APPLICATION VULNERABIILITY SCANNER REPORT

XYZ INC

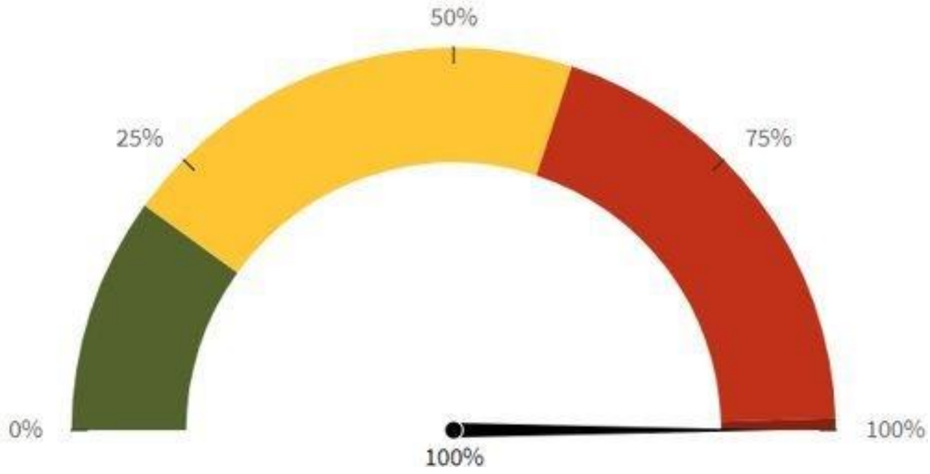
< DATE OF ATTACK>

radware

BAD BOT VULNERABIILITY SCAN

POTENTIAL VULNERABILITIES ON YOUR PLATFORM

BOT RISK-O-METER



Note: *We use proxy IP Addresses worldwide with our BVS tool to conduct scans for comprehensive results and to avoid any sort of GEO blocking.*

Generation Based Attacks

Attacks Made	Requests Bypassed/Made	Risk Level
Generation 1	**150/150**	**High**
Generation 2	**150/150**	**High**
Generation 3	**148/150**	**High**
Generation 4	**150/150**	**High**

Use Case Based Attacks

Attacks Made	Requests Bypassed/Made	Risk Level
Account Takeover	**100/100**	**High**
Fake Registration	**50/50**	**High**
Form Spam	**110/110**	**High**
Content Scraping	**517 Products Scrapred**	**High**

BOT GENERATIONS: TYPICAL CHARACTERISTICS

Generation 1 Bots

Blacklisting IP, User Agent

- Scripted Bots (Tools like Curl, Wget & Requests Package are used).
- Data Center IP Address is used in a uniform programmatic pattern.
- User Agent & IP Address is typically not spoofed.

Generation 2 Bots

Headless Browser, Maintain Cookies

- Headless Browser Bots (Tools like Puppeteer or Selenium used).
- Bots coupled with spoofed user agents typically bypass generic WAF solution.

Generation 3 Bots

Botnet Attack (shallow)

- Distributed IPs alternating multiple user agents, this combination makes it difficult to fingerprint and detect.
- Botnets are typically used to perform the attack.

Generation 4 Bots

Interactions & Intent (Deep)

- Human-like Bots typically distributed and make minimal hits to remain well below the radar.
- Mimic/replicate human traversal across sections that are difficult to differentiate.

The purpose of this scan is to find vulnerabilities of your website which may expose your content to different types of bot attacks. The data will not be used in any way that may harm or hinder your business activity.

