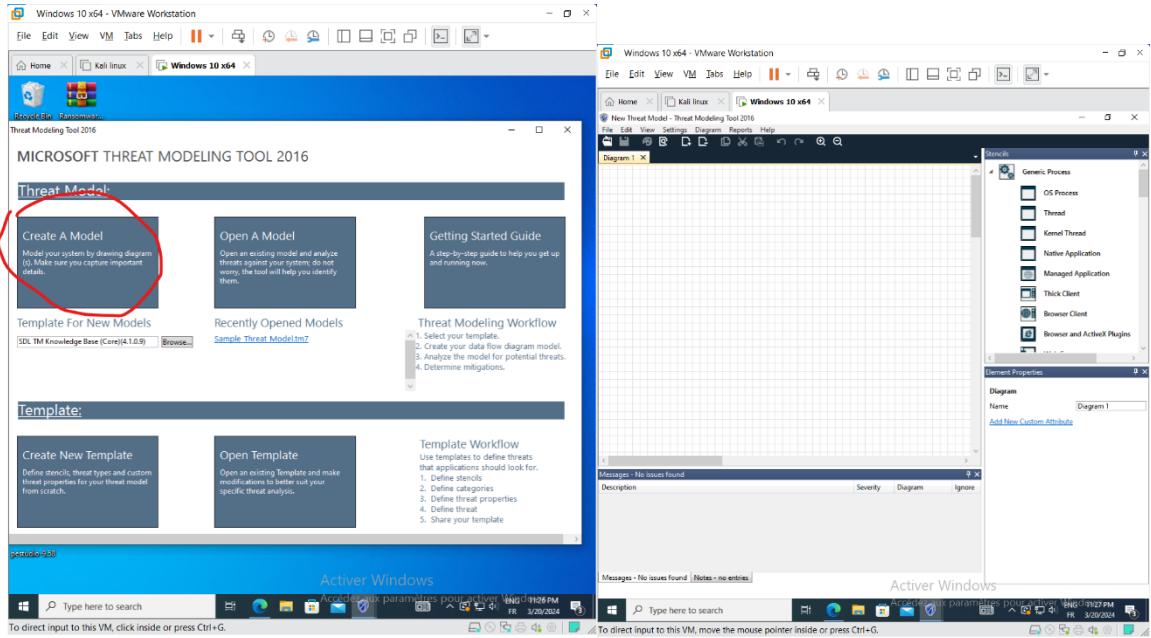
1.1. Modélisation des menaces avec « Microsoft Threat Modeling Tool 2016 »

1- Télécharger l'outil « Microsoft Threat Modeling Tool 2016 » depuis son source officiel

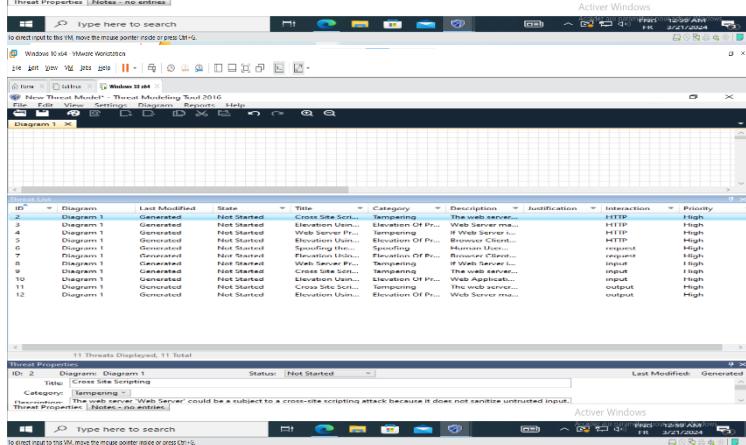
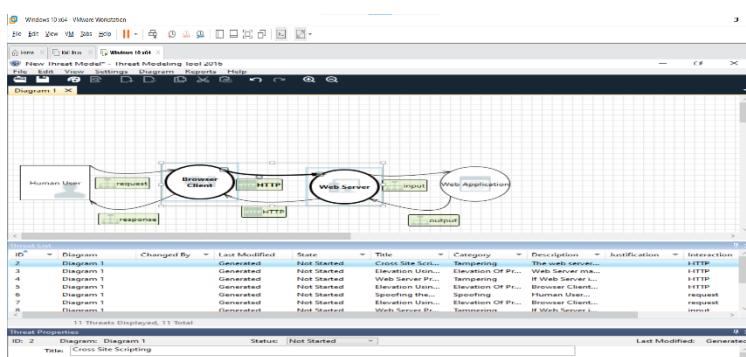
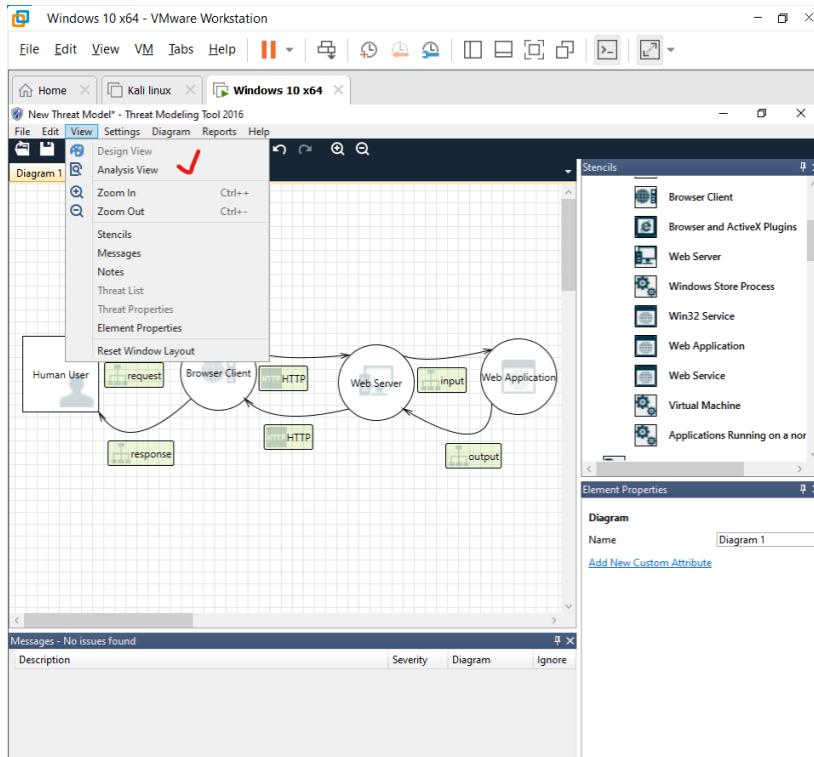


3- Reproduire l'architecture suivante et créer sa modèle de menace :

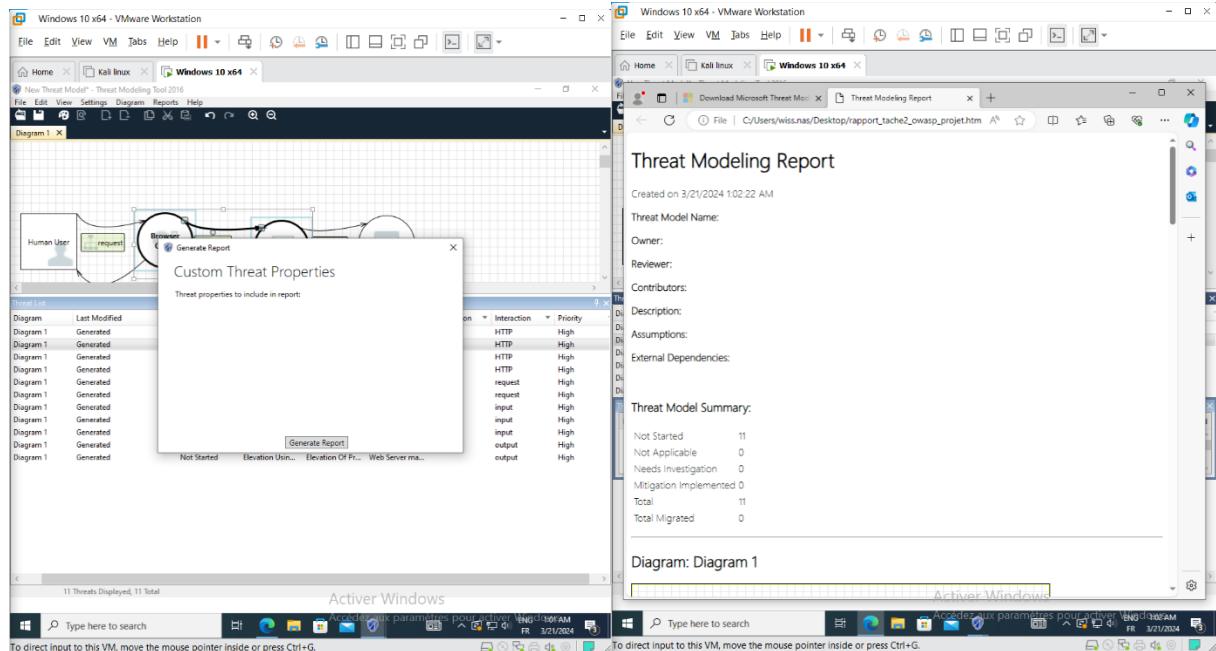
Etapes pour reproduire l'architecture



4- l'outil Microsoft Threat Modeling Tool possède deux vues : Vue de conception et Vue d'analyse, utiliser la vue d'analyse pour déterminer les vulnérabilités liée à cette architecture



Threat modeling report :



