

Sri Lanka Institute of Information Technology

Web Security - IE2062



Bug Bounty Report 1

PERERA A.P.J

IT22280992

Group Y2S2.CS

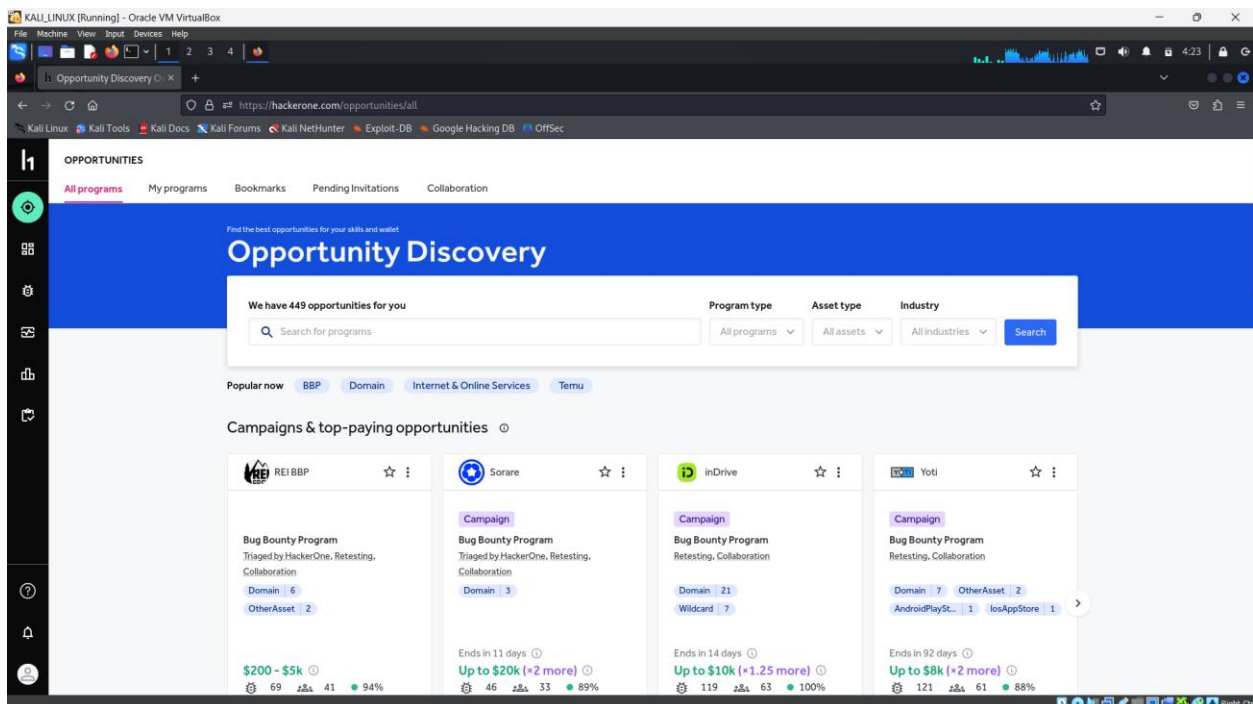
Table of Contents

1. Purpose	4
2. INTRODUCTION	5
3. Information Gathering.	8
3.1 Passive information gathering	8
3.2 Active information gathering	8
4. Passive information gathering tools	9
4.1 sublist3r	9
5. Target:	15
5.1 Nslookup	18
5.2 Nmap	18
5.3 Dmitry	19
6. Using OWASP Zap tool Scanning the vulnerabilities in the	21
6.1 Vulnerability	22
6.1.1 Vulnerability title	22
6.1.2 Vulnerability description	22
6.1.3 Affected components	24
6.1.4 Impact assessment	26
6.1.5 Steps to reproduce	27
6.1.6 Proof of concept	29
6.1.7 Proposed mitigation or fix	29
6.2 Vulnerability	30
6.2.1 Vulnerability title	30
6.2.2 Vulnerability description	30
6.2.3 Affected components	31
6.2.4 Impact assessment	34
6.2.5 Steps to reproduce	36
6.2.6 Proof of concept	38

6.2.7 Proposed mitigation or fix _____	38
6.3 Vulnerability. _____	39
6.3.1 Vulnerability title _____	39
6.3.2 Vulnerability description _____	39
6.3.3 Affected components _____	40
6.3.4 Impact assessment _____	42
6.3.5 Steps to reproduce _____	45
6.3.6 Proof of concept _____	46
6.3.7 Proposed mitigation or fix _____	46

1. Purpose

The purpose of this assignment is to assess vulnerabilities of the web application. So, <https://hackerone.com/> platform is used to find the websites and web applications for the Bug Bounty hunting. And there are a lot of Bug Bounty hunting platforms to improve our vulnerability assessing skills. So, the purpose of using this Hackerone platform is because this website legally protects us to do Bug Bounty hunting for real-world web applications. Using these websites benefits to get powerful knowledge about the penetration testing tool and how to use those tools. And these web audit reports are giving an excellent understanding of how to handle cybersecurity profession skills.



2. INTRODUCTION

Bug bounty hunting is an essential part of modern cybersecurity, allowing ethical hackers to test systems for vulnerabilities in exchange for rewards. Companies launch bug bounty programs to identify potential security flaws in their applications, networks, or infrastructure before malicious hackers can exploit them. As a bug bounty hunter, you'll use your skills to find security weaknesses such as cross-site scripting (XSS), SQL injection, or other critical vulnerabilities. After reporting these issues through a structured process, you may receive monetary rewards, recognition, or even job opportunities, depending on the program.

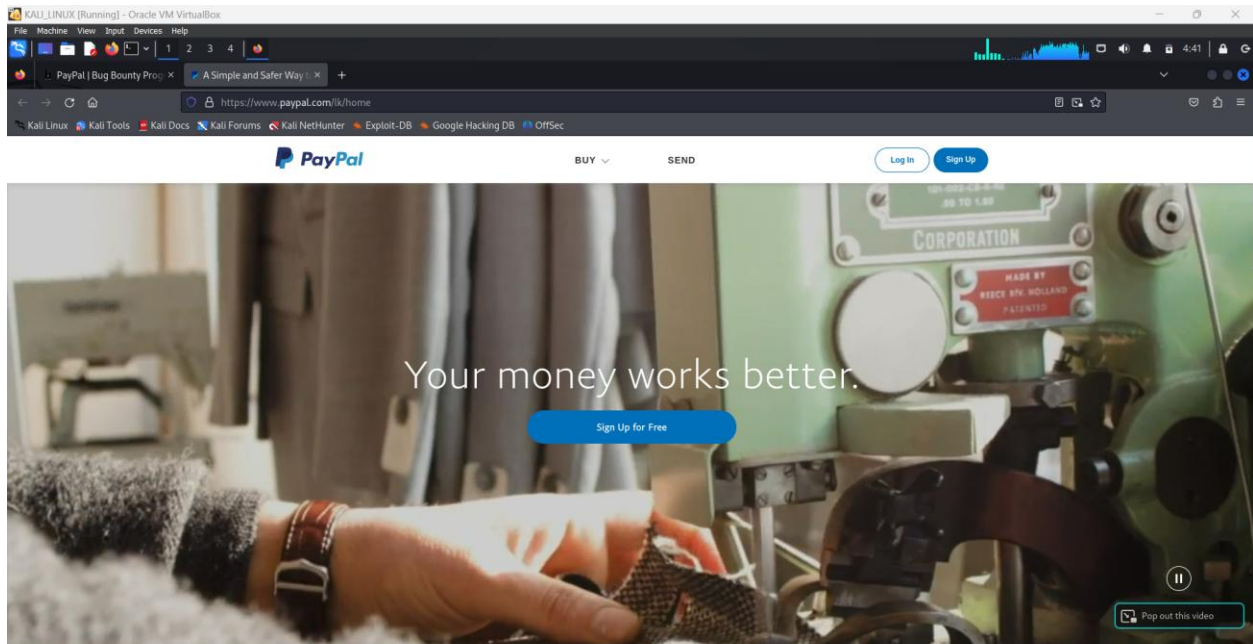
So, a lot of web-based companies and businesses are assigned to Bug Bounty programs to detect the vulnerabilities and fix those vulnerable domains before getting into attack. Hackerone(<https://hackerone.com/>) is one of the platforms that help web-based companies to fix vulnerabilities through Bug Bounty programs. And Hackerone platform and web-based companies are paying for penetration testing their web domains. So, I selected a web-based company called PayPal (<https://www.paypal.com>) for my Bug Bounty hunting program. The main purpose of PayPal is to facilitate secure, fast, and convenient online payments. It acts as a middleman between users and merchants, allowing individuals and businesses to send and receive money without sharing sensitive financial information like credit card or bank details. This makes online transactions safer and more accessible on a global scale.

The screenshot shows the PayPal Bug Bounty program page on the HackerOne platform. The page is titled "PayPal" and includes a sidebar with navigation links: Program guidelines, Scope, Hacktivity, Thanks, Updates, and Collaborators. The main content area is divided into several sections:

- Program highlights:** Includes "Platform Standards" (Fully compliant with Platform Standards), "Top Response Efficiency" (This program's response efficiency is above 90%), and "Managed by HackerOne", "Collaboration Enabled", and "Includes Retesting".
- Rewards:** A table showing rewards by severity level. Each severity lists the 90-day average bounty and the percentage of total resolved reports, if applicable.
- Scope Exclusions:** A section for "Core Ineligible Findings" that are out of scope and won't be rewarded.
- Overview:** A section for "Response Efficiency" showing a 91% response rate.

Low (0.1-3.9)	Medium (4.0-6.9)	High (7.0-8.9)	Critical (9.0-10.0)
Avg. bounty \$411 13.04% submissions	Avg. bounty \$2,983 56.04% submissions	Avg. bounty \$11,743 17.23% submissions	Avg. bounty \$29,867 7.62% submissions
\$50-\$1,000	\$1,000-\$10,000	\$10,000-\$20,000	\$20,000-\$30,000

<http://paypal.com/>



This Bug Bounty Assignment is used to be done according to the following web application security testing methodology.

Web Application Security Testing Methodology



Before moving into the information gathering stage, we need to consider the top 10 web application's Security Risks and vulnerabilities in 2024. Because we can get an excellent idea for success in our information gathering stage. According to the Sucuri Guides, The Open Web Application Security Project (OWASP) <https://owasp.org/> is an online community that creates web application security papers, techniques, documentation, tools, and technologies.

The OWASP Top 10 is a list of the top ten most frequent application Security Risks and vulnerabilities.

- Security misconfiguration.
- Insecure design.
- Broken Access control.
- Injection.
- Cryptographic failures.
- Insecure data storage.
- Data Leakage.
- Cross-site scripting.
- Input Validation.
- Server-side request forgery (SSRF).

So, these are the top 10 vulnerabilities found by the OWASP in 2024. We need to focus on these types of vulnerabilities according to the scope and rules provided by PayPal web-based company and the Hackerone platform.

I used this tools in those Bug Bountty reports.

- Nslookup.
- Nmap.
- Dmitry.
- OWASP ZAP
- Sublist3r.
- Nikto.
- Brupsuite.
- Knockpy.