GENERATION 1 : ATTACK VECTORS

BVS Attack A Details:

Targeted URL: https://<URL_1>/en-in/ambidextrous
Check your Server Logs / Time: 2025-01-28 11:46:56 UTC
Attack Vectors: Data Center IP's

How did your site respond to bots generated from cURL ?
Attack Penetrated Application: YES

How did your site respond to bots generated from Requests?
Attack Penetrated Application: YES

How did your site respond to bots generated from wget?
Attack Penetrated Application: YES

VULNERABILITY STATUS: HIGH RISK

BVS Attack A Overview:

Total Requests Made: 150
Requests Penetrated: 150

Request Name	User Agent	IP Address
		35.239.172.232
Curl Request	Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.6.6880.81 Safari/537.36	50
Requests Package	Mozilla/5.0 (Windows NT 10.0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36	50
wGet Request	Mozilla/5.0 (Linux; Android 10; U325AC Build/QP1A.190711.020) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.6045.67 Mobile Safari/537.36	50

GENERATION 2 : ATTACK VECTORS

BVS Attack B Details:

Targeted URL: https://<URL_1>/en-in/sets/adidas-x-moon-boot

Check your Server Logs / Time: 2025-01-28 11:47:19 UTC

Attack Vectors: Spoofing UA's and Proxy IP

How did your site respond to bots generated from Selenium ?
Attack Penetrated Application:
YES

How did your site respond to bots generated from PhantomJS?
Attack Penetrated Application:
YES

PhantomJS

Puppeteer

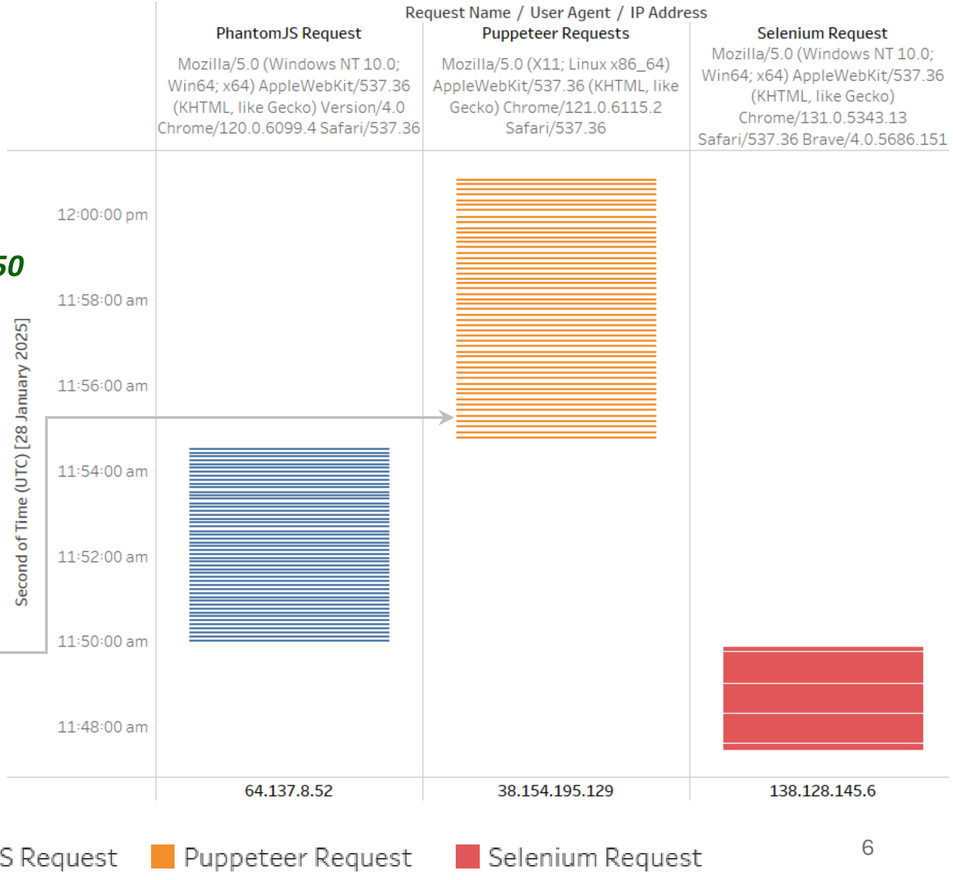
How did your site respond to bots generated from Puppeteer?
Attack Penetrated Application:
YES

VULNERABILITY STATUS: HIGH RISK

BVS Attack B Overview:

Total Requests Made: 150

Requests Penetrated: 150



Each line represents an attack in the mentioned time frame.