Voir le lien qui comporte se rapport du 2eme partie

les vulnérabilités liée à cette architecture :

interaction http

1.elevation using impersonation

2.Cross site scripting

interaction http

3.web server process memory tampered

4.elevation using impersonation

Interaction input

5. webserver process memory tampered

6.cross site scripting

7.elevation using impersonation

Interaction output

8.cross site scripting

9.elevation using impersonation

Interaction request

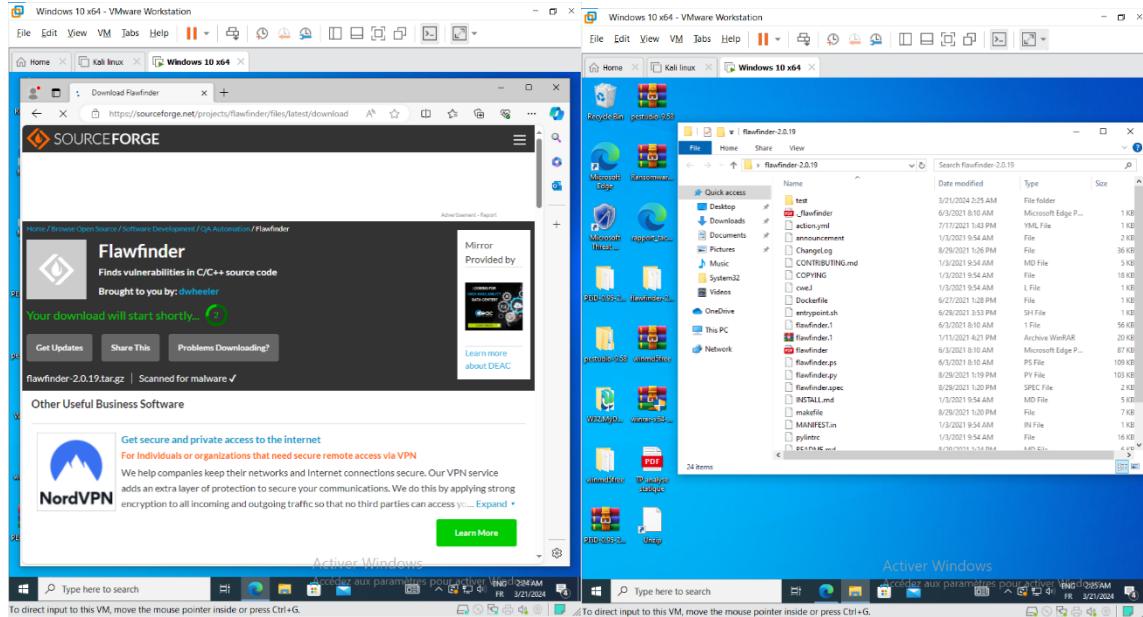
10.spoofing using impersonation

11.elevation using impersonation

2. Tests statiques de sécurité des applications (SAST)

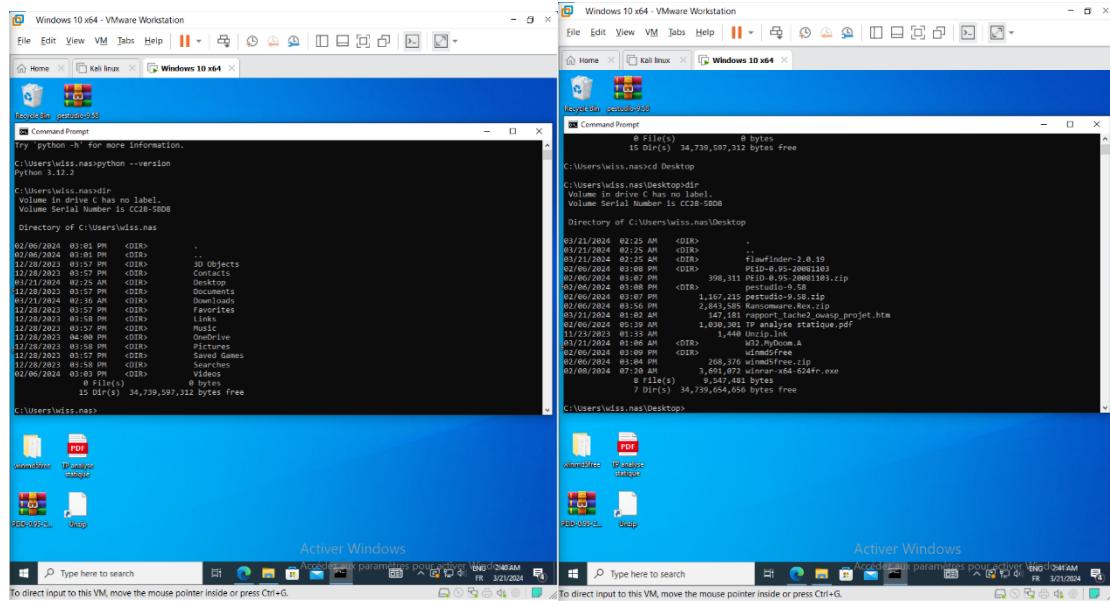
2.1. Analyse du code c à l'aide de l'outil « flaw finder »

Exercie1 :



1. To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

2.



```

Windows 10 x64 - VMware Workstation
File Edit View VM Tabs Help ||| Home X Kali Linux X Windows 10 x64 X
Recycle Bin  pebble0-033
Command Prompt
C:\Users\wliss\Desktop\FlawFinder-2.0.19>dir
Volume in drive C has no label
Volume Serial Number is CCED-B50B

Directory of C:\Users\wliss\Desktop\FlawFinder-2.0.19
03/21/2024 02:25 AM <DIR> .
03/21/2024 02:25 AM <DIR> ..
06/03/2021 01:43 AM 176 FlawFinder.pdf
07/17/2021 01:43 AM 408 action.yml
01/03/2021 10:54 AM 1,364 announcement
03/21/2024 02:25 AM 534 CHANGELOG
01/03/2021 10:54 AM 4,981 CONTRIBUTING.md
01/03/2021 10:54 AM 18,087 COPYING
01/03/2021 10:54 AM 352 Dockerfile
06/27/2021 01:28 PM 310 Dockerfile
06/29/2021 03:53 PM 217 entrypoint.sh
06/03/2021 08:10 AM 57,258 FlawFinder-1
01/03/2021 08:10 AM 19,193 FlawFinder-1.gz
06/03/2021 08:10 AM 88,149 FlawFinder.pdf
06/03/2021 08:10 AM 110,804 FlawFinder.ps
06/03/2021 08:10 AM 103 FlawFinder.spec
06/29/2021 01:20 PM 1,249 FlawFinder.spec
01/03/2021 10:54 AM 4,914 INSTALL.md
06/03/2021 08:10 AM 6,000 LICENSE
01/03/2021 10:54 AM 247 MANIFEST.in
01/03/2021 10:54 AM 15,537 pylintrc
06/03/2021 08:10 AM 5,000 README
06/31/2021 12:11 PM 1,671 release-process.md
01/03/2021 10:54 AM 200 setup.cfg

Windows 10 x64 - VMware Workstation
File Edit View VM Tabs Help ||| Home X Kali Linux X Windows 10 x64 X
Recycle Bin  pebble0-033
Command Prompt
3 Dr(s) 34,739,618,272 bytes free
C:\Users\wliss\Desktop\FlawFinder-2.0.19>python flawfinder.py testtest
FlawFinder version 2.0.19, (C) 2001-2019 David A. Wheeler.
Number of rules (primarily dangerous function names) in C/C++ ruleset: 222
Examining testtest.c

FINAL RESULTS:
testtest.c:32: [S] (Buffer) gets:
    Doesn't check for buffer overflow (CWE-120, CWE-20). Use fgets() instead.
testtest.c:68: [S] (Buffer) strncat:
    Easily used incorrectly (e.g., incorrectly computing the correct maximum size to add). [McDonald] (CWE-120) Risk is high; the length parameter appears to be a constant, instead of computing the number of characters left.
testtest.c:64: [S] (Buffer) MultiByteToWideChar:
    Requires maximum length in CHARACTERS, not bytes (CWE-120). Risk is high, it appears that the size is given as bytes, but the function requires size as characters.
testtest.c:64: [S] (Buffer) MultiByteToWideChar:
    Requires maximum length in CHARACTERS, not bytes (CWE-120). Risk is high, it appears that the size is given as bytes, but the function requires size as characters.

Windows 10 x64 - VMware Workstation
File Edit View VM Tabs Help ||| Home X Kali Linux X Windows 10 x64 X
Recycle Bin  pebble0-033
Command Prompt
Type here to search  Accéder aux paramètres pour ce poste de travail  ENG 12:42AM  FR 3/21/2024
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Windows 10 x64 - VMware Workstation
File Edit View VM Tabs Help ||| Home X Kali Linux X Windows 10 x64 X
Recycle Bin  pebble0-033
Command Prompt
Type here to search  Accéder aux paramètres pour ce poste de travail  ENG 12:42AM  FR 3/21/2024
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Windows 10 x64 - VMware Workstation
File Edit View VM Tabs Help ||| Home X Kali Linux X Windows 10 x64 X
Recycle Bin  pebble0-033
Command Prompt
ANALYSIS SUMMARY:
Hits = 39
Lines analyzed = 125 in approximately 0.33 seconds (381 lines/second)
Physical Source Lines of Code (SLOC) = 86
Hits/SLOClevel = [0+] 10 [1+] 10 [2+] 4 [3+] 10 [4+] 10 [5+] 7
Hits/SLOClevel = [0+] 55 [1+] 55 [2+] 55 [3+] 55 [4+] 55 [5+] 7
Hits/SLOClevel = [0+] 639,535 [1+] 453,488 [2+] 348,837 [3+] 244,186 [4+] 197,674 [5+] 81,3953
Suppressed hits = 2 (use --neverignore to show them)
Minimum risk level = 1

Not every hit is necessarily a security vulnerability,
you can ignore a report by adding a comment in this form:
// ignore
// ignore="It's a false positive"
Make "sure" it's a false positive!
You can use the option --neverignore to show these.

There may be other security vulnerabilities; review your code!
See 'Secure Programming HOWTO'
(http://duheeler.com/secure-programs) for more information.

C:\Users\wliss\Desktop\FlawFinder-2.0.19>

Windows 10 x64 - VMware Workstation
File Edit View VM Tabs Help ||| Home X Kali Linux X Windows 10 x64 X
Recycle Bin  pebble0-033
Command Prompt
Type here to search  Accéder aux paramètres pour ce poste de travail  ENG 12:42AM  FR 3/21/2024
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

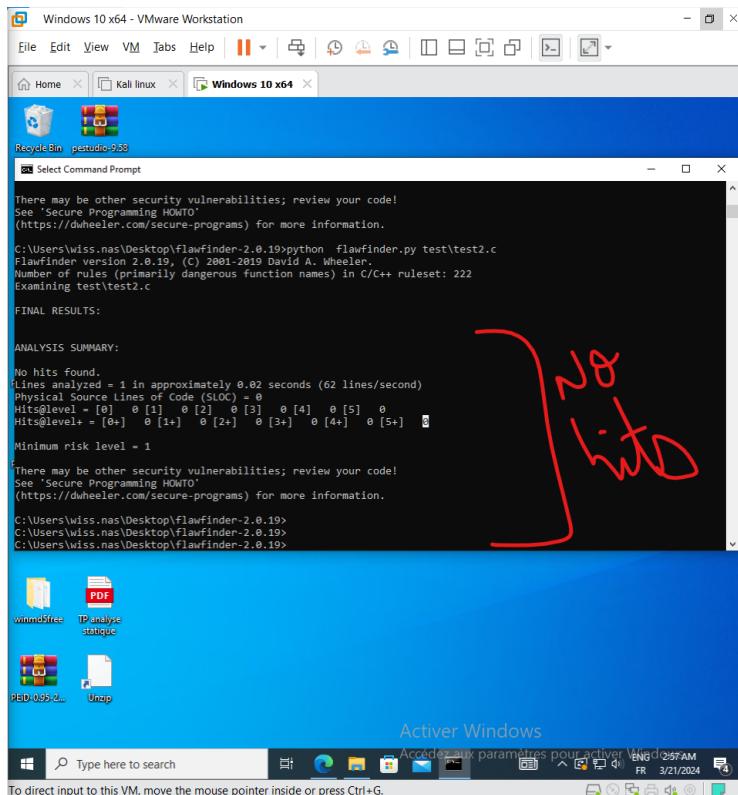
→ Hits=39

```

3.