3. Information Gathering.

Information gathering is the first step to building a strong foundation for this Bug Bounty hunting program. Because this step is about collecting the critical details of the targeted web application. If this step is not done well entire project can be a useless effort. So, more information means that we can capture more vulnerabilities from targeted domains. As an example, we have to find the targeted domain's IP addresses, details about open ports in the targeted domain, and what type of protection they use to protect their web application. According to the All About Testing (AAT), “The more useful information you have about a target, the more you can find vulnerabilities in the target and find more serious problems in the target by exploiting them. So, perfect information gathering is key to unlocking vulnerabilities from the target and it will help improve our vulnerability scanning process. Information Gathering can be divided into two parts. They are,

3.1 Passive information gathering

- Passive information gathering is collecting information from the targeted domain without invoking any kind of communication with the target systems.

Passive information gathering tools

- sublist3r.
- nslookup.

3.2 Active information gathering

➤ Active information gathering is collecting information from the targeted domain involves monitoring the target systems by building communication with the target. This method is detectable to the targeted system. Considering the Passive and Active information gathering, there are many tools to gather information from the target domain using both methods.

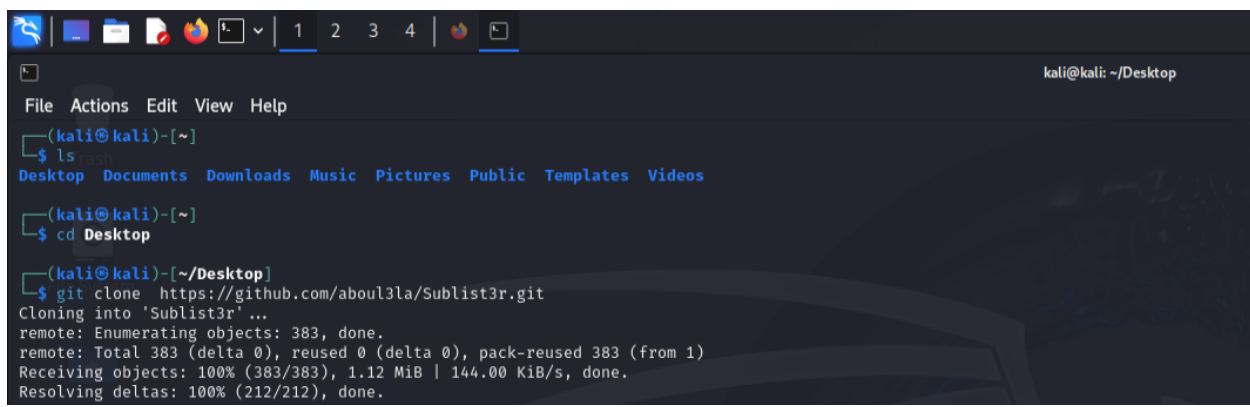
Active information gathering tools

- Nmap.
- Dmitry.

4. Passive information gathering tools

4.1 sublist3r

- Sublist3r is a tool designed in python and uses OSINT in order to enumerate subdomains of websites. It helps pen-testers in collecting and gathering subdomains for a domain which is their target.
- Download the sublist3r

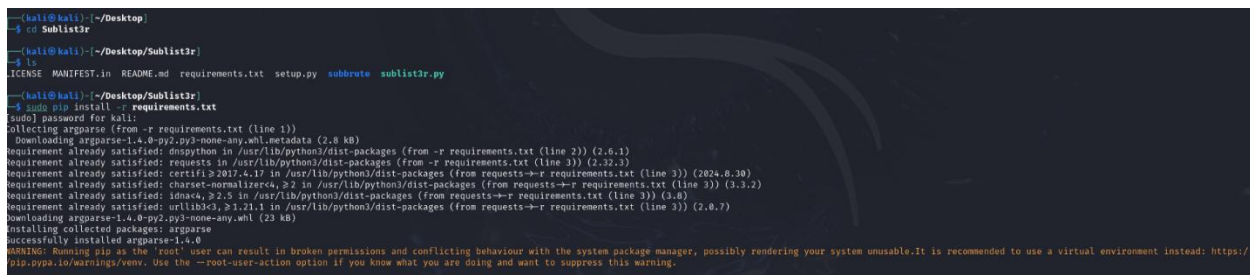


```
(kali@kali)-[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos

(kali@kali)-[~]
$ cd Desktop

(kali@kali)-[~/Desktop]
$ git clone https://github.com/about3la/Sublist3r.git
Cloning into 'Sublist3r'...
remote: Enumerating objects: 383, done.
remote: Total 383 (delta 0), reused 0 (delta 0), pack-reused 383 (from 1)
Receiving objects: 100% (383/383), 1.12 MiB | 144.00 KiB/s, done.
Resolving deltas: 100% (212/212), done.
```

- Install Dependencies in the Sublist3r directory.



```
(kali@kali)-[~/Desktop]
$ cd Sublist3r

(kali@kali)-[~/Desktop/Sublist3r]
$ ls
LICENSE MANIFEST.in README.md requirements.txt setup.py subbrute sublist3r.py

(kali@kali)-[~/Desktop/Sublist3r]
$ sudo pip install -r requirements.txt
[sudo] password for kali:
Collecting argparse (from -r requirements.txt (line 1))
  Downloading argparse-1.4.0-py2.py3-none-any.whl.metadata (2.8 kB)
Requirement already satisfied: dnspython in /usr/lib/python3/dist-packages (from -r requirements.txt (line 2)) (2.6.1)
Requirement already satisfied: requests in /usr/lib/python3/dist-packages (from -r requirements.txt (line 3)) (2.32.3)
Requirement already satisfied: certifi>=2017.4.17 in /usr/lib/python3/dist-packages (from requests==>r requirements.txt (line 3)) (2024.8.10)
Requirement already satisfied: charset-normalizer<4, >2 in /usr/lib/python3/dist-packages (from requests==>r requirements.txt (line 3)) (3.3.2)
Requirement already satisfied: idna<4, >2.5 in /usr/lib/python3/dist-packages (from requests==>r requirements.txt (line 3)) (3.8)
Requirement already satisfied: urllib3<3, >1.21.1 in /usr/lib/python3/dist-packages (from requests==>r requirements.txt (line 3)) (2.0.7)
Downloading argparse-1.4.0-py2.py3-none-any.whl (23 kB)
Installing collected packages: argparse
Successfully installed argparse-1.4.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager, possibly rendering your system unusable. It is recommended to use a virtual environment instead: https://pip.pypa.io/en/latest/faq/#issues. Use the --root-user-action option if you know what you are doing and want to suppress this warning.
```

```
(kali@kali)-[~/Desktop/Sublist3r]
$ sudo apt-get install python-requests
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package python-requests
```

- Install argparse module in the Sublist3r directory.

```
(kali@kali)-[~/Desktop/Sublist3r]
$ sudo apt-get install python-argparse
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'libpython2.7-stdlib' instead of 'python-argparse'
libpython2.7-stdlib is already the newest version (2.7.18-13.2).
libpython2.7-stdlib set to manually installed.
The following packages were automatically installed and are no longer required:
 fonts-liberation2 libverbs-providers libassuan libsigfilter libboost-iostreams1.63.0 libboost-thread1.63.0 libcephfs2 libdaxctl1 libgeos3.12.1t64 libgfan10 libgfrpc8 libgfxdr8 libglusterfs8 liblibverbs1 libmobiledevice6
 libsoncpp25 libndctl6 libplacebo338 libplist3 libpme1 libpostproc57 librados2 librdmacm164 libre2-10 libroc0.3 libu2f-udev libusbmuxd6 openjdk-17-jre openjdk-17-jre-headless python3-diskcache python3-mistune0 python3-pendulum
 python3-pytzdata rwho rwhoel
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 187 not upgraded.
```

- Check Sublist3r is ready to use and test the tool.

```
(kali@kali)-[~/Desktop/Sublist3r]
$ sublist3r -d paypal.com
```

```
KAU_LINUX [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Desktop/Sublist3r
$ sublist3r -d paypal.com

Sublist3r
# Coded By Ahmed Aboul-Ela - Baboul3la

[-] Enumerating subdomains now for paypal.com
[-] Searching now in Baidu..
[-] Searching now in Yahoo..
[-] Searching now in Google..
[-] Searching now in Bing..
[-] Searching now in Ask..
[-] Searching now in Netcraft..
[-] Searching now in DNSDumpster..
[-] Searching now in Virustotal..
[-] Searching now in Threatcrowd..
[-] Searching now in SSL Certificates..
[-] Searching now in PassiveDNS..
[-] Searching now in new 4 blocking our requests
[-] Total Unique Subdomains Found: 2030

cards--paypal.com
claim--paypal.com
www.claim--paypal.com
autodiscover.claim--paypal.com
cpuel.claim--paypal.com
mail.claim--paypal.com
webdisk.claim--paypal.com
webmail.claim--paypal.com
home--paypal.com
www.home--paypal.com
login--paypal.com
www.login--paypal.com
www.paypal.com
3pinages.paypal.com
API.paypal.com
API-3T.paypal.com
LVS2SIP.paypal.com
Lyncdiscover.paypal.com
LyncdiscoverInternal.paypal.com
access.paypal.com
ad.paypal.com
advendor.paypal.com
admin.paypal.com
advertising.paypal.com
apw.paypal.com
aktest.paypal.com
alp.paypal.com
am-vip.paypal.com
amex-xnl-vpn-live.paypal.com
amex_signing_cert.paypal.com
api.paypal.com
```

```
KAU_LINUX [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Desktop/Sublist3r

www.login--paypal.com
www.paypal.com
3pinages.paypal.com
API.paypal.com
API-3T.paypal.com
LVS2SIP.paypal.com
Lyncdiscover.paypal.com
LyncdiscoverInternal.paypal.com
access.paypal.com
ad.paypal.com
advendor.paypal.com
admin.paypal.com
advertising.paypal.com
apw.paypal.com
aktest.paypal.com
alp.paypal.com
am-vip.paypal.com
amex-xnl-vpn-live.paypal.com
amex_signing_cert.paypal.com
api.paypal.com
cors.api.paypal.com
sts-ft.api.paypal.com
api-3t.paypal.com
api-aa.paypal.com
api-aa-3t.paypal.com
api-sa.paypal.com
api-pxp.paypal.com
api-s.paypal.com
appmanagement.paypal.com
apps.paypal.com
autorec.paypal.com
autodiscover.paypal.com
av.paypal.com
avlb.paypal.com
beta-developer.paypal.com
www.beta-sandbox.paypal.com
3pinages-beta-sandbox.paypal.com
api.beta-sandbox.paypal.com
api-3t.beta-sandbox.paypal.com
api-aa-beta-sandbox.paypal.com
api-aa-3t-beta-sandbox.paypal.com
business.beta-sandbox.paypal.com
fido-beta-sandbox.paypal.com
mobileclient.beta-sandbox.paypal.com
pointofsale.beta-sandbox.paypal.com
securepayments.beta-sandbox.paypal.com
svcs.beta-sandbox.paypal.com
blueprint.paypal.com
bouthi.paypal.com
brandpermission.paypal.com
business.paypal.com
business-cog.paypal.com
business-signup.paypal.com
buyersauth-post.paypal.com
c.paypal.com
c3.paypal.com
```