<DATE OF ATTACK>

radware

GENERATION 3 : ATTACK VECTORS

BVS Attack C Details:
URL Targeted: https://<URL_1>/en/it/makingoftheicon.html
Check your Server Logs/Time:
2025-01-28 11:48:40 UTC
Request Frequency: 1 Sec

Total Requests Made: 150
Requests Penetrated: 150

Attack Description:

Mentioned to the right is a depiction of attack scenario performed through our internal tool. Basically, **300 requests** are to be made alternating between a fixed User Agents at a fixed interval of time. Attacks would be targeted using headless browsers like **PhantomJS, Puppeteer** or **Selenium.**

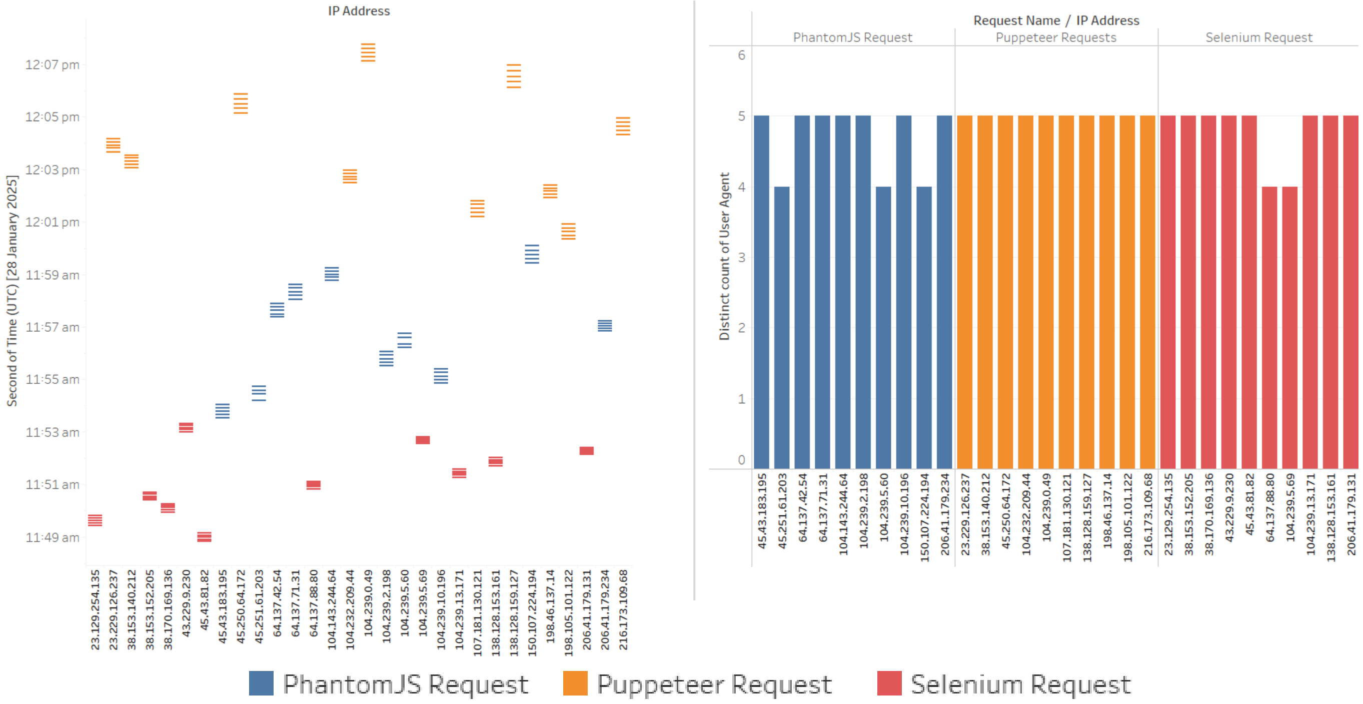
How did your site respond to bots generated from Selenium ?
Requests Penetrated: YES