Il existe 39 vulnérabilités au niveau du code C du fichier test.c → le code est vulnérable

4.



→ NO hits found

Le code c du fichier test2.c n'est pas vulnerable

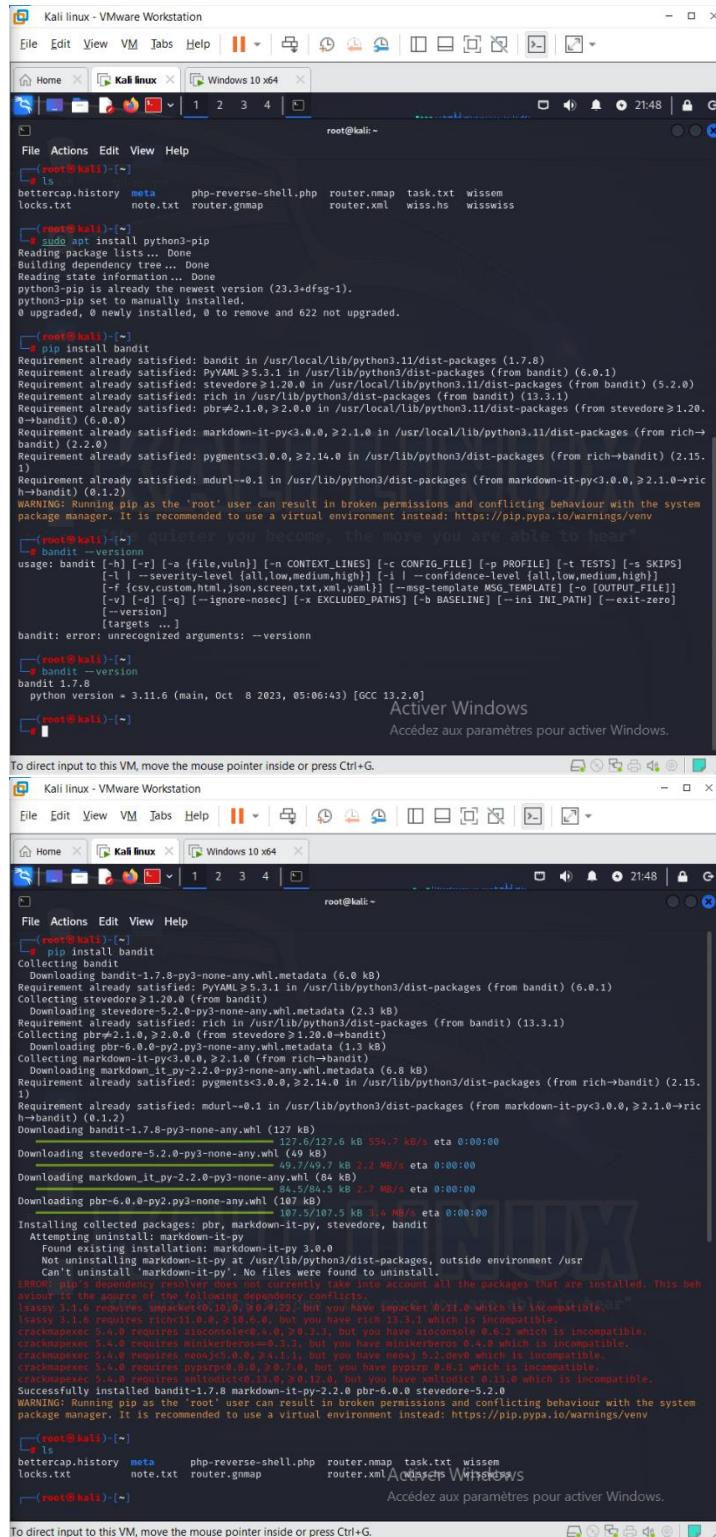
5.

Flawfinder utilise une méthode de recherche de motifs pour analyser le code source à la recherche de constructions potentiellement dangereuses ou sujettes à des vulnérabilités de sécurité. Il recherche des chaînes de caractères correspondant à des appels de fonctions ou à des motifs de code qui sont souvent associés à des problèmes de sécurité connus, tels que les dépassesments de tampon, les vulnérabilités de format de chaîne, etc. Il repose sur une base de données de vulnérabilités connues pour identifier ces motifs.

2.2.Analyse du code Python à l'aide de l'outil « Bandit »

Bandit est un outil open source d'analyse statique de code pour Python, conçu pour détecter les vulnérabilités de sécurité potentielles dans les applications Python. Son objectif principal est d'identifier les pratiques de programmation dangereuses qui pourraient conduire à des failles de sécurité, en se concentrant principalement sur les erreurs liées à l'exécution de commandes potentiellement dangereuses, à l'utilisation de fonctions ou de modules non sécurisés, et à d'autres vulnérabilités courantes.

L'installation de l'outil <<bandit>> en utilisant la commande : pip install bandit



```

root@kali:~# pip install bandit
Requirement already satisfied: bandit in /usr/local/lib/python3.11/dist-packages (1.7.8)
Requirement already satisfied: PyYAML>=5.3.1 in /usr/lib/python3/dist-packages (from bandit) (6.0.1)
Requirement already satisfied: stevedore>=1.20.0 in /usr/local/lib/python3.11/dist-packages (from bandit) (5.2.0)
Requirement already satisfied: rich in /usr/lib/python3/dist-packages (from bandit) (13.3.1)
Requirement already satisfied: pbr>=2.1.0, >=2.0.0 in /usr/local/lib/python3.11/dist-packages (from stevedore>=1.20.0)
Requirement already satisfied: markdown-it-py<3.0.0, >=2.1.0 in /usr/local/lib/python3.11/dist-packages (from rich->bandit) (2.2.0)
Requirement already satisfied: pygments<3.0.0, >=2.14.0 in /usr/lib/python3/dist-packages (from rich->bandit) (2.15.1)
Requirement already satisfied: mdurl=>0.1 in /usr/lib/python3/dist-packages (from markdown-it-py<3.0.0, >=2.1.0->rich->bandit) (0.1.2)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system
package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv

root@kali:~# bandit --version
bandit: error: unrecognized arguments: --version

root@kali:~# bandit --version
bandit 1.7.8
python version = 3.11.6 (main, Oct 8 2023, 05:06:43) [GCC 13.2.0]

```

Activer Windows
Accédez aux paramètres pour activer Windows.

```

root@kali:~# pip install bandit
Collecting bandit
  Downloading bandit-1.7.8-py3-none-any.whl.metadata (6.0 kB)
Requirement already satisfied: PyYAML>=5.3.1 in /usr/lib/python3/dist-packages (from bandit) (6.0.1)
Collecting stevedore>=1.20.0 (from bandit)
  Downloading stevedore-1.20.0-py3-none-any.whl.metadata (2.3 kB)
Requirement already satisfied: rich in /usr/lib/python3/dist-packages (from bandit) (13.3.1)
Collecting pbr>=2.1.0, >=2.0.0 (from stevedore>=1.20.0->bandit)
  Downloading pbr-6.0.0-py2.py3-none-any.whl.metadata (1.3 kB)
Collecting markdown-it-py<3.0.0, >=2.1.0 (from rich->bandit)
  Downloading markdown_it_py-2.2.0-py3-none-any.whl.metadata (6.8 kB)
Requirement already satisfied: pygments<3.0.0, >=2.14.0 in /usr/lib/python3/dist-packages (from rich->bandit) (2.15.1)
Requirement already satisfied: mdurl=>0.1 in /usr/lib/python3/dist-packages (from markdown-it-py<3.0.0, >=2.1.0->rich->bandit) (0.1.2)
  Downloading bandit-1.7.8-py3-none-any.whl (127 kB)
    127.6/127.6 kB 554.7 kB/s eta 0:00:00
  Downloading stevedore-5.2.0-py3-none-any.whl (44 kB)
    49.7/49.7 kB 2.2 MB/s eta 0:00:00
  Downloading markdown_it_py-2.2.0-py3-none-any.whl (94 kB)
    84.5/84.5 kB 2.7 MB/s eta 0:00:00
  Downloading pbr-6.0.0-py2.py3-none-any.whl (107 kB)
    107.5/107.5 kB 3.4 MB/s eta 0:00:00
Installing collected packages: pbr, markdown-it-py, stevedore, bandit
  Attempting download of: bandit-1.7.8-py3-none-any.whl
    Found existing installation: bandit-1.7.8-py3.0.0
      Not uninstalling markdown-it-py at /usr/lib/python3/dist-packages, outside environment /usr
        Can't uninstall 'markdown-it-py'. No files were found to uninstall.
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts.
  |saasy 3.1.0 requires impacket<0.9.0,>0.9.22, but you have impacket 0.23.0 which is incompatible.
  |saasy 3.1.0 requires rich<1.0.0,>0.13.0, but you have rich 1.0.1 which is incompatible.
  |crackmapexec 5.4.0 requires enum34<5.0.0,>4.1.1, but you have enum34 5.2.0 which is incompatible.
  |crackmapexec 5.4.0 requires minikerberos==0.1.3, but you have minikerberos 0.4.0 which is incompatible.
  |crackmapexec 5.4.0 requires pypyserde<0.0.0,>0.7.0, but you have pypyserde 0.8.3 which is incompatible.
  |crackmapexec 5.4.0 requires xldictio<0.10.0,>0.12.0, but you have xldictio 0.11.0 which is incompatible.
Successfully installed bandit-1.7.8 markdown-it-py-2.2.0 pbr-6.0.0 stevedore-5.2.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system
package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv

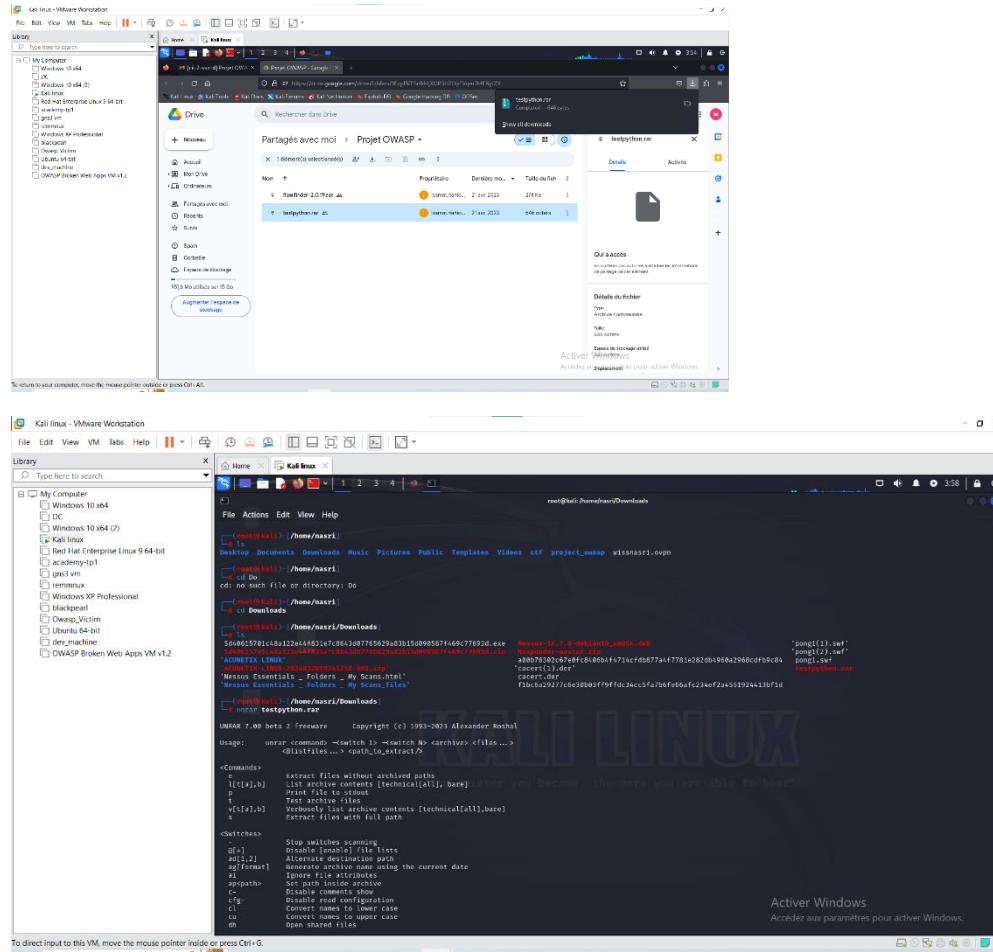
root@kali:~# ls
bettercap.history metasploit modules note.txt router.nmap task.txt wissens
locks.txt note.txt router.gnmap router.xml wiss.h5 wississ