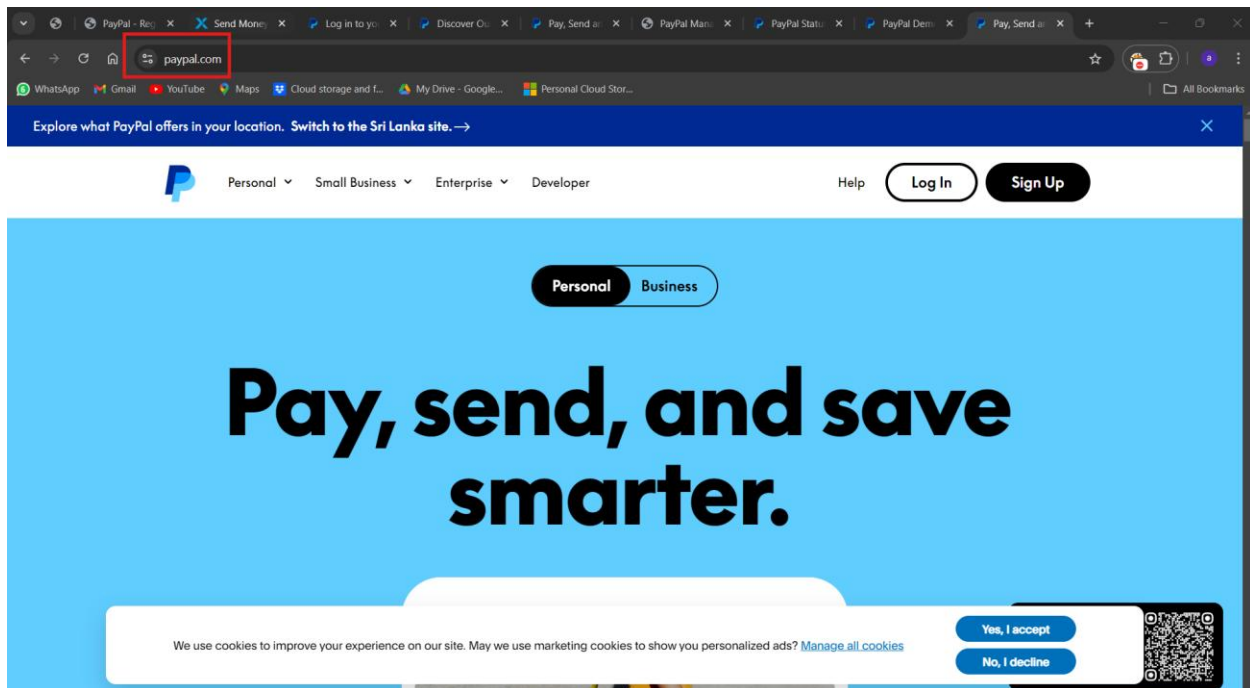
```
KAU_LINUX (Running) - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Desktop/Sublist3r
File Actions Edit View Help
cloudmonitor6.paypal.com
cloudmonitor7.paypal.com
cloudmonitor8.paypal.com
cloudmonitor9.paypal.com
capi.paypal.com
cpasapi.paypal.com
cpi.paypal.com
commerce.paypal.com
compass.paypal.com
compliance.paypal.com
coupons.paypal.com
cr-payflow.paypal.com
credit.paypal.com
credittaply.paypal.com
creditcenter.paypal.com
cs.paypal.com
cup-api.paypal.com
cup-api-phx.paypal.com
cup-api-slc-a.paypal.com
cup-api-slc-b.paypal.com
cup-mw.paypal.com
cup-mw-phx.paypal.com
cup-mw-slc-a.paypal.com
cup-mw-slc-b.paypal.com
cupapi.paypal.com
cxml-xmlpay.paypal.com
dmout.paypal.com
demo.paypal.com
densiplb.paypal.com
denwebconfb.paypal.com
dev.paypal.com
devblog.paypal.com
developer.paypal.com
beta.developer.paypal.com
dialin.paypal.com
dl.paypal.com
dme-origin-mw-1.paypal.com
dme-origin-mw-2.paypal.com
docmanager.paypal.com
dropzone.paypal.com
dubavlb.paypal.com
dubavlb.paypal.com
dubwebconfb.paypal.com
ebayee.paypal.com
www.ebayee.paypal.com
ecommerce.paypal.com
email-edg.paypal.com
email.paypal.com
click.emails.paypal.com
image.emails.paypal.com
view.emails.paypal.com
empresas.paypal.com
splunkng1.es.paypal.com
topoproduct01.es.paypal.com
topoproduct02.es.paypal.com
topoproduct03.es.paypal.com
```

```
KAU_LINUX (Running) - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Desktop/Sublist3r
File Actions Edit View Help
api.financing-wint.paypal.com
forms.paypal.com
fpdbs.paypal.com
fringe.paypal.com
fts.paypal.com
gadget.paypal.com
github.paypal.com
hermes.paypal.com
history.paypal.com
hotspot.paypal.com
hvd.paypal.com
ic.paypal.com
ice.paypal.com
icw.paypal.com
id-auth.paypal.com
idm-vip.paypal.com
images.paypal.com
information.paypal.com
ipamro.paypal.com
ipamrw.paypal.com
ipmb.paypal.com
ivrapi.paypal.com
keymaker.paypal.com
FMS-client.live.paypal.com
FMS01.live.paypal.com
login.paypal.com
lp.paypal.com
bb-vip.lvs.paypal.com
cg21splunkindexmaster1.lvs.paypal.com
infrastructure-vip.lvs.paypal.com
lvspuazlcs-vip.lvs.paypal.com
lvspuazoum1.lvs.paypal.com
lvspuazoum2.lvs.paypal.com
lvslip.paypal.com
lvslwebcom.paypal.com
lvslip.paypal.com
lvslwebcom.paypal.com
lynccdiscover.paypal.com
lynccdiscoverinternal.paypal.com
lynccdiscoverinternal.paypal.com
m.paypal.com
manager.paypal.com
meet.paypal.com
meetmer.paypal.com
meetemea.paypal.com
merchant.paypal.com
messageverificationcerts.paypal.com
mmlapi.paypal.com
mobile.paypal.com
mobileclient.paypal.com
mobilepayments.paypal.com
mpapi.paypal.com
mpitapi.paypal.com
offers.paypal.com
dev.offers.paypal.com
o-dev.offers.paypal.com
```

```
KAU_LINUX (Running) - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Desktop/Sublist3r
File Actions Edit View Help
mpapi.paypal.com
mltest1.paypal.com
offers.paypal.com
dev.offers.paypal.com
el-dev.offers.paypal.com
qa.offers.paypal.com
olp.paypal.com
onboarding.paypal.com
origin-chd-api-m.paypal.com
origin-chd-api-s.paypal.com
p.paypal.com
pandora.paypal.com
panel.paypal.com
partnermanager.paypal.com
payflow.paypal.com
pilot-plcc.payflow.paypal.com
plcc.payflow.paypal.com
payflowlink.paypal.com
images.payflowlink.paypal.com
payflowpro.paypal.com
payflowpro2.paypal.com
payflowprointernal.paypal.com
payment.paypal.com
payments-reports.paypal.com
paymentsdmr.paypal.com
paypalmanager.paypal.com
paypalreports.paypal.com
pdr.paypal.com
pep.paypal.com
personal.paypal.com
petition.paypal.com
Siteview.phx.paypal.com
Turboeuler-api.phx.paypal.com
lhw-vip.phx.paypal.com
calhadoop-vip.phx.paypal.com
cmapi.phx.paypal.com
docmanager.phx.paypal.com
fcquery-vip.phx.paypal.com
mailbatch-vip.phx.paypal.com
monitor-vip.phx.paypal.com
phx2wlp proxy1.phx.paypal.com
phx2wlp proxy2.phx.paypal.com
phx2wlp proxy3.phx.paypal.com
phx2wlp proxy4.phx.paypal.com
phxisplunkdeploy1.phx.paypal.com
phxisplunkdeploy2.phx.paypal.com
phxleemill1.phx.paypal.com
phxleemill2.phx.paypal.com
phxleemill3.phx.paypal.com
phxregapi.phx.paypal.com
phxoud1.phx.paypal.com
phxoud2.phx.paypal.com
phxoud3.phx.paypal.com
phxoud4.phx.paypal.com
phxppazcs-vip.phx.paypal.com
phxppasmon10.phx.paypal.com
```

```
KAU_LINUX (Running) - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Desktop/Sublist3r
File Actions Edit View Help
phx11av.paypal.com
phx11vip.paypal.com
phx11webconf.paypal.com
phx11webfcon.paypal.com
pics.paypal.com
pilot-agr.paypal.com
pilot-buyerauth-gest.paypal.com
pilot-campaigns.paypal.com
pilot-coupons.paypal.com
pilot-cml-smlpay.paypal.com
www-pilot-payflowlink.paypal.com
www-pilot-payflowlink.paypal.com
pilot-payflowpro.paypal.com
pilot-payflowprointernal.paypal.com
pilot-rm.paypal.com
pilot-xml-reg.paypal.com
pointofsale.paypal.com
pointofsale-new.paypal.com
pointofsale-s.paypal.com
posprivate.paypal.com
posprivate-api.paypal.com
posprivate-api-3t.paypal.com
posprivate-svcs.paypal.com
posprivatevpn.paypal.com
posprivatevpn-api.paypal.com
posprivatevpn-api-3t.paypal.com
posprivatevpn-svcs.paypal.com
pp-oud.paypal.com
pp-oud-mjs.paypal.com
pph-util-services.paypal.com
api.ppm.paypal.com
reporting.ppm.paypal.com
ppn.paypal.com
svcs.private.paypal.com
ppn.paypal.com
properties.paypal.com
proxo.paypal.com
prtc.paypal.com
pyp.paypal.com
arora.qa.paypal.com
hastimtest.qa.paypal.com
bt-infra-ci.qa.paypal.com
ci.qa.paypal.com
ci.qa.paypal.com
ci-cloud.qa.paypal.com
concordia-test.qa.paypal.com
api.concordia-test.qa.paypal.com
controlm-s-dev.qa.paypal.com
cronus-srepo.qa.paypal.com
csknowledgebassess.qa.paypal.com
cyclodge.qa.paypal.com
docker-artifactory-vip.qa.paypal.com
etleng3.qa.paypal.com
frontline.qa.paypal.com
github1-ete.qa.paypal.com
```


5. Target: <http://paypal.com>



KALI_LINUX [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

1 2 3 4

PayPal | Bug Bounty Pro: X

https://hackerone.com/paypal/policy_scopes

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

PayPal

Program guidelines

Scope

Hackactivity

Thanks

Updates

Collaborators

Search

Scope: All scopes

Maximum severity: Any

Bounty eligibility: All

Download Burp Suite Project Configuration File

Download CSV

View changes (Last updated on April 10, 2024)

1-48 of 48

Asset name	Type	Coverage	Max. severity	Bounty	Last update	Resolved Reports
py.pl	Domain	In scope	Critical	Eligible	Jan 24, 2023	6 (0%)
com.venmo	Android: Play Store	In scope	Critical	Eligible	Jul 13, 2023	7 (0%)
com.xoom.app	iOS: App Store	In scope	Critical	Eligible	Jul 13, 2023	0 (0%)
*.paypal.com	Other	In scope	Critical	Eligible	Jan 24, 2023	8 (0%)
paypal.me	Domain	In scope	Critical	Eligible	Jan 24, 2023	9 (0%)

1-48 of 48

PayPal

https://paypal.com/

@paypal

Send Money, Pay Online or Set Up a Merchant Account - PayPal.