```

Activer Windows
Accédez aux paramètres pour activer Windows.

-Télécharger et copier le « fichier testpython.py » dans la machine kali linux

- Télécharger et copier le « fichier testpython2.py » dans la machine kali linux



Evaluation de la sécurité des codes « testpython.py », « testpython2.py » en utilisant la commande **bandit testpython.py** et **bandit testpython2.py** et la retour de l'analyse effectué

A screenshot of a Kali Linux desktop environment. The terminal window shows the following command and its output:

```
root@kali:~/Downloads$ unrar e testpython.rar
UNRAR 7.00 beta 2 Freeware Copyright (c) 1993-2023 Alexander Roshal

Extracting from testpython.rar
Extracting testpython.py          OK
Extracting testpython2.py        OK
All OK

root@kali:~/Downloads$ ls
5d460157e4ca8a22e4fa81e7c64a208765629a83b150d98587f469c77693d.exe
ACMEUTIX_LNX_1.00-2024030101252-001.sqf
ACMEUTIX_LNX_1.00-2024030101252-001.zip
'Acmeus Essentials - Folders - My Scans.html'
'Acmeus Essentials - Folders - My Scan_files'

root@kali:~/Downloads$
```

The desktop background features a large "KALI LINUX" watermark with the tagline "the quieter you become, the more you are able to hear".

The screenshot displays two side-by-side VMware Workstation windows. The top window is titled 'Kali linux - VMware Workstation' and shows a terminal session on the Kali Linux VM. The user has run a security audit command, likely 'wpscan', against a target website. The output shows several findings, including SQL injection vulnerabilities ('B611:django_rawsql_used') and TLS/SSL protocol version issues ('B502:ssl_with_bad_version'). The results are color-coded by severity: red for critical, orange for high, and green for low. The bottom window is titled 'Kali linux - VMware Workstation' and shows a terminal session on the Windows 10 VM. It displays similar audit results, with findings such as 'ssl.wrap_socket call with insecure SSL/TLS protocol version identified, security issue' and 'ssl_context call with insecure SSL/TLS protocol version identified, security issue'. Both windows have a watermark in the center reading 'KALI LINUX' and the tagline 'the quieter you become, the more you are able to hear'. At the bottom right of each window, there is an 'Activer Windows' button.

The screenshot shows two windows of the Kali Linux terminal within a VMware Workstation interface. Both windows have the title 'Kali Linux' and are running as root. The top window displays a script named 'testpython2.py' which performs SSL/TLS protocol analysis. It includes sections for test results, security issues (e.g., CVE-1998-0001), and specific function calls like 'ssl.wrap_socket'. The bottom window also shows a similar script, 'testpython2.py', with some additional comments and a different file path. Both windows show the Kali Linux desktop environment in the background, featuring a large 'KALI LINUX' watermark. The VMware toolbar at the top includes icons for file operations, windows management, and system status.

Kali linux - VMware Workstation

File Edit View VM Tabs Help |

Library Type here to search

My Computer

- Windows 10 x64
- DC
- Windows 10 x64 (2)
- Kali Linux**
- Red Hat Enterprise Linux 9 64-bit
- academy-tp1
- gn3 vm
- remnux
- Windows XP Professional
- blackpearl
- Owasp_Victim
- Ubuntu 64-bit
- dev_machine
- OWASP Broken Web Apps VM v1.2

Home Kali Linux

```
root@kali:~/home/narsi/Downloads
File Actions Edit View Help
B050:ssl_with_no_version SSLContext call with insecure SSL/TLS protocol version identified, security issue.
Severity: Low Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:22:10
    ssl.wrap_socket(ssl_version=ssl.PROTOCOL_TLSv1_1)
    SSLContext(method=ssl.TLSv1_1_METHOD)

>> Issue: [B050:ssl_with_no_version] SSLContext call with insecure SSL/TLS protocol version identified, security issue.
Severity: High Confidence: High
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:23:10
    ssl.wrap_socket(ssl_version=ssl.PROTOCOL_TLSv1_1)
    SSLContext(method=ssl.TLSv1_1_METHOD)

>> Issue: [B050:ssl_with_no_version] Function call with insecure SSL/TLS protocol identified, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:24:10
    herp_derp(ssl_version=ssl.PROTOCOL_TLSv1_1)
    herp_derp(method=ssl.TLSv1_1_METHOD)

>> Issue: [B050:ssl_with_no_version] Function call with insecure SSL/TLS protocol identified, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:25:10
    herp_derp(ssl_version=ssl.PROTOCOL_TLSv1_1)
    herp_derp(method=ssl.TLSv1_1_METHOD)

>> Issue: [B050:ssl_with_no_version] Function call with no SSL/TLS protocol version specified, the default SSLv23 could be insecure, possible security issue.
Severity: Low Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:26:10
    ssl.wrap_socket()

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:27:10
    def open_ssl_socket():

Activer Windows
Accédez aux paramètres pour activer Windows.
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Kali Linux - VMware Workstation

File Edit View VM Tabs Help |

Library Type here to search

My Computer

- Windows 10 x64
- DC
- Windows 10 x64 (2)
- Kali Linux**
- Red Hat Enterprise Linux 9 64-bit
- academy-tp1
- gn3 vm
- remnux
- Windows XP Professional
- blackpearl
- Owasp_Victim
- Ubuntu 64-bit
- dev_machine
- OWASP Broken Web Apps VM v1.2

Home Kali Linux

```
root@kali:~/home/narsi/Downloads
File Actions Edit View Help
B050:ssl_with_no_version SSLContext call with insecure SSL/TLS protocol version identified, security issue.
Severity: Low Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:30:10
    ssl.wrap_socket()

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:31:10
    def open_ssl_socket():

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:32:10
    def open_ssl_socket(version=ssl.PROTOCOL_SSLv2):

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:33:10
    pass

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:34:10
    def open_ssl_socket(version=SSL.SSLv2_METHOD):
    pass

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:35:10
    def open_ssl_socket(version=SSL.TLSv1_1_METHOD):
    pass

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:36:10
    def open_ssl_socket(version=SSL.TLSv1_1_METHOD):
    pass

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:37:10
    def open_ssl_socket(version=SSL.TLSv1_1_METHOD):
    pass

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:38:10
    def open_ssl_socket(version=SSL.TLSv1_1_METHOD):
    pass

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:39:10
    def open_ssl_socket(version=SSL.TLSv1_1_METHOD):
    pass

>> Issue: [B050:ssl_with_no_version] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium Confidence: Medium
CWE: CWE-327 (https://cwe.mitre.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b050_ssl_with_no_version.html
Location: ./testpythond2.py:40:10
    def open_ssl_socket(version=SSL.TLSv1_1_METHOD):
    pass

Activer Windows
Accédez aux paramètres pour activer Windows.
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

The screenshot shows a Kali Linux VM in VMware Workstation. The terminal window displays the output of a security audit script, likely OWASP ZAP, against a local host. The output includes several SSL/TLS protocol version issues:

```
root@kali:~/Downloads# ./testpython2.py:35:0
34
35     def open_ssl_socket(version=SSL.SSLv2_METHOD):
36         pass
37

>> Issue: [B03ssl_with_bad_defaults] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium    Confidence: Medium
CWE: CWE-327 (https://www.wireshark.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b03_ssl_with_bad_defaults.html
Location: ./testpython2.py:38:0
38
39     def open_ssl_socket(version=SSL.SSLv23_METHOD):
40         pass
41

>> Issue: [B03ssl_with_bad_defaults] Function definition identified with insecure SSL/TLS protocol version by default, possible security issue.
Severity: Medium    Confidence: Medium
CWE: CWE-327 (https://www.wireshark.org/data/definitions/327.html)
More Info: https://bandit.readthedocs.io/en/1.7.8/plugins/b03_ssl_with_bad_defaults.html
Location: ./testpython2.py:41:0
40
41     def open_ssl_socket(version=SSL.TLSv1_METHOD):
42         pass
43

Code scanned:
    Total lines of code: 31
    Total lines skipped (mosec): 0

Run metrics:
    Total issues (by severity):
        Undefined: 0
        Low: 1
        Medium: 13
        High: 9
    Total issues (by confidence):
        Undefined: 0
        Low: 1
        Medium: 14
        High: 9
    Files skipped (0):
```

The terminal prompt at the bottom is `[root@kali:~/Downloads]`.

Le fichier « testpython2.py » est plus vulnérable que l'autre fichier « testpython.py »

« testpython2.py »

```
Code scanned:
  Total lines of code: 31
  Total lines skipped (#nosec): 0

Run metrics:
  Total issues (by severity):
    Undefined: 0
    Low: 1
    Medium: 13
    High: 9
  Total issues (by confidence):
    Undefined: 0
    Low: 0
    Medium: 14
    High: 9

Files skipped (0):

root@kali:~/home/nasri/Downloads$
```

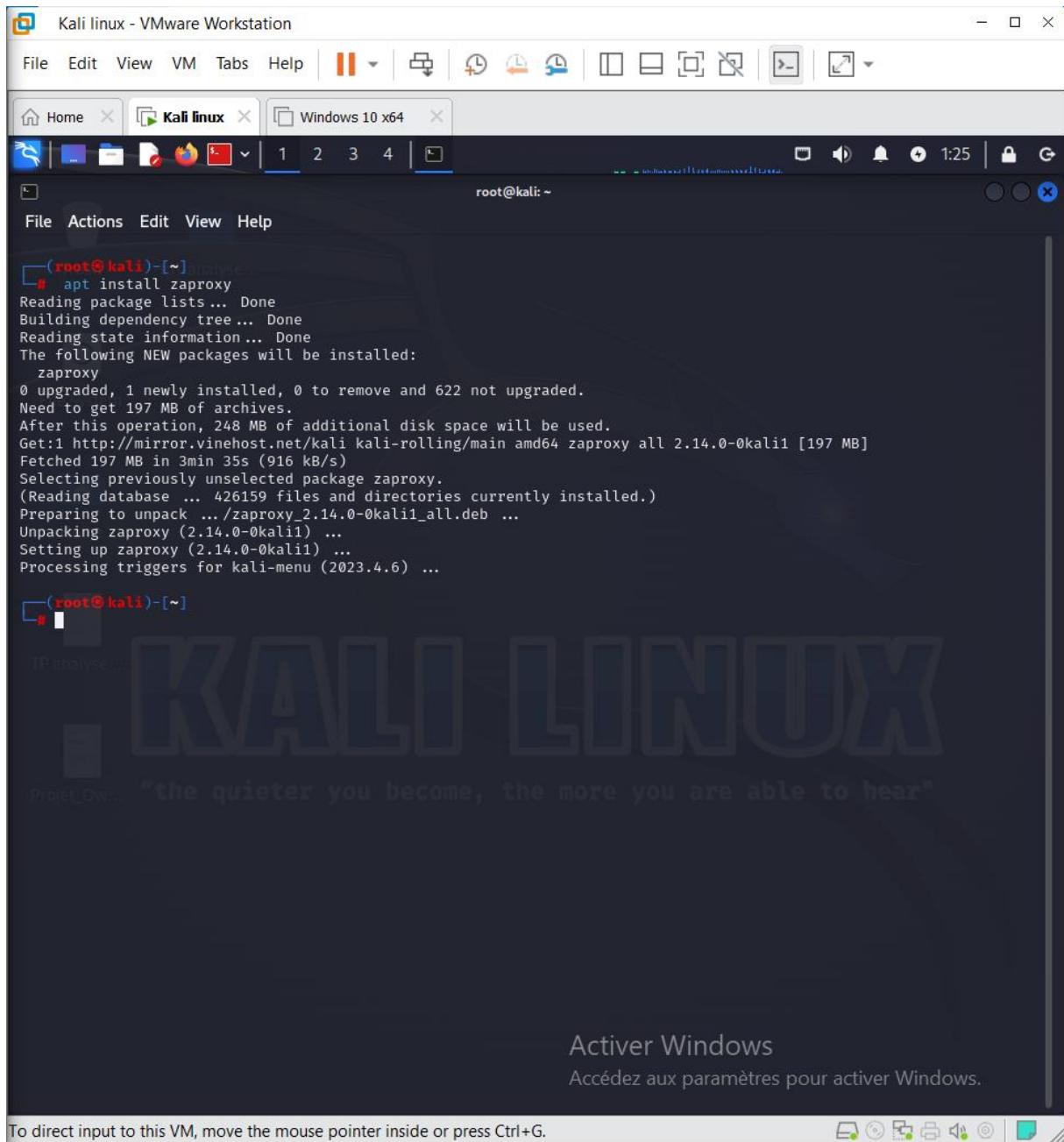
« testpython.py »

```
Code scanned:  
    Total lines of code: 10  
    Total lines skipped (#nosec): 0  
  
Run metrics:  
    Total issues (by severity):  
        Undefined: 0  
        Low: 0  
        Medium: 4  
        High: 0  
    Total issues (by confidence):  
        Undefined: 0  
        Low: 0  
        Medium: 4  
        High: 0  
Files skipped (@):
```

3. Tests dynamiques de la sécurité des applications (DAST)

3.1 Analyse du code de l'application « ghost » de la machine vulnérable « OWASP_Broken_Web_Apps_VM_1.2 » à l'aide de l'outil « Zaproxy »