Bug Bounty Program launched in Sep 2018

Response efficiency: 91%

Submit report

Response Efficiency

11 hours

Average time to first response

2 weeks, 1 day

Average time from triage to bounty

2 weeks, 1 day

Average time from submission to bounty

2 months, 3 weeks

Average time to resolution

91% of reports

KALI_LINUX [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

1 2 3 4

PayPal | Bug Bounty Pro: X

https://hackerone.com/paypal/policy_scopes

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

PayPal

Program guidelines

Scope

Hackactivity

Thanks

Updates

Collaborators

Search

Scope: All scopes

Maximum severity: Any

Bounty eligibility: All

Download Burp Suite Project Configuration File

Download CSV

View changes (Last updated on April 10, 2024)

1-48 of 48

Asset name	Type	Coverage	Max. severity	Bounty	Last update	Resolved Reports
*.paydiant.com	Other	In scope	Critical	Eligible	Jul 13, 2023	31 (2%)
com.paypal.merchant	iOS: App Store	In scope	Critical	Eligible	Jul 13, 2023	2 (0%)
com.paypal.android.p2pmobile	Android: Play Store	In scope	Critical	Eligible	Jul 13, 2023	19 (1%)
Braintree SDK	Other	In scope	Critical	Eligible	Apr 10, 2024	2 (0%)
*.venmo.com	Other	In scope	Critical	Eligible	Jul 13, 2023	67 (3%)

1-48 of 48

PayPal

https://paypal.com/

@paypal

Send Money, Pay Online or Set Up a Merchant Account - PayPal.

Bug Bounty Program launched in Sep 2018

Response efficiency: 91%

Submit report

Response Efficiency

11 hours

Average time to first response

2 weeks, 1 day

Average time from triage to bounty

2 weeks, 1 day

Average time from submission to bounty

2 months, 3 weeks

Average time to resolution

91% of reports

KALI_LINUX [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

1 2 3 4

PayPal | Bug Bounty Program

https://hackerone.com/paypal

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Program guidelines

- Scope
- Hacktivity
- Thanks
- Updates
- Collaborators

Disclosure Guidelines

By providing a Submission or agreeing to the Program Terms, You agree that you may not publicly disclose your findings or the contents of your Submission to any third parties in any way without PayPal's prior written approval.

Failure to comply with the Program Terms will result in **immediate** disqualification from the Bug Bounty Program and ineligibility for receiving any Bounty Payments.

Scope for Web Applications

In-Scope Vulnerabilities

Accepted in-scope vulnerabilities include, but are not limited to:

- Disclosure of sensitive or personally identifiable information that does not belong to you.
- Cross-Site Scripting (XSS)
- Cross-Site Request Forgery (CSRF) for sensitive functions in a privileged context
- Server-side or remote code execution (RCE)
- Authentication or authorization flaws, including IDOR and authentication bypass.
- Injection vulnerabilities, including SQL and XML injection.
- Directory traversal
- Significant security misconfiguration with a verifiable vulnerability
- Exposed credentials, disclosed by PayPal or its employees, that pose a valid risk to an in-scope asset.

Out-of-Scope Vulnerabilities

Certain vulnerabilities are considered out-of-scope for the Bug Bounty Program. Those out-of-scope vulnerabilities include, but are not limited to:

- Any physical attacks against PayPal property or data centers
- Username enumeration on customer facing systems (i.e. using server responses to determine whether a given account exists)
- Scanner output or scanner-generated reports, including any automated or active exploit tool.
- Man-in-the-Middle attacks.
- Vulnerabilities involving stolen employee/consumer/merchant credentials or physical access to a device.
- Social engineering attacks, including those targeting or impersonating internal employees by any means (e.g. customer service chat features, social media, personal domains, etc.)
- Open redirection, except in the following circumstances:
 - Clicking a PayPal-owned URL immediately results in a redirection, and/or

5.1 Nslookup

- Nslookup is perfect DNS enumeration. That means this is a tool for gathering information about the Domain Name System (DNS) of the targeted system. Nslookup tool help to find out the information related to DNS record names, IP addresses of a target, DNS domain names, and the MX records for the domain or the NS servers of the domain. This tool is already built in the Kali Linux environment. So, I gather the information that all selected domains to get a better understanding of DNS information related to the web application (PayPal.com).

```
(kali㉿kali)-[~/Desktop/Sublist3r]
$ nslookup paypal.com
Server:      172.16.10.100
Address:     172.16.10.100#53

Non-authoritative answer:
Name:   paypal.com
Address: 151.101.129.21
Name:   paypal.com
Address: 192.229.210.155
Name:   paypal.com
Address: 151.101.65.21
```

- We can see 3 Ip addresses.

5.2 Nmap

- Nmap is a powerful tool for network scanning, and the way it operates can reveal crucial information about network infrastructure. Let's break down what happens in more detail when using Nmap and why it's useful, particularly in cybersecurity and system administration.

```
(kali@kali)-[~/Desktop/Sublist3r]
$ nmap 172.16.10.100
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-30 07:00 EDT
Nmap scan report for MADCSTD01.sliitstd.local (172.16.10.100)
Host is up (0.011s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
8008/tcp  open  http  unsafe-inline (6)
8010/tcp  open  xmpp   unsafe-inline (2754)
Nmap done: 1 IP address (1 host up) scanned in 5.00 seconds
```

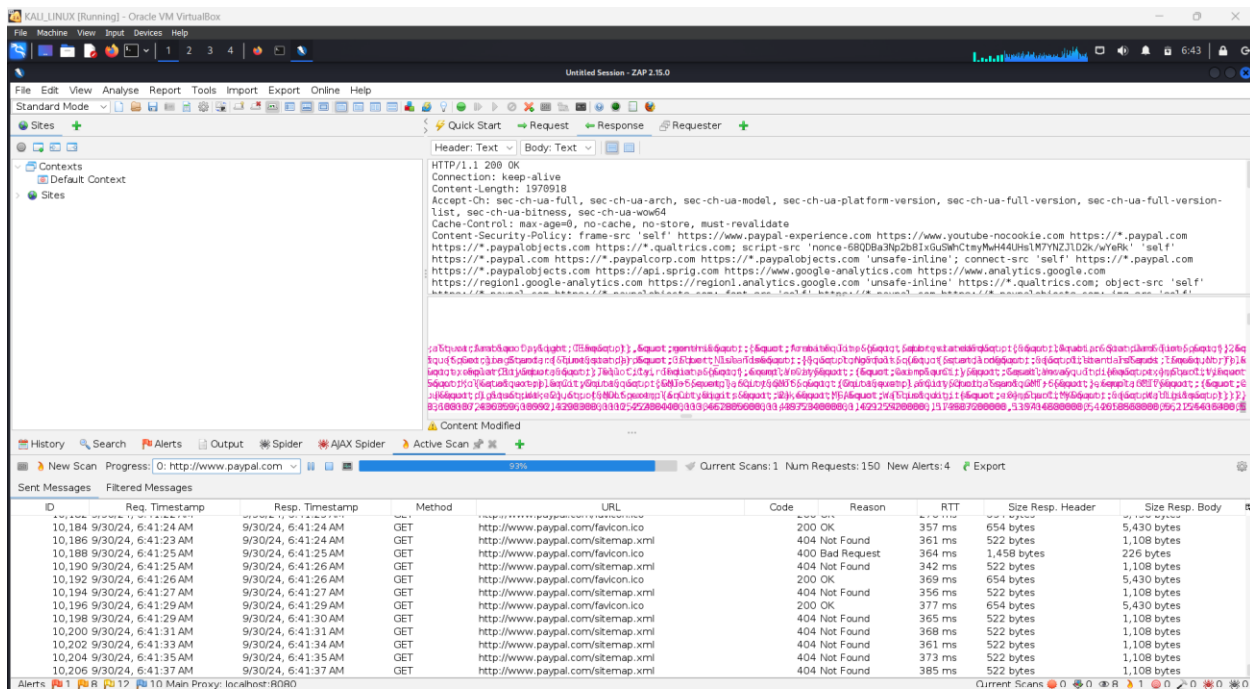
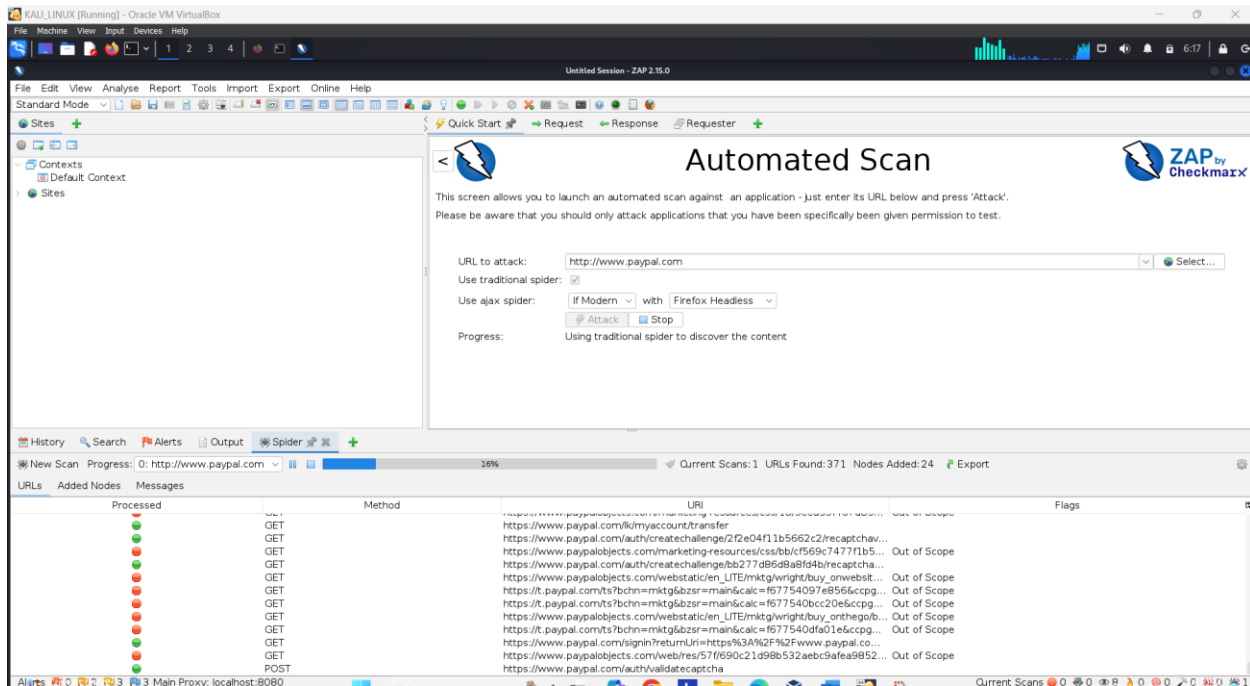
5.3 Dmitry