1. installer zaproxy

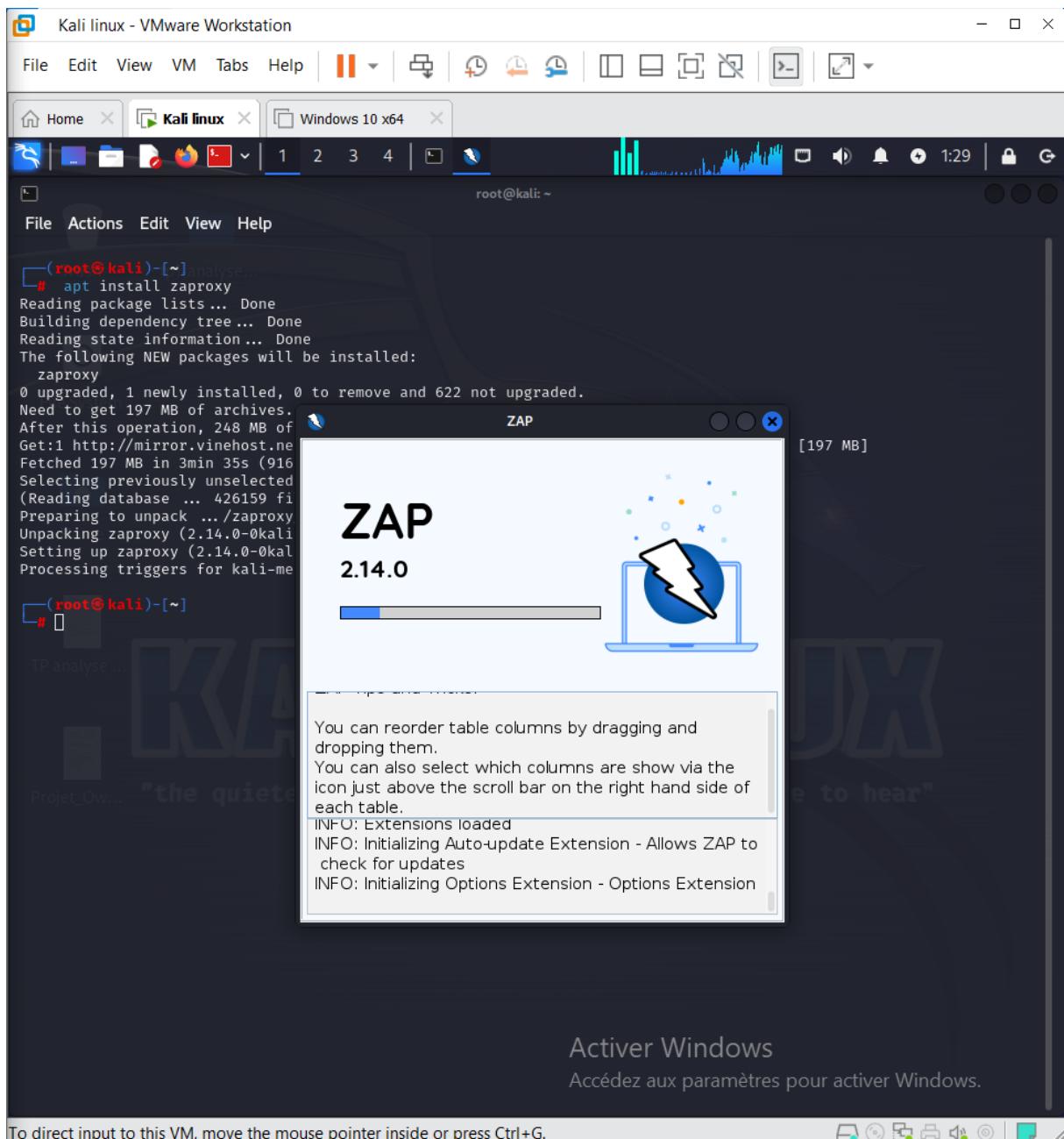


```
(root@kali)-[~]
# apt install zaproxy
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  zaproxy
0 upgraded, 1 newly installed, 0 to remove and 622 not upgraded.
Need to get 197 MB of additional disk space will be used.
After this operation, 248 MB of additional disk space will be used.
Get:1 http://mirror.vinehost.net/kali kali-rolling/main amd64 zaproxy all 2.14.0-0kali1 [197 MB]
Fetched 197 MB in 3min 35s (916 kB/s)
Selecting previously unselected package zaproxy.
(Reading database ... 426159 files and directories currently installed.)
Preparing to unpack .../zaproxy_2.14.0-0kali1_all.deb ...
Unpacking zaproxy (2.14.0-0kali1) ...
Setting up zaproxy (2.14.0-0kali1) ...
Processing triggers for kali-menu (2023.4.6) ...

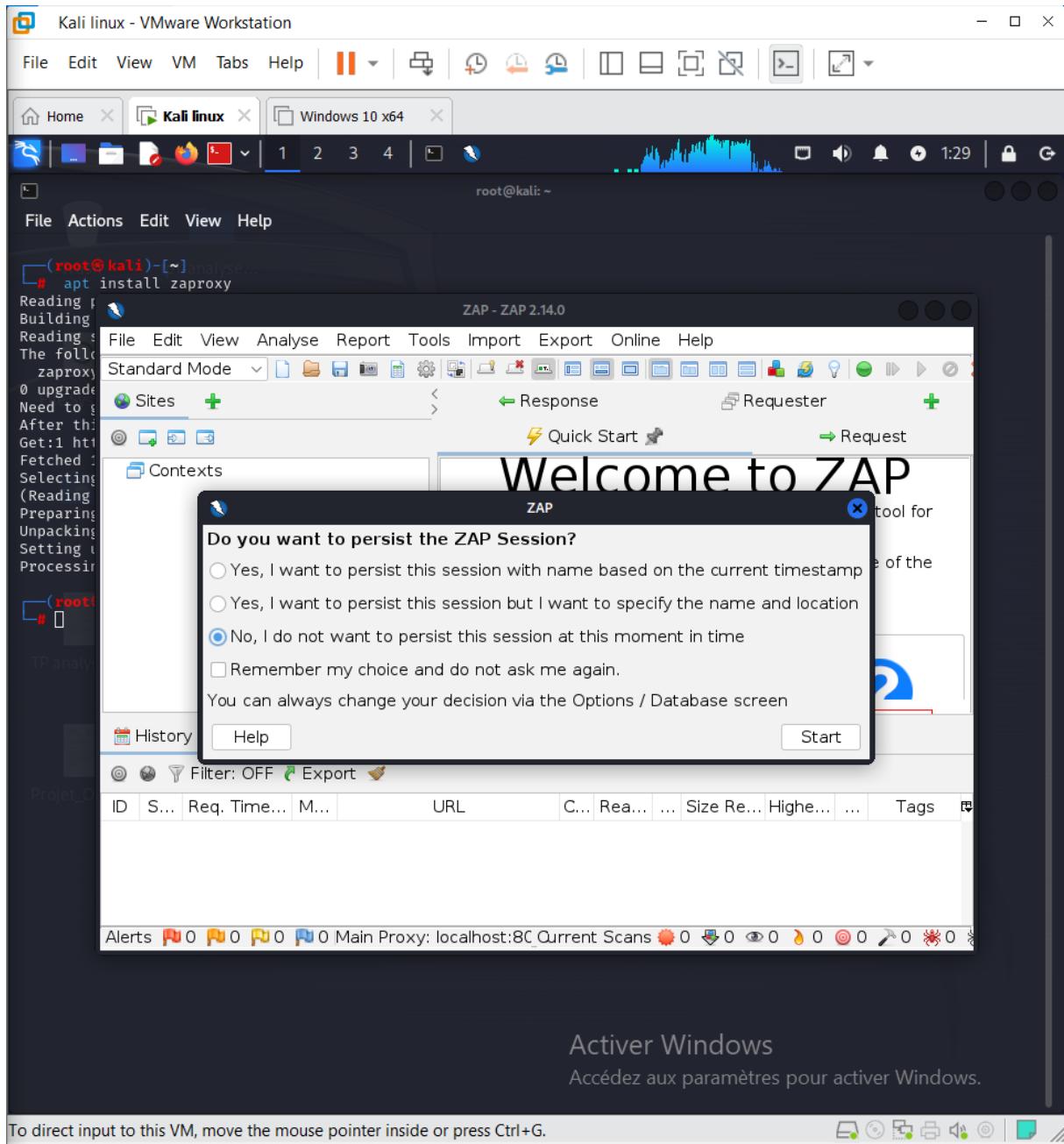
[root@kali]-[~]
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

2.lancer zaproxy



3. choisir l'option



4- Lancer un test automatique sur l'application "ghost" de la machine vulnérable « OWASP_Broken_Web_Apps_VM_1.2 »

The screenshot shows a Kali Linux desktop environment within a VMware Workstation window. The taskbar at the top has icons for Home, Kali linux, Windows 10 x64, and OWASP Broken Web Apps VM v... . The Firefox browser window is open to the URL 192.168.127.132/ghost/. The page content is as follows:

XSS & CSRF IFrame Injection RFI & LFI Flash Code Injection & Cookies Tutorials & Walkthroughs

This site is vulnerable to many different web attacks, the examples are very basic to get more information i will leave a set of references.

User Name:

Password:

Show Hints

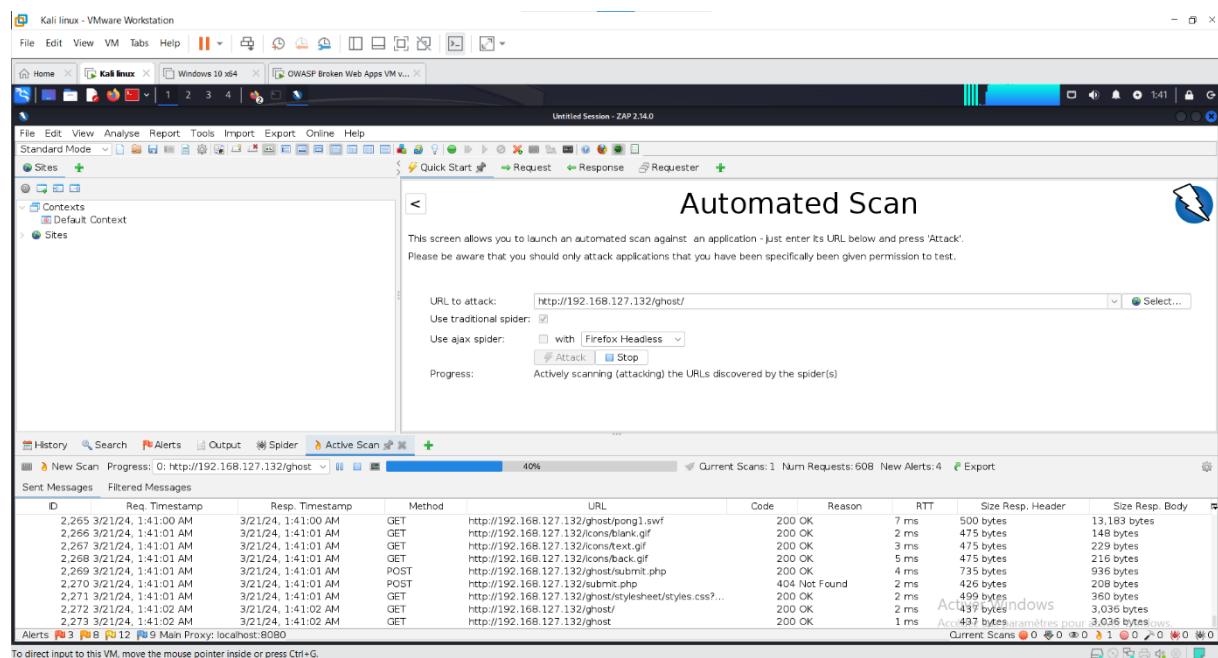
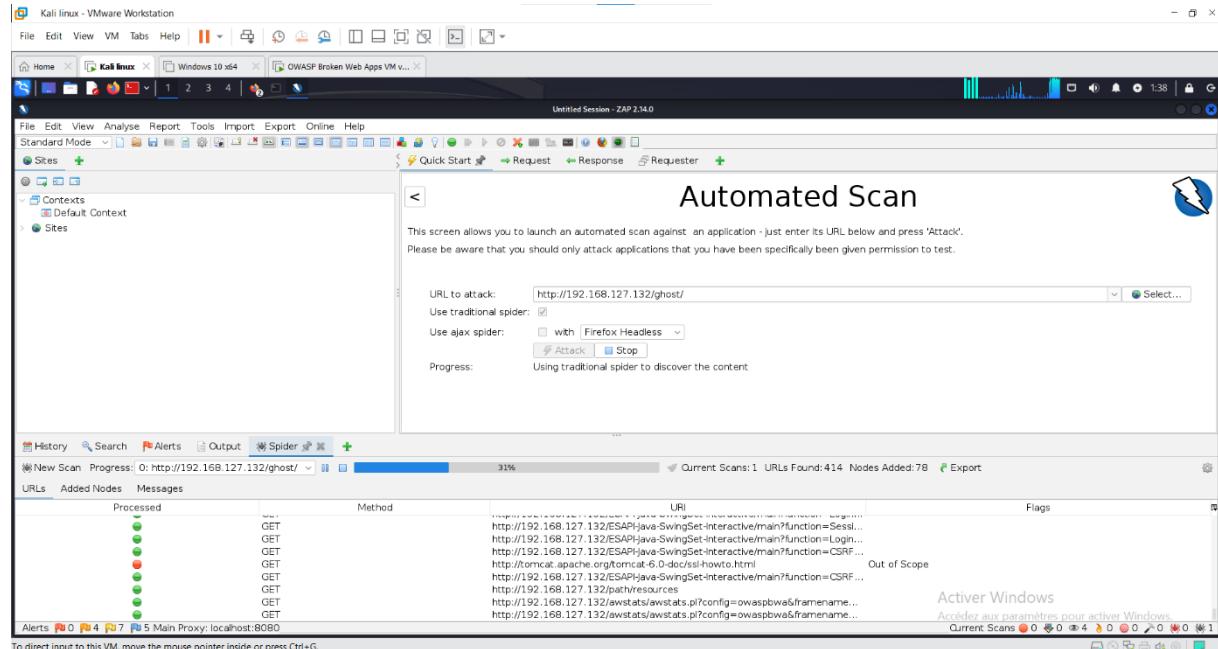
Developed By: Gh0\$7

Activer Windows
Accédez aux paramètres pour activer Windows.

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Ils'agit de decouvrir tous les repertoires , tous les fichier , tous les scripts et tout ce qui touve a l'interieur de site web

Afin qu'une fois que vous avez la liste de tous ces differents repertoires , nous saura exactement ou se trouvent les formulaires d'entree ,ou se trouve les zones possibles sur lesquelles nous pouvons executer nos charges utiles ou notre injection



5- Retourner les résultats du test

Il existe 34 alertes

The screenshot shows the ZAP 2.14.0 interface. In the main pane, titled "Automated Scan", there is a form to enter a URL (http://192.168.127.132/ghost/) and options for spider type (checkboxes for "Use traditional spider" and "Use ajax spider" with dropdowns for "wth" and "Firefox Headless"). Below the form, a progress bar indicates "Attack complete - see the Alerts tab for details of any issues found". At the bottom of the main pane, there is a note about adding alerts via context menu or double-clicking. The bottom-left corner of the interface shows a list of 34 alerts, including categories like Cross Site Scripting (DOM Based) and Content Security Policy Header Not Set.

Citons l'exemples de ses trois alertes :

Cross site scripting :

The screenshot shows the ZAP interface with the following details:

- Edit Alert** window for "Cross Site Scripting (DOM Based)" at URL `http://192.168.127.132/ghost/index.php`.
- Risk: High, Confidence: High.
- Parameter: user.
- Attack: `>lx3csVg/<sVg/oNloAd=alert(5397)//>`.
- Evidence: CWE ID: 79, WASC ID: 8.
- Description: Cross-site Scripting (XSS) is an attack technique that involves echoing attacker-supplied code into a user's browser instance. A browser instance.
- Other Info: Tag name: Input Att name: Att id:.

The main ZAP interface shows a scan for "Automated Scan" against `http://192.168.127.132/ghost/`. The Alerts tab lists the found XSS attack.

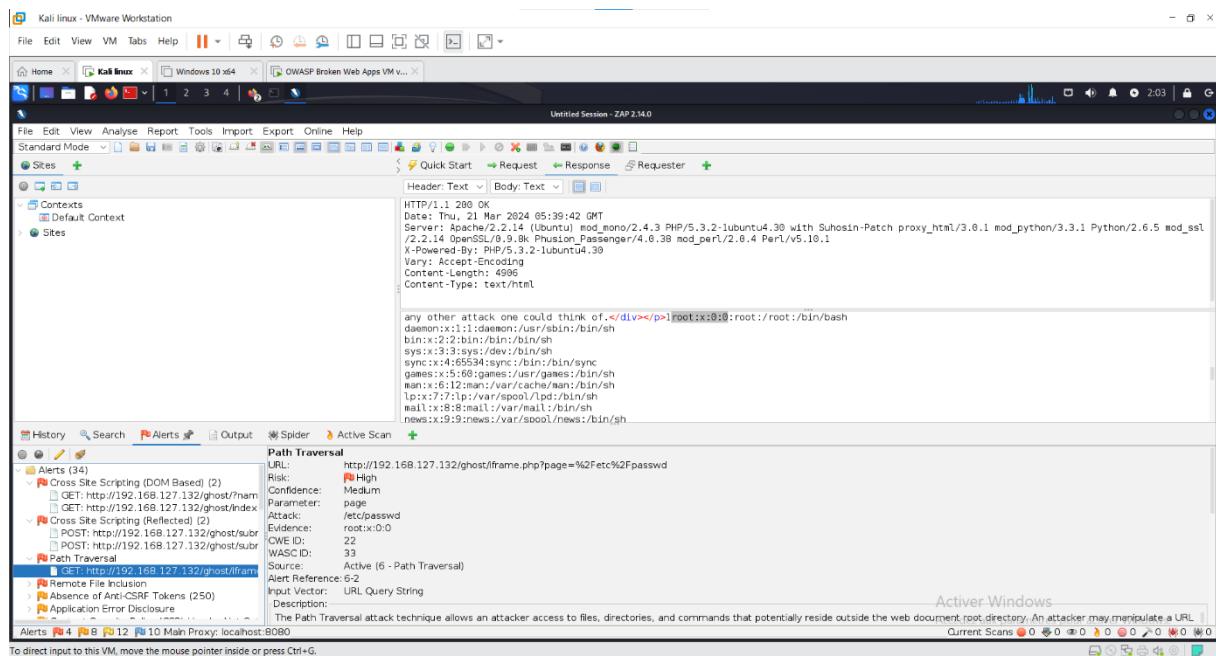
Cross site scripting (reflected)

The screenshot shows the ZAP interface with the following details:

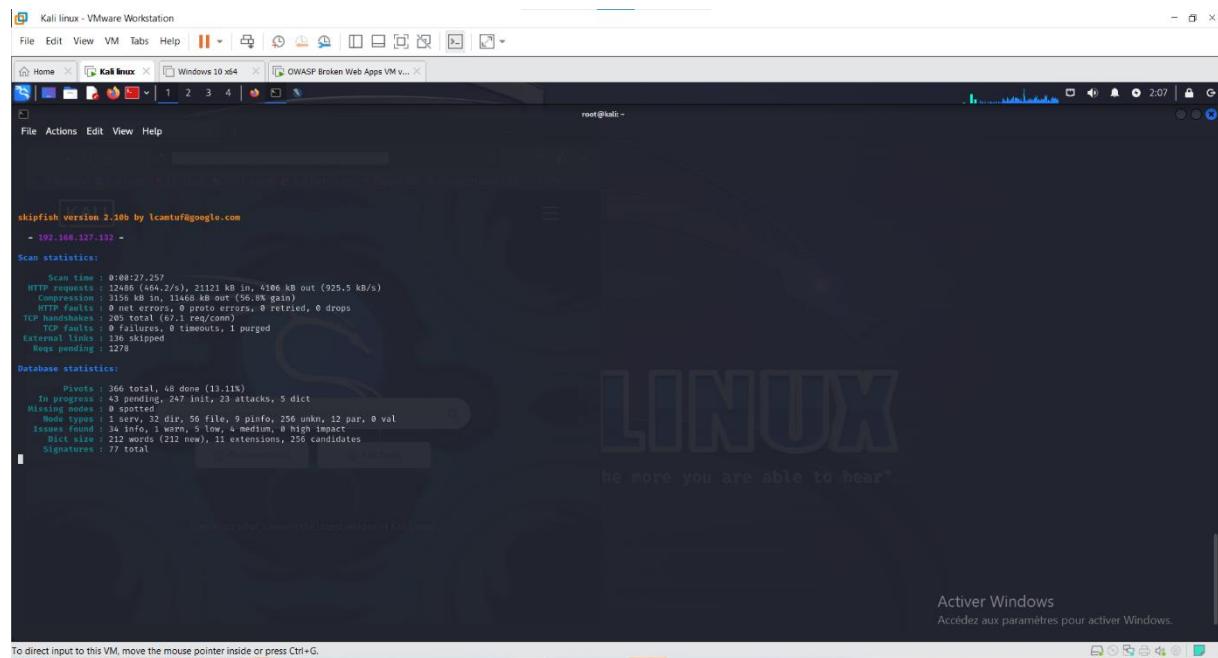
- Edit Alert** window for "Cross Site Scripting (Reflected)" at URL `http://192.168.127.132/ghost/submit.php`.
- Risk: High, Confidence: Medium.
- Parameter: user.
- Attack: `</div><script>alert(1);</scRipt></div>`.
- Evidence: CWE ID: 79, WASC ID: 8.
- Description: Cross-site Scripting (XSS) is an attack technique that involves echoing attacker-supplied code into a user's browser instance. A browser instance can be a standard web browser client, or a browser object.

The main ZAP interface shows a scan for "Automated Scan" against `http://192.168.127.132/ghost/submit.php`. The Requests tab shows the reflected XSS payload being sent.

Path traversal



Exercice2

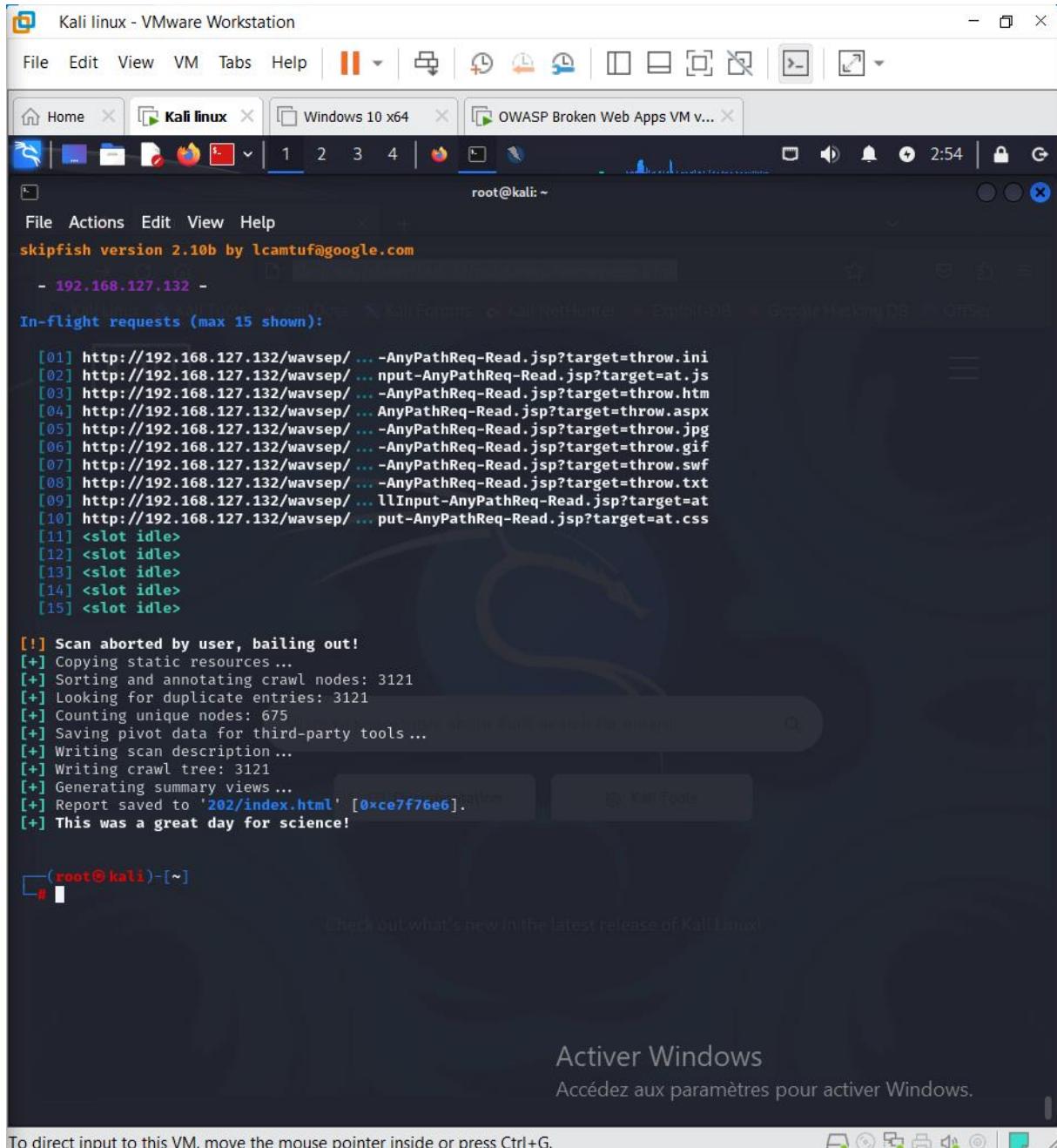


```
skipfish version 2.10b by lcamtuf@google.com
- 192.168.127.832 -
Scan statistics:
  Scan time : 0:00:27.257
  HTTP requests : 12486 (464.2/s), 21121 kB in, 4106 kB out (925.5 kB/s)
  Compression : 3156 kB in, 11468 kB out (56.8% gain)
  HTTP errors : 0 (0.0%), 0 timeouts, 0 retried, 0 dropped
  TCP handshakes : 205 total (67.1 req/sec)
  TCP faults : 0 failures, 0 timeouts, 1 purged
  External links : 139 skipped
  keeps pending : 1278

Database statistics:
  Pivots : 366 total, 48 done (13.11%)
  In progress : 43 pending, 247 init, 23 attacks, 5 dict
  Missing nodes : 0 spotted
  Issues Found : 34 info, 3 warn, 5 low, 4 medium, 0 high impact
  Dict size : 212 words (212 new), 11 extensions, 256 candidates
  Signatures : 77 total

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
```

Activer Windows
Accédez aux paramètres pour activer Windows.



Kali linux - VMware Workstation

File Edit View VM Tabs Help

Home Kali linux Windows 10 x64 OWASP Broken Web Apps VM v...

1 2 3 4

root@kali: ~

File Actions Edit View Help

skipfish version 2.10b by lcamtuf@google.com

- 192.168.127.132 -

In-flight requests (max 15 shown):