- Dmitry is a collection of information-gathering tools. Because of that, this tool is a combination or package of tools. Using this tool, we can gather details related to Whois lookup web tool information, Netcraft information, and open port details. Because this tool gathers information about open ports, Dmitry is an Active information gathering tool.

```
KALI LINUX (Running) - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Desktop/Sublist3r
$ dmitry paypal.com
Deepmagic Information Gathering Tool
"There be some deep magic going on"
HostIP:151.101.65.21
HostName:paypal.com
Gathered Inet-whois information for 151.101.65.21
Inetnum: 151.101.0.0 - 151.105.255.255
netname: NON-RIPE-NCC-MANAGED-ADDRESS-BLOCK
descr: IPv4 address block not managed by the RIPE NCC
remarks:
remarks: For registration information,
remarks: you can consult the following sources:
remarks: IANA
remarks: http://www.iana.org/assignments/ipv4-address-space
remarks: ial-registry
remarks: http://www.iana.org/assignments/ipv4-recovered-address-space
remarks: AFRINIC (Africa)
remarks: http://www.afrinic.net/ whois.afrinic.net
remarks: APNIC (Asia Pacific)
remarks: http://www.apnic.net/ whois.apnic.net
remarks: ARIN (Northern America)
remarks: http://www.arin.net/ whois.arin.net
remarks: LACNIC (Latin America and the Caribbean)
remarks: $ lacnic.net
remarks:
remarks: EU # Country is really world wide
admin-c: IANA1-RIPE
tech-c: IANA1-RIPE
status: ALLOCATED UNSPECIFIED
mnt-by: RIPE-NCC-MNT
created: 2019-01-07T18:46:38Z
last-modified: 2019-01-07T18:46:38Z
source: RIPE
role: Internet Assigned Numbers Authority
address: see http://www.iana.org.
admin-c: IANA1-RIPE
ANI1-RIPE
nic-hdl:
remarks: For more information on IANA services
remarks: go to IANA web site at http://www.iana.org.
mnt-by: RIPE-NCC-MNT
created: 1978-01-01T00:00:00Z
last-modified: 2001-09-22T09:31:27Z
```

```
KALI_LINUX (Running) - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali:~/Desktop/Sublist3r
File Actions Edit View Help
created: 1978-01-01T00:00:00Z
last-modified: 2024-09-27T09:31:27Z
source: RIPE # filtered
% This query was served by the RIPE Database Query Service version 1.114 (ABERDEEN)
Gathered Inic-whois information for paypal.com
Domain Name: PAYPAL.COM
Registry Domain ID: R01P048-DOMAIN.COM-VRSN
Registrar WHOIS Server: whois.markmonitor.com
Registrar URL: http://www.markmonitor.com
Updated Date: 2024-06-11T18:17:32Z
Creation Date: 1999-07-15T03:32:11Z
Registry Expiry Date: 2025-07-15T03:32:11Z
Registrar: MarkMonitor Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
Registrar Abuse Contact Phone: +1.2866811750
Domain Status: clientTransferProhibited https://icann.org/epp/clientTransferProhibited
Domain Status: clientUpdateProhibited https://icann.org/epp/clientUpdateProhibited
Domain Status: serverDeleteProhibited https://icann.org/epp/serverDeleteProhibited
Domain Status: serverTransferProhibited https://icann.org/epp/serverTransferProhibited
Name Server: NS1.P57.DVNET.NET
Name Server: NS2.P57.DVNET.NET
Name Server: PDNS100.ULTRADNS.COM
Name Server: PDNS100.ULTRADNS.NET
DNSSEC: signedDelegation
DNSSEC DS Data: 34888 13 2 D9E6A8ACB718F093B596F9D189090AC47C3F557312201DFCC50D4128C8BF50
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2024-09-30T11:01:58Z <<<
NOTICE: The expiration date displayed in this record is the date the
registrar's sponsorship of the domain name registration in the registry is
currently set to expire. This date does not necessarily reflect the expiration
date of the domain name registrant's agreement with the sponsoring
registrar. Users may consult the sponsoring registrar's Whois database to
view the registrar's reported date of expiration for this registration.
our Whois
database through the use of electronic processes that are high-volume and
automated except as reasonably necessary to register domain names or
modify existing registrations; the Data in VeriSign Global Registry
Services' ("VeriSign") Whois database is provided by VeriSign for
information purposes only, and to assist persons in obtaining information
about or related to a domain name registration record. VeriSign does not
warrant or agree to abide
by the following terms of use: You agree that you may use this Data only
for lawful purposes and that under no circumstances will you use this Data
to (i) allow, enable, or otherwise support the transmission of mass
```

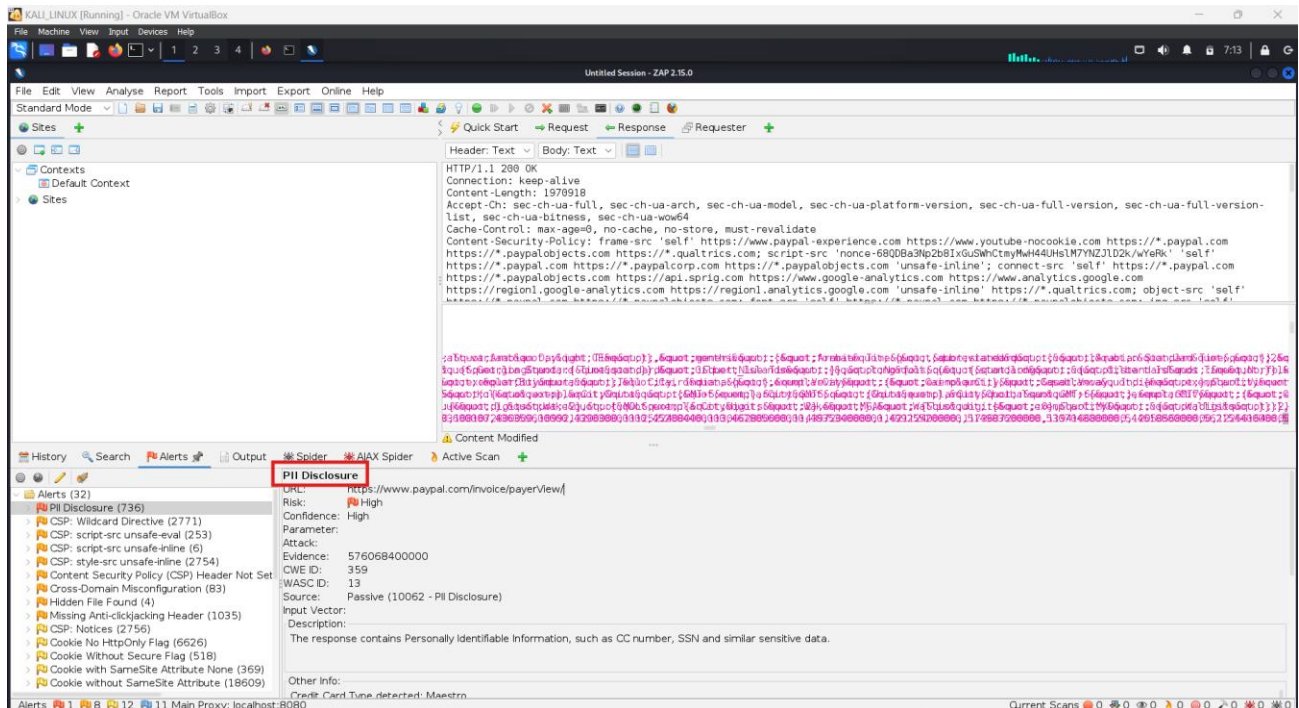
6. Using OWASP Zap tool Scanning the vulnerabilities in the <http://www.paypal.com>.



6.1 Vulnerability

6.1.1 Vulnerability title

- PII Disclosure (Personally Identifiable Information).



6.1.2 Vulnerability description

- PII is a short form used for "personally identifiable information", which refers to any piece of information that can be used to identify an individual such as CC number, SSN and similar sensitive data. The term "PII disclosure" means an occurrence of such information being revealed with a possibility of causing damage to the concerned individual. This may consist of elements ranging from simple information like name, address, phone numbers, and social security numbers to more private information such as individual health condition, financial status or any other personal activity. The description in the above stated vulnerability concerns a PII disclosure that occurs in situations where the information has been misappropriated or even just shared inappropriately which can lead to subversion... identity theft among other malpractices. But under normal

circumstances, such a risk can be mitigated because the system usually tends to categorize this as a false alarm. Therefore, the tool unnecessarily alerts this because it is not aware of the data that matters to you.

6.1.3 Affected components

- The lack of proper PII protection could lead to breach of such information at different levels of the organization system hardware, structure, and practices. The following are some of the areas that experience this threat most often.

1. DBMS

- Relational DBMS: A poorly secured or exposed to the internet SQL database, for instance, may turn out sensitive if appropriate countermeasures are not employed.
NoSQL: On the other hand, NoSQL databases might also pose a risk in the absence of proper access and encryption mechanisms.

2. Web-Based Software

- Code of the Application: Security risks such as SQL injection, cross site scripting (XSS), and parameter tampering may expose PII.
User Authentication Strategies: The absence of multi- factor authentication or the presence of policies allowing weak passwords would give an opportunity for ‘hackers’ to access applications with stored PII.

3. Network Architecture

- Firewalls and Routers: Firewalls and Routers when not appropriately configured can expose the sensitive information (PII) within the data while it is in transit.
VPN tunnels and other Secure channels: Situation where applicable secured protocols are not followed e.g. not using HTTPS may cause threats from Body Snatchers.

4. Services offered in the Cloud

- Storing Data in the Cloud: Malconfigured cloud storage facilities (amazon S3, google cloud nowadays) can cause accidental data leaks.
SaaS: Cloud applications can also carry weaknesses that could be exploited to extract the PII.

5. Stationary and mobile devices

- Devices used by the Users: Devices like laptops and desktops or even mobile phones which hold and process PII can be infected by malware or even phishing or loss of the actual device.

Cessable Storage: Unencrypted Personal Identifiable Information stored in USB flash drives and portable external hard drives would also be prone to loss.

6.1.4 **Impact assessment**

- An impact assessment for PII disclosure entails the appreciation of the impact of a breach or access to sensitive personal information which can be classified as personal data. This assessment leaves the organization in a position to grasp the threats, rank the approaches for alleviation, and also avoid the risk of being intransigent with the laws. There is a stepwise process of carrying out a PII disclosure impact assessment explained as follows:

1. Identification of Affected PII

Types of PII Involved:

- Identify the specific types of PII that have been compromised which may include, names, Social Security numbers, financial details, and medical records of individuals.

Volume of Affected Data:

- Establish the number of records compromised and the extent of the data breach (e.g., how many people are affected).