```
[01] http://192.168.127.132/wavsep/... -AnyPathReq-Read.jsp?target=throw.ini
[02] http://192.168.127.132/wavsep/... nput-AnyPathReq-Read.jsp?target=at.js
[03] http://192.168.127.132/wavsep/... -AnyPathReq-Read.jsp?target=throw.htm
[04] http://192.168.127.132/wavsep/... AnyPathReq-Read.jsp?target=throw.aspx
[05] http://192.168.127.132/wavsep/... -AnyPathReq-Read.jsp?target=throw.jpg
[06] http://192.168.127.132/wavsep/... -AnyPathReq-Read.jsp?target=throw.gif
[07] http://192.168.127.132/wavsep/... -AnyPathReq-Read.jsp?target=throw.swf
[08] http://192.168.127.132/wavsep/... -AnyPathReq-Read.jsp?target=throw.txt
[09] http://192.168.127.132/wavsep/... llInput-AnyPathReq-Read.jsp?target=at
[10] http://192.168.127.132/wavsep/... put-AnyPathReq-Read.jsp?target=at.css
[11] <slot idle>
[12] <slot idle>
[13] <slot idle>
[14] <slot idle>
[15] <slot idle>
```

[!] Scan aborted by user, bailing out!

[+] Copying static resources ...

[+] Sorting and annotating crawl nodes: 3121

[+] Looking for duplicate entries: 3121

[+] Counting unique nodes: 675

[+] Saving pivot data for third-party tools ...

[+] Writing scan description ...

[+] Writing crawl tree: 3121

[+] Generating summary views ...

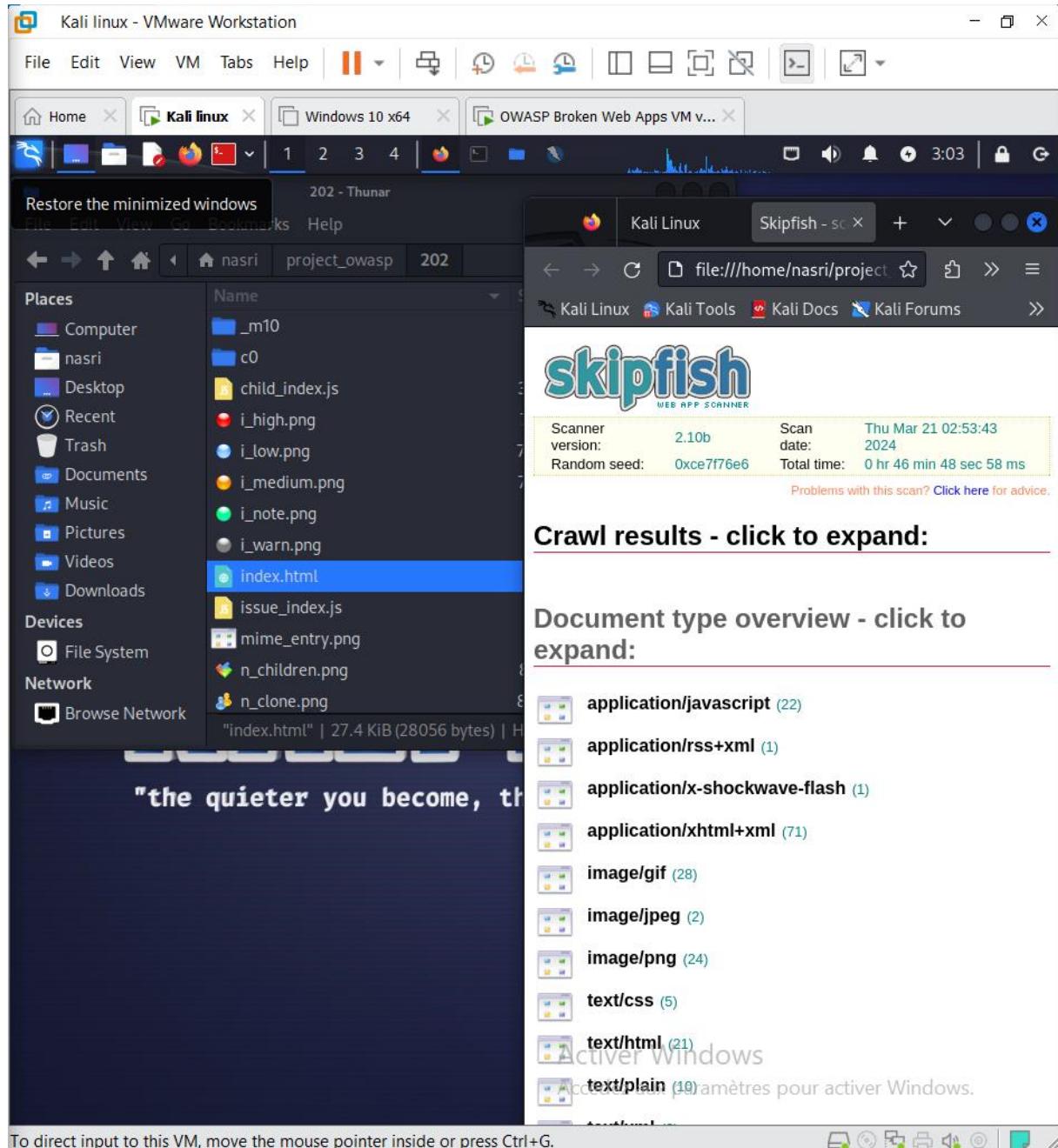
[+] Report saved to '202/index.html' [0xce7f76e6]. •

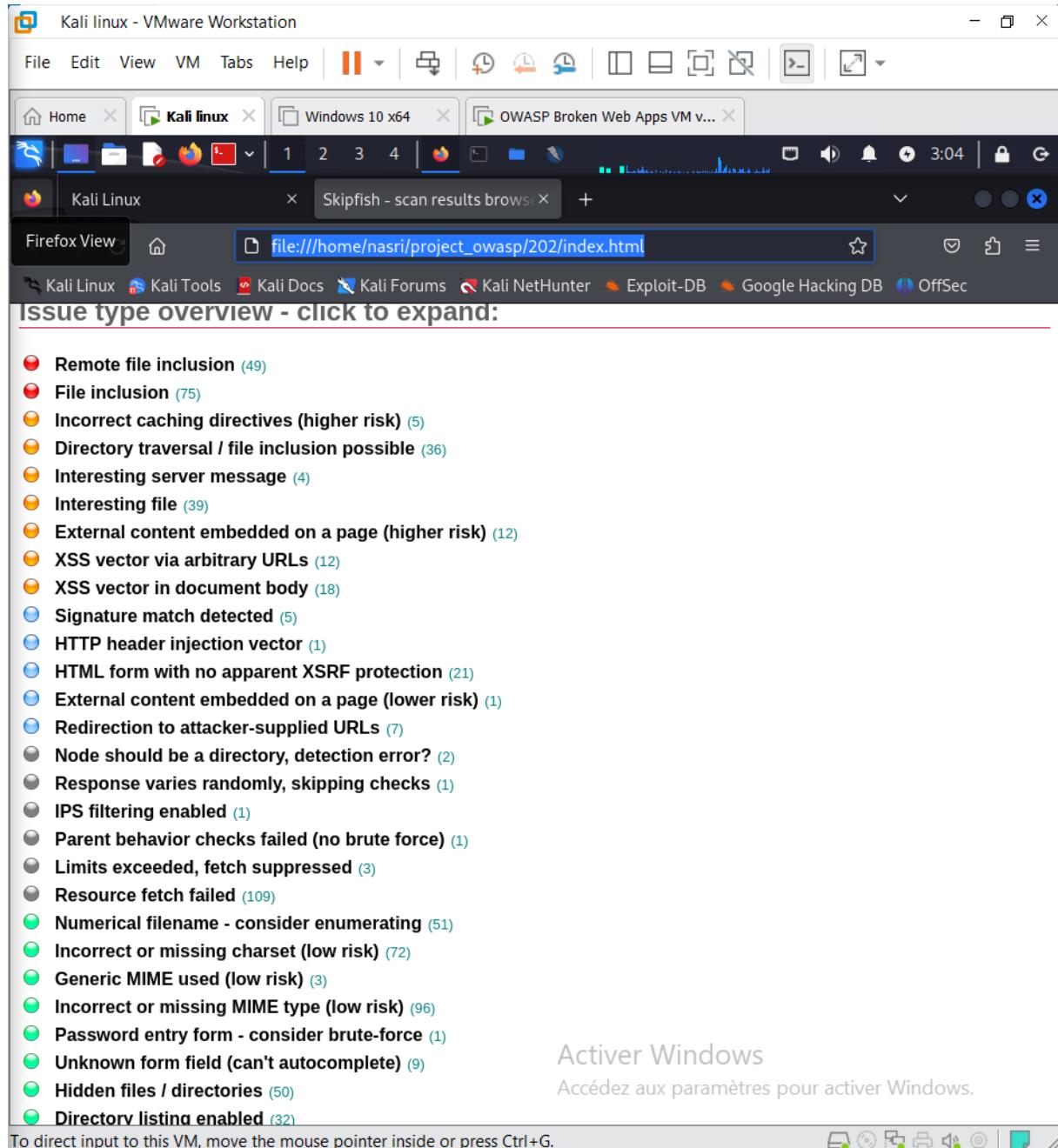
[+] THIS WAS A GREAT DAY FOR SCIENCE!

(root@kali)-[~]

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Activer Windows
Accédez aux paramètres pour activer Windows.





4.Analyses (SAST vs DAST)

Partie5 :

- 1- Déterminer les points forts des tests statiques de la sécurité des applications (SAST)
 - Détection précoce des vulnérabilités : SAST analyse le code source ou le bytecode avant même son exécution, permettant ainsi d'identifier les failles de sécurité dès le stade de développement.
 - Analyse approfondie du code : SAST examine le code ligne par ligne, ce qui permet de détecter les vulnérabilités potentielles même dans les parties du code rarement ou jamais exécutées
 - Intégration dans le processus de développement : Les outils SAST peuvent être intégrés dans les environnements de développement intégrés (IDE) ou les systèmes de gestion de versions (VCS), offrant ainsi une intégration transparente dans le processus de développement logiciel.
- 2- Déterminer les points forts des tests dynamiques de la sécurité des applications (DAST)
 - Évaluation de la sécurité en conditions réelles : Contrairement à SAST, DAST évalue les applications en cours d'exécution dans des conditions réelles, ce qui permet de détecter les vulnérabilités liées à la configuration du serveur, à la gestion des sessions, etc.
 - Détection des vulnérabilités liées à l'infrastructure : DAST identifie les vulnérabilités potentielles résultant de la configuration des serveurs, des paramètres réseau, des interactions avec d'autres systèmes, etc.
 - Facilité de déploiement : Les tests DAST peuvent être déployés facilement, car ils ne nécessitent pas l'accès au code source de l'application et peuvent être exécutés à partir de l'extérieur.

3- Comparer SAST et DAST

- Nature de l'analyse : SAST analyse le code source ou le bytecode, tandis que DAST teste l'application en cours d'exécution.
- Moment de détection des vulnérabilités : SAST détecte les vulnérabilités lors de la phase de développement, tandis que DAST identifie les failles lors de l'exécution de l'application.
- Profondeur de l'analyse : SAST offre une analyse plus approfondie du code, tandis que DAST se concentre sur les vulnérabilités exposées lors de l'exécution.

-Types de vulnérabilités détectées : SAST est efficace pour détecter les vulnérabilités liées au code, tandis que DAST est plus approprié pour identifier les vulnérabilités liées à l'infrastructure et aux interactions avec d'autres systèmes.

-Intégration dans le processus de développement : SAST peut être intégré dans les IDE ou les VCS, tandis que DAST est généralement utilisé en tant qu'outil externe.

-Complémentarité : Les deux approches sont souvent complémentaires, car elles détectent différents types de vulnérabilités à différents stades du cycle de vie du développement logiciel.

Ce tableau aussi montre une comparaison entre les deux :

	SAST	DAST
Quand l'utiliser	Pendant le développement du code	Après le déploiement de l'application
Couverture de test	Code source, byte code et binaires	Applications en cours d'exécution
Avantages	Détecte les vulnérabilités avant le déploiement, réduisant ainsi les coûts de réparation	Détecte les vulnérabilités qui ne sont visibles qu'une fois l'application déployée et en cours d'exécution
Inconvénients	Peut produire de nombreux faux positifs	Ne peut pas détecter certaines vulnérabilités qui n'apparaissent que lors de l'exécution de scénarios spécifiques
Idéal pour	Les phases de développement précoce et les projets avec un code base large	Les applications déjà en production et les situations nécessitant des tests en temps réel