2. Assessment of Impact on Individuals

Risk of Identity Theft:

- Assess the risk for individuals being exposed to identity theft or fraud in relation to the nature of the PII which may have been exposed.

Emotional Distress:

- Assess the impact caused to the potential victims, for instance, any stress, anxiousness, or fear concerning the abuse of their private information.

3. Assessment of Impact on the Organization

Regulatory Compliance:

- Assess the risk that the organization might fail to comply with relevant data protection legislation, for example, GDPR and HIPAA, and what penalties or summons would apply.

Financial Loss:

- Discuss the overall financial effect that such an episode would have on the firm including spending for action against the incident, lawyer, promotion, and likely rewards.

6.1.5 Steps to reproduce

- This is done by recreating the context in which the Personally Identifiable Information (PII) leak may happen or be exposed or accessed by unauthorized parties. It should be remembered that these activities are illegal and amoral especially when they are done without permission or in an unauthorized environment. The following steps are illustrative and can only be performed in a safe environment where hacking is performed ethically such as a penetration testing lab or with permission.

- Recreating a PII Disclosure Vulnerability Step by Step

1. Target System Selection

- Web Application Selection: Here select any web application or service that has any PII data integrated like, say coronation online registration, an online shopping website, customer database etc.
- Information Needs: Understand the system structure, features and the issues associated with that system by carrying out a background study of the application. Use basic tools such as whois, nslookup or netcraft to get data about the target server and its technology resources.

2. Reconnaissance

- Locate Possible Points of Entry: Go through the application and look for places where PII is imputed such as login fields, search fields etc.
- Automated Tools: Use tools such as OWASP ZAP, Burp Suite or Nikto and find weak points within the application and assess respective vulnerabilities.

3. Assess the Risk for Insecure Direct Object References(IDOR).

- URL Parameter Manipulation: Test how URL parameters such as example.com/user?id=123 can be changed to get another person's PII by accessing the PII of User 123 of the application database.
- Use of Id Enumeration: Different Ids are tried to see whether the application will fetch any PII of other users as well.

4. Analyze Input Validation

- SQL Injection: Devise Attack Vectors: Load a number of fields with SQL query injections, such as ' OR '1'='1, to return some records from the database.

5. Test for Insecure API Endpoints

- **API Exploration** We are tasked here finding out any public API if available and its endpoints for possible sensitive data leaking. Examine Request/Response So use postman or Insomnia etc. to hit the API with a query and see if some sensitive information is being given in response without authentication.

1. Check for misconfigured permissions

- **Role based access control testing** Log in as a user with a role and try to access information of PII level where it shouldn't be available based on the user role. Try unauthorized access Low privilege account login and accessing data which is otherwise restricted.

2. Assess Data Storage and Transmission

- **Inspect Storage Mechanisms** Determine if explicit information is stored in plain text in Data bases or in altering files. Intercept Network Traffic Equipment like Wireshark and Fiddler may be used to tap network connection and inspect its contents particularly searching in the files for the presence of PII that have not been encrypted.

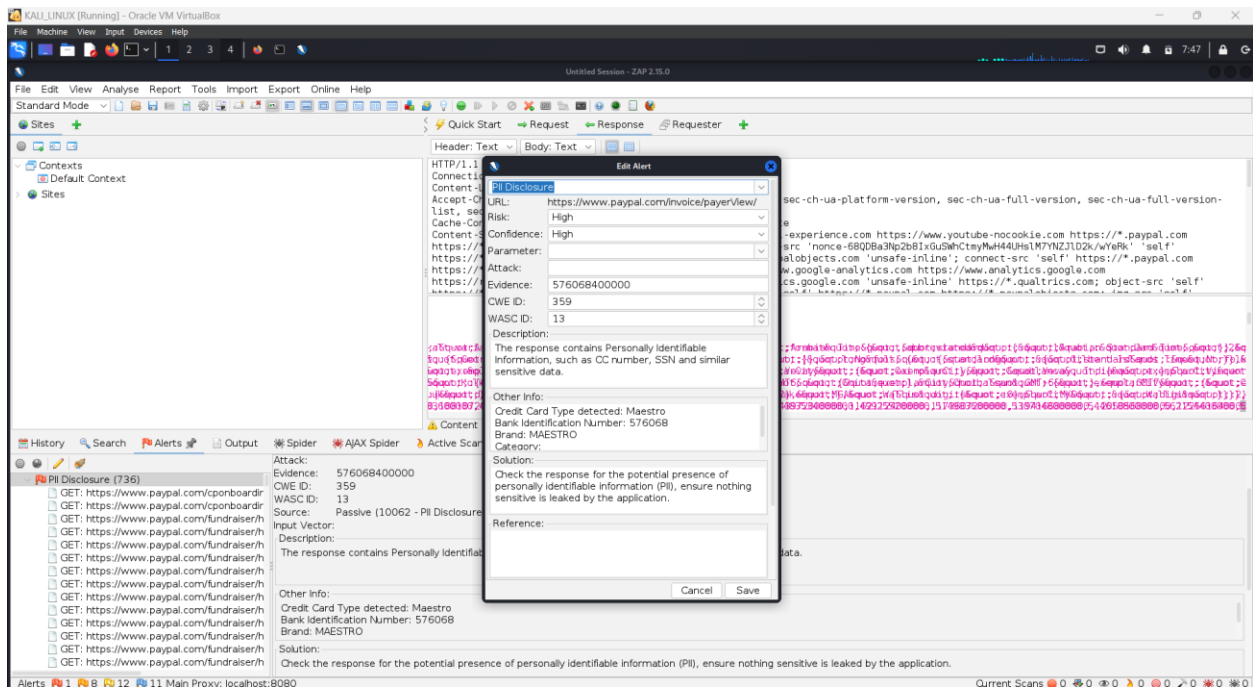
3. Review Backup and Logs Access Backup Systems:

- if possible try to look out for any backups stored containing PIIs in an unscrambled version. Log Review: examine the application logs for risk of possible unintentional leakage of sensitive information. Including user data in error logs.

4. Perform a Security Assessment Analyze the Source Code:

- When a source code repository is available, examine it to understand areas that may contain security flaws (Nt direction restriction, absence of input checking). Control the security measures of the organization: Evaluate their safeguarding measures in terms of risk or security related policies and standards to ascertain areas of vulnerability which may lead to accidental dissemination of PII.

6.1.6 Proof of concept



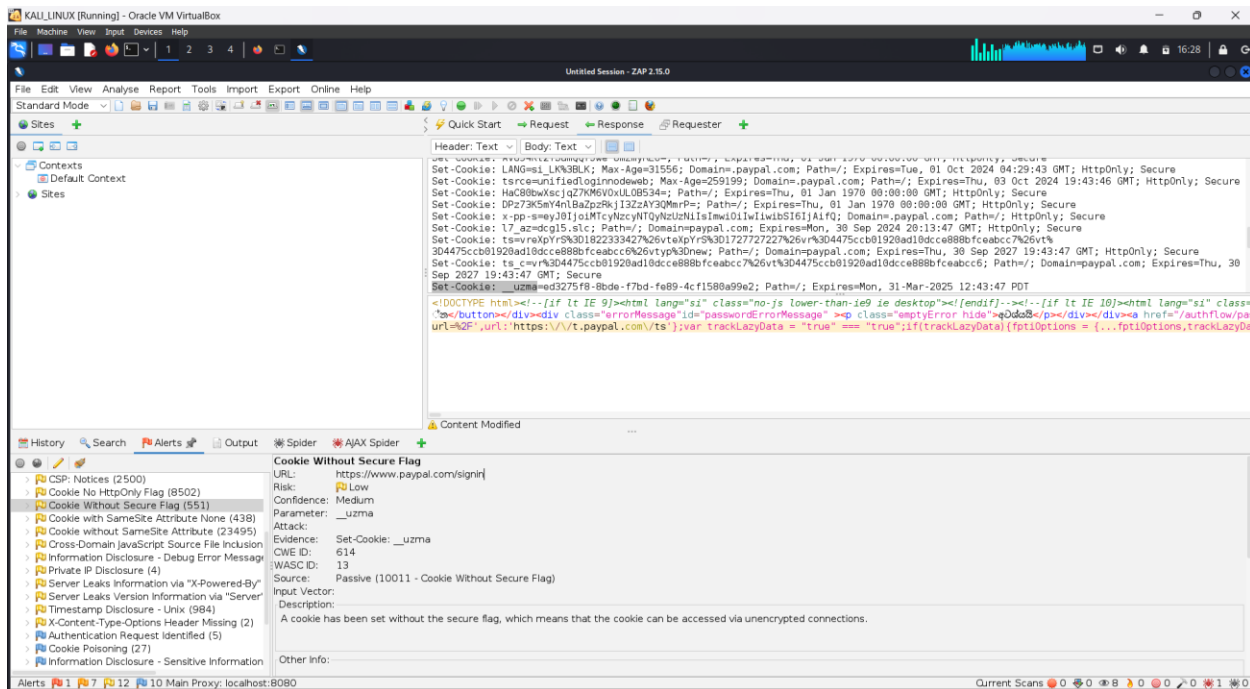
6.1.7 Proposed mitigation or fix

- Check the response for the potential presence of personally identifiable information (PII), ensure nothing sensitive is leaked by the application.
- Implement the strong Access Control.
- Data Encryption.
- Data Minimization.
- Monitor and log Access to sensitive Data.

6.2 Vulnerability

6.2.1 Vulnerability title

- Cookie Without Secure Flag.



6.2.2 Vulnerability description

- A "Cookie Without Secure Flag" vulnerability is a web security flaw in which cookies can be sent insecurely with a non-encrypted channel (for instance, HTTP) because the Secure attribute is not set in the cookie. This further explains this vulnerability.
Cookies are tiny bits of data which are sent to the user's browser by the web server in order to keep track of the activity of the user during a particular session or his/her preferences. Cookies are essential for the functioning of most modern interactive web applications (for instance, to remember the login once it has been done). In cases when cookies are sent with no Secure flag, they can be used in HTTP transmissions as well as HTTPS.

6.2.3 Affected components

- The „Cookie Without Secure Flag” weakness holds important repercussions on the infrastructure of a web application because it emphasizes on the use and control of cookies for various interactions. The most relevant components to this attack include, but not limited to, the following:

1. Web Application Server

The function of the web application is to establish both the presence of cookies and the parameters. Where there is no secure flag set towards the cookies by the application server, this may potentially lead to sending cookies over unsecured connection channels like HTTP.

Affected components:

Session management logic (responsible for creating and managing session cookies): typically defined in same way as routing policies

Authentication systems (cookies used for login sessions or persistent logins): these are the controls used to gain access to the application services

Stateful applications (apps that depend on cookies for keeping the user, their preferences, etc.): involves cookies maintaining the logged in state of the user and their preferences.

2. Browser (Client-Side)

Web browsers are the means through which cookies are stored and sent to the web servers. In the absence of the Secure flag, cookies will be accepted and sent for both HTTP and HTTPS request regardless of any unsecure connection.

Affected components:

At the same time, the browser has to manage any cookies sent by the server above a certain private session level.

Session persistence (where the browser omits the Secure flag while sending session cookies in the presence of non-HTTPS links)

3. Network

In case, such cookies are used over plain HTTP, in-transit cookies become prone to a network attacker thanks to the lack of proper encryption.

Affected components:

Network communications: this includes without limitation instances where interception can take place in unsecure Wi-Fi hotspots, public networks, any environments at odds with packet transmissions such as sniffer attacks, and or man in the middle attacks.

Proxies and intermediaries: Each and every person or add-on in between a client and a server (for example, routers, ISPs) can see the cookies being transmitted.

4. End User Systems

When a cookie lacking the Secure flag has critical session information, it can be grabbed from a system and used to impersonate the valid session. This can facilitate access into the internal user accounts and information.

Affected elements:

User accounts and sessions (by means of taking control of the user's session)

Personal information or preferences saved in Browser Cookies are prone to intruder's attacks.

5. Web Application Firewalls (WAF)

Protecting a web application from a broader range of attacks, WAFs often screen the incoming traffic to look for signs of an attack. However, unless properly set, they may also not enforce safe cookie parameters. Such that, in a situation where a WAF fails to block secure cookie requests, then every other preventive measure to limit insecure cookies use is rendered ineffective.

Affected elements:

Traffic analysis and filtration: If the WAF is not configured to block or alert insecure cookies.

6. Content Delivery Networks (CDNs)

The principle of CDNs relies on enhancing the delivery of content by the storage of files. Sending cookies with neither the Secure flag nor same-site attribute appended means these cookies can travel unencrypted from a client to a CDN, or from a CDN to its provenance.

Affected elements:

Cookie completion via CDN nodes: If in certain instances, it is permissible to use http, these unsecured cookies are at risk of being in the uncensored alter.

7. API Endpoints

Certain APIs will require certain authentication tokens contained in cookies. If these API requests are made via the HTTP, not protected with the Secure attributes, the site can easily leak sensitive tokens contained in the cookies.

Affected elements:

Cookies in api requests containing authentication and session tokens. API security: APIs implementing cookie-based client-server state management could be exposed in case most secure flag is overlooked.

8. Mobile Applications and Hybrid Apps

Certain mobile or hybrid applications (for example those made using Cordova or React Native) may implement cookies for authentication and/or session management and therefore lose the effect of secure cookies. Should the application fail to impose any form of HTTPS connection, the absence of the Secure flag in such cookies would make them susceptible to any possible attacker.

Affected elements:

Management of sessions in mobile applications: If the use of cookies is implicated without adequate safety measures in place.

6.2.4 Impact assessment

- The evaluation of the risk associated with the vulnerability labeled ‘Cookie Without Secure Flag’ focuses on the threat and effects arising from the failure to apply the Secure flag, especially in those scenarios where the cookies are associated with potentially sensitive information such as session identifiers, authentication credentials and user information. Here is a detailed impact assessment:

1. Risk of Intercepting Communications (Man-in-the-Middle Attacks)

Impact: High

Description: In the absence of the Secure flag, such cookies are suited to be transmitted over the non-secured HTTP channel. Any would-be attacker can thus capture these cookies as they do not use encryption and buffer themselves over a network (booth which is unsecure) such as a café or airport and so on.

Consequence: Such sensitive cookies such as session tokens or even authentication tokens can easily be picked up leading to account takeover and access to the user data held within the application.

2. Session Hijacking

Impact: High

Description: When an intruder intercepts a session ID cookie, he is able to access that application under the user’s identity since the intruder possesses the user’s access means.

Consequence: The user’s actions inside the application may be replicated by the attacker, and the attacker may also initiate personal and payment data requests as well as privilege escalation actions depending on the application.

3. Theft of Non-Public Information

Impact: Medium to High

Description: A situation arises where cookies are used to keep such information that includes but is not limited to names, addresses, social security information, and that information that is transmitted without the secure flag is prone to exposure.

Consequence: Consequences of harm in this case are the exposure of private individual’s information that will breach their privacy as well as cause violations to laws such legal ones as GDPR, CCPA or patronage of the company’s image.

4. Legal Non-Conformities

Severity: Severe

Definition: Most of the data protection and privacy laws such as GDPR, HIPAA, PCI-DSS, etc. mandate secure transmission of personal data. The absence of the Secure flag might lead to a violation.

Impact: This may bring about fines, affiliated restrictions and penalties, as well as, most importantly, users' and customers' trust losses.

5. Downgrade vulnerabilities

Severity: Moderate

Definition: Quite a few times, attackers may want to downgrade the communications from HTTPS to HTTP, and this may be the case with cookies. The Secure flag not being set will result in cookies being passed across HTTP even though the session was previously in HTTPS.

Impact: Consequently, this makes the security of that layer of communication less effective and it can allow the transmission of cookies over an unencrypted channel.

6.2.5 Steps to reproduce

1. Open the Web Application

- Visit the target web application that uses cookies to manage sessions or user data.
- Log in or interact with the application to generate cookies (e.g., a session ID).

2. Open Browser Developer Tools

- Press F12 or right-click and select Inspect to open the browser's Developer Tools.
- Navigate to the "Application" tab (or "Storage" in Firefox).
- Under Cookies, find the cookies for the current site.

3. Inspect the Cookies

- Look at the list of cookies stored by the application.
- Check whether the Secure attribute is set for the cookies. If the Secure flag is missing, the cookie is vulnerable to being transmitted over an unencrypted HTTP connection.

4. Switch Between HTTP and HTTPS (if available)

- If the web application is available over both HTTP and HTTPS:
 - Visit the HTTP version of the website (e.g., <http://example.com>).
 - Perform a login or other interaction that generates cookies.
 - Go to the Network tab in Developer Tools and refresh the page.
 - Observe the network requests and confirm whether the cookies are being sent over HTTP.
 - Check if the same cookies are being transmitted over HTTPS (if available).

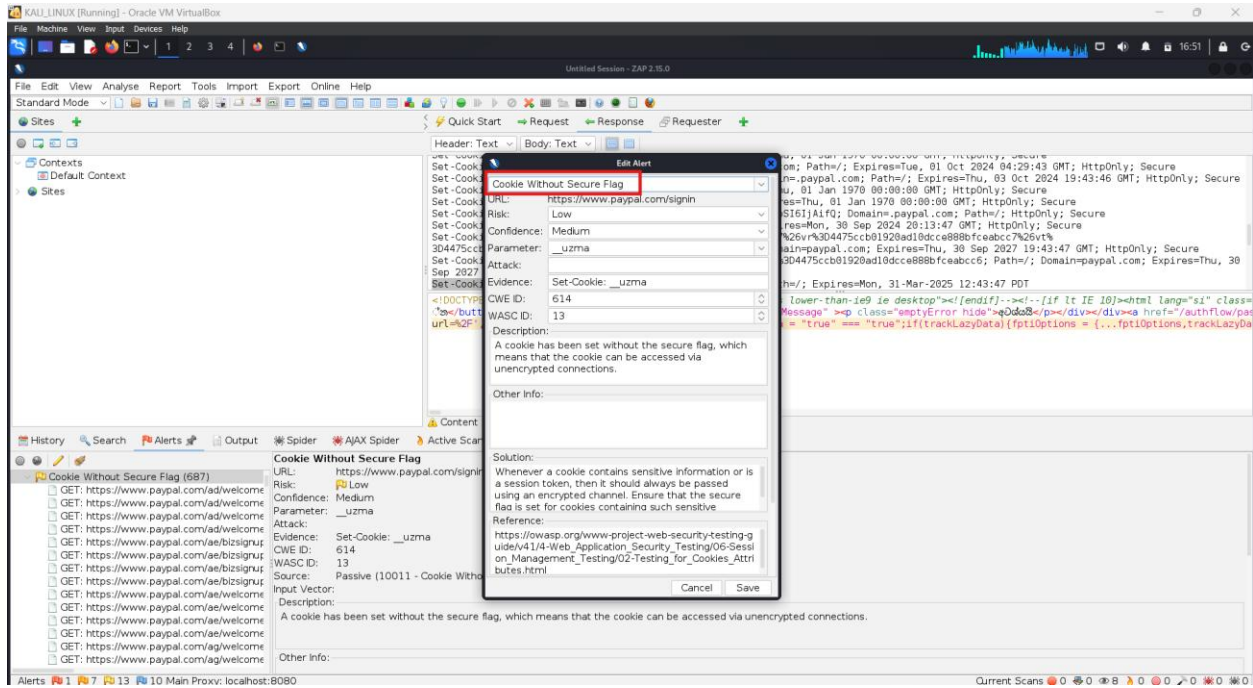
5. Use a Proxy to Capture Network Traffic (Optional)

- Use Burp Suite or OWASP ZAP to intercept and monitor the traffic between the browser and the web server.
- Set up the proxy and configure your browser to route traffic through it.
- Visit the web application via HTTP and log in or perform an action that generates cookies.
- Inspect the HTTP requests in Burp or ZAP, looking for cookies in the request headers.
- Confirm if the cookies are transmitted without the Secure flag in clear text over HTTP.

6. Verify the Risk

- If the Secure flag is missing and the cookies are sent over an HTTP connection, the cookies are exposed to potential interception by an attacker.
- Use a public or unsecured Wi-Fi network to simulate a scenario where an attacker could easily capture the cookies.

6.2.6 Proof of concept



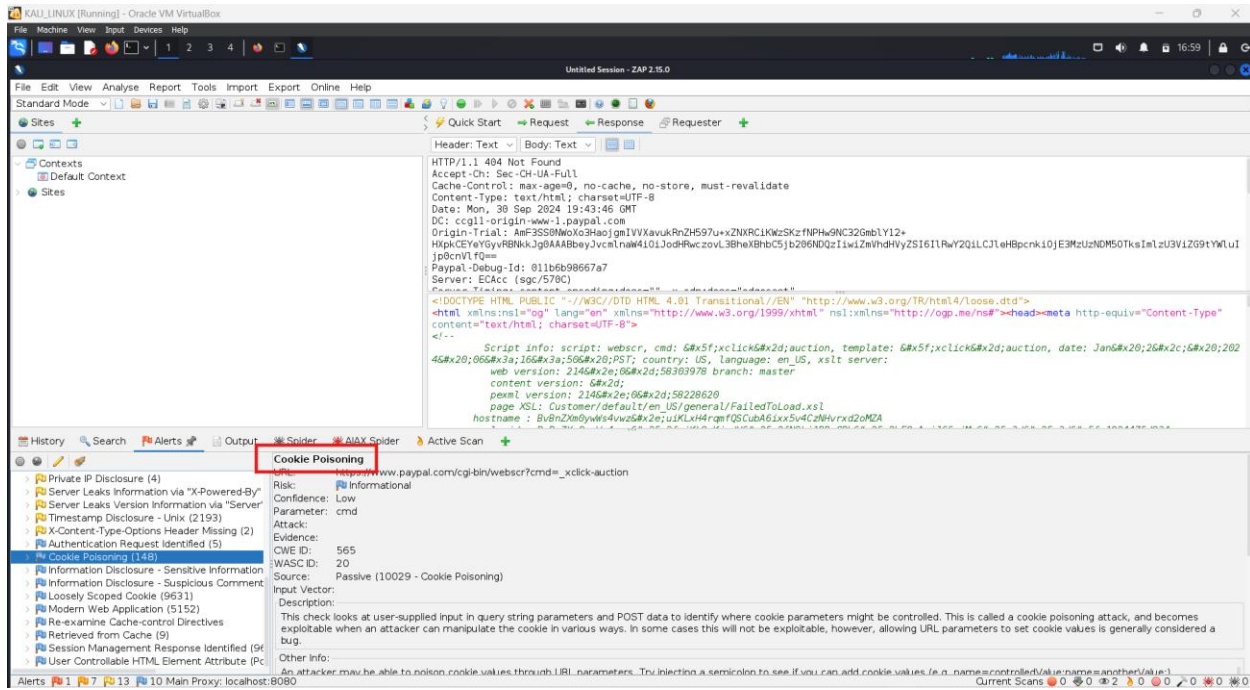
6.2.7 Proposed mitigation or fix

- A session token, then it should always be passed using an encrypted channel. Ensure that the secure flag is set for cookies containing such sensitive information.
 - Enforce HTTPS Connections.
 - Use HttpOnly Flag for Additional Security.
 - Implement the SameSite Flag to Mitigate CSRF Attacks.

6.3 Vulnerability.

6.3.1 Vulnerability title

- Cookie Poisoning.



6.3.2 Vulnerability description

- Cookie Poisoning is a flaw in web security that exists where a person or entity with malicious intent distorts the information held in cookies for nefarious purposes. Cookies are created to store information with respect to a lull in traffic between the client (the browser) and server, hence the ‘state-keeping’ explanation. For instance, they may save pieces of specific information that are unique to a user, for instance, a session id, user preference and even an authentication token. Moreover, if these cookies are designed with no restriction or validation then an evildoer can change the contents and engage in various attacks rendering the web application insecure.

6.3.3 Affected components

- The Cookie Poisoning security flaw pertains to several vital aspects of web application security. Such aspects mostly focus on how cookies are generated, stored, sent, and verified. Some of the areas within applications that are susceptible to cookie poisoning are:

1. Web Application Logic

Description: The main logic structure of the web application gets compromised whenever there is an over-reliance on sessions and cookies to save and authenticate certain private information as session data, user groups, or even preferences.

Impact: In applications where untrusted cookies can be accepted, a malicious user would be able to modify these cookies for their malicious purposes (for instance modifying user's access level or session information).

Affected Functions: Any user/role based feature such as user login, access management, order placement, or else, that is dependent on persistence using cookies.

2. Client-Side Storage (Browser Cookies)

Description: These are files that are kept in the consumer's web browser and as a result they can be easily tampered with. An attacker can change cookie values by either using a browser or a proxy server.

Impact: The client-side storage becomes the battleground for storing objects considered safe in the application caches as long as cookies are not securely stored or encrypted.

Affected Functions: Any cookie saved in a client device, more so those that have content such as session ID, user group or preferences.

3. Cookie Management (Server-Side)

Description: This refers to the web application's server-side part that creates, processes, and sends cookies to the users. As a result, because of the lack of proper management or controls pertaining to security of cookies, the server becomes an easy target for cookie poisoning attacks.

Impact: Users trusted with cookies may sometime neglect to encrypt or sign the cookies before storing them in their web browsers which will be beneficial to malicious users who will so be able to change the cookies for their gains and in turn misuse the server application.

Affected Functions: Session management, authentication, access control, user tracking, personalization settings and so on.

4. Session Management

Description: To allow users to remain logged in between requests, many applications utilize cookies and store a session ID in them. Attackers can use their cookie poisoning techniques to takeover a man's active session including the session data and modify it for his/her advantage.

Impact: Attackers sometimes will modify session based cookies in order to present themselves as target users in order to gain access to otherwise restricted sections, or the even a modified hostile account of the targets.

Affected Functions: Login sessions, user authentication processes, shopping carts and previous orders, and related session based functionalities.

5. Authentication and Authorization Systems

Description: Authentication systems employed in many web applications use cookies in storing user roles or even authentication tokens, which also can be misused to breach authentication or elevate privileges.

Impact: A particular user can evade the policy and present as if an authenticator or an authorization mechanism by changing to a countermeasure user who has higher permissions than eligibility.

Affected Functions: User role privileges, user role verification processes, prosecution codes, and user AP connections that are either saved in cookies or employed as cookies.

6.3.4 **Impact assessment**

1. Unauthorized Access and Privilege Escalation

Description:

- Manipulating datastores in cookies that contain sensitive information such as user roles or session identifiers enables an attacker to manipulate their privileges causing them to gain access to restricted applications. Such attacks, for instance, give the invaders access to administrative features or other user accounts without the authorization of the person whose account it is.

Impact:

High Impact: It is very dangerous for an intruder to assume administrative privileges or to have access to data that is not intended for him. This may result in access of sensitive information, loss of finances and damage to the organisation's image.

Examples:

A person who is an attacker changes a cookie which contains a role such as user to that of an admin thereby enabling such a person to perform administrative tasks.

For being anonymous, changing the session token with the one which is active in order to take control of that particular person's session.

Consequences:

Data breaches which result in exposing of proprietary or sensitive information.

Mission Critical systems are compromised rendering the organization unable to manage the features of its web application.

Costly legal and regulatory fines due to failure to comply with the relevant data protective policies and regulations (e.g. GDPR, HIPAA).

2. Violation of Data Integrity

Description:

In semi-secure or insecure applications where cookies contain crucial data such as shopping cart, pricing, and preferences, the users are allowed to change the contents of the cookies and the data may end up being altered with the intention of committing fraud and causing data corruption.

Impact:

Medium to High Impact: Tolerable data tampering varies depending on what kind of data it is. It can lead to adverse application performance, monetary fraud, or even the abuse of business operations.

Examples:

An attacker accesses a cookie, containing prices of items, and alters it by checking out expensive items at an unreasonably low price.

Malicious editing of a user's profile or preference settings.

Consequences:

Loss of Revenue: Misuse such as fraudulent purchases or changes in the pricing strategy may incur losses that affect e-commerce industries considerably.

Operational Difficulties: Understanding and ensuring critical data is safe, the trust given to the business process will be affected, and the connections in business processes disrupted especially by the users.

3. Session Hijacking and Account Takeover

Introduction:

Session tokens are usually kept inside cookies that can be altered with the help of cookie poisoning. Secondary, session cookies can also be modified or removed by attackers to seize control of active sessions, which enables them to portray the victim without having at hand the knowing credentials.

Impact:

High Impact: The act of hijacking of sessions can sink deeper into account taking over completely and thus the intruders get the ability to carry out activities in the name of the prey.

Examples:

A session cookie is captured or altered, allowing the intruder to take over the user's session without the need for logging in again.

Accessing the user's account with the help of an invalid or expired session cookie to retrieve sensitive or financially related information.

Consequences:

Exploitation of vulnerable databases regarding personal identities, banking records, and other vital services like social services or insurance.

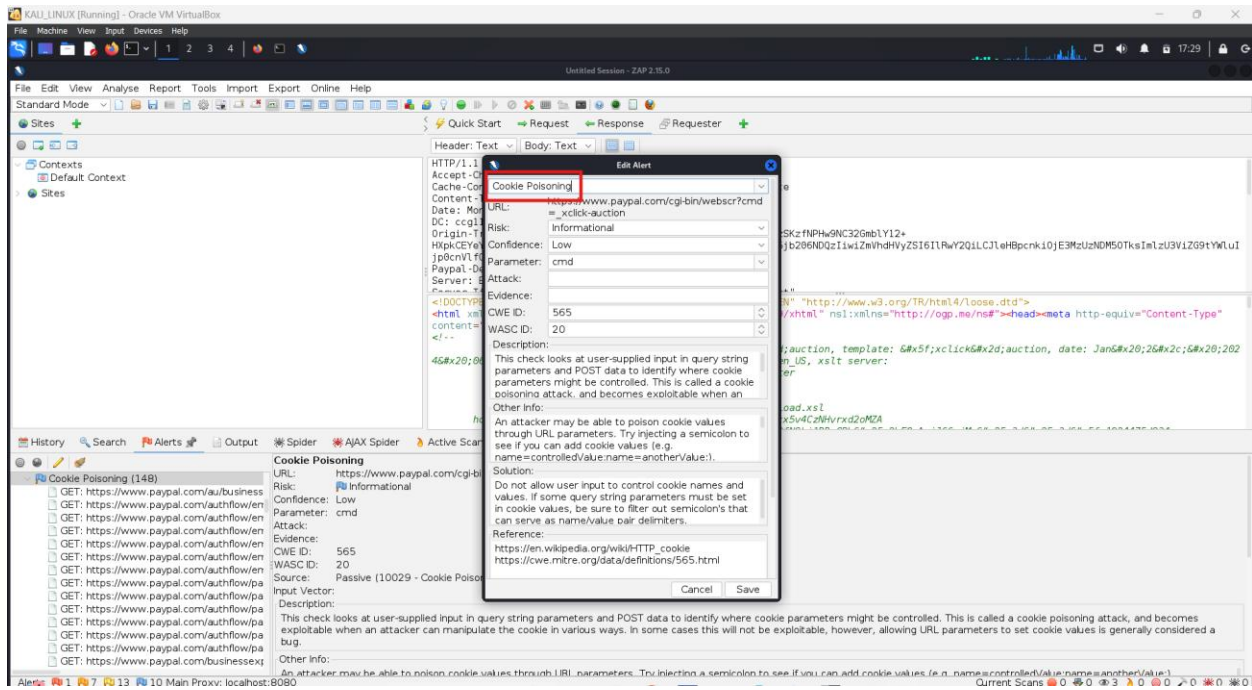
Possible loss of trustworthiness from the current users of the platform, as well as loss of potential new ones.

Fines and lawsuits caused by leakage of personal identifiable information and failure to follow operational guidelines.

6.3.5 Steps to reproduce

1. Identify the Application:
 - Find a web application that uses cookies to store sensitive information (e.g., user roles, session IDs, pricing).
2. Inspect Cookies:
 - Use browser Developer Tools (F12) to check the cookies set by the application under the Application tab.
3. Select Target Cookie:
 - Identify a cookie that can be manipulated, such as a user role cookie (userRole) or a session ID cookie.
4. Modify the Cookie:
 - Option 1: Directly edit the cookie value in Developer Tools (double-click the cookie and change its value).
 - Option 2: Use a proxy tool (like Burp Suite) to intercept and modify the cookie in an HTTP request.
5. Reload the Application:
 - Refresh the webpage or navigate to another part of the application that relies on the modified cookie.
6. Observe Changes:
 - Check if the application's behavior has changed (e.g., gaining unauthorized access, altered pricing, or session hijacking).

6.3.6 Proof of concept



6.3.7 Proposed mitigation or fix

- Do not allow user input to control cookie names and values. If some query string parameters must be set in cookie values, be sure to filter out semicolon's that can serve as name/value pair delimiters.
- Implement Cookie Security Attributes.
- Validate cookie data on the server side.
- Avoid Storing sensitive data in Cookies.
- Keep software and Libraries updated.