How did your site respond to bots generated from Puppeteer?
Requests Penetrated: YES

How did your site respond to bots generated from PhantomJS?
Requests Penetrated: YES

VULNERABILITY STATUS: HIGH RISK

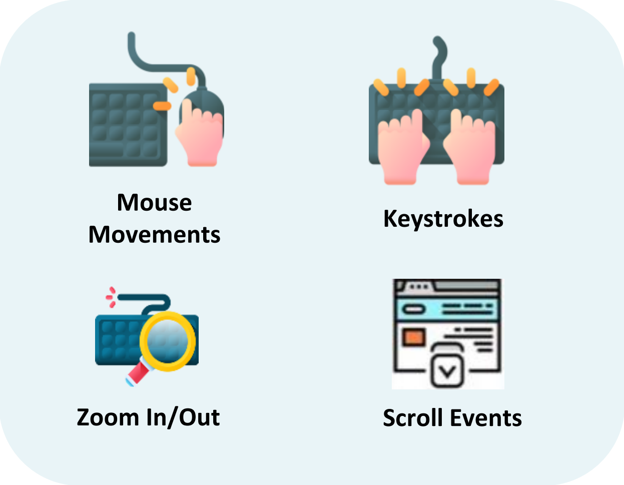
BVS Attack C Overview:
Total Requests Made: 300
Requests Penetrated: 300



GENERATION 4 : ATTACK VECTORS

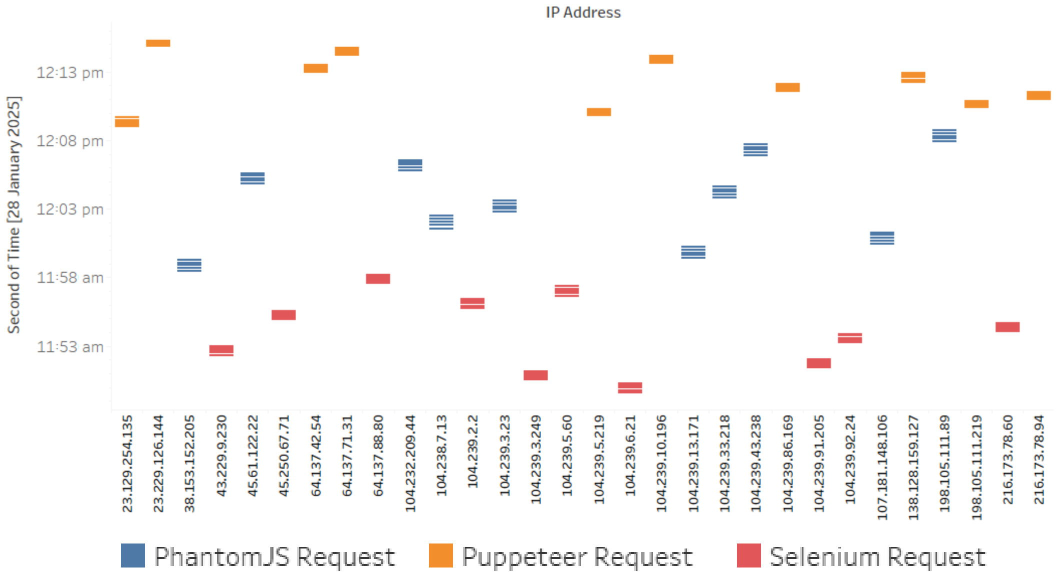
BVS Attack D Details:

Following Tests have been conducted to evaluate if your site is able to block bots that execute JavaScript events having a ***Randomized movement, Element Clicks*** and ***Keystrokes*** that resemble human behavior.



No of Attack Requests: 150
Time of Attack: 2024-12-19 12:58:05 UTC
Attack Vector: Distributed IPs
Attack Penetrated Application: YES

VULNERABILITY STATUS: HIGH RISK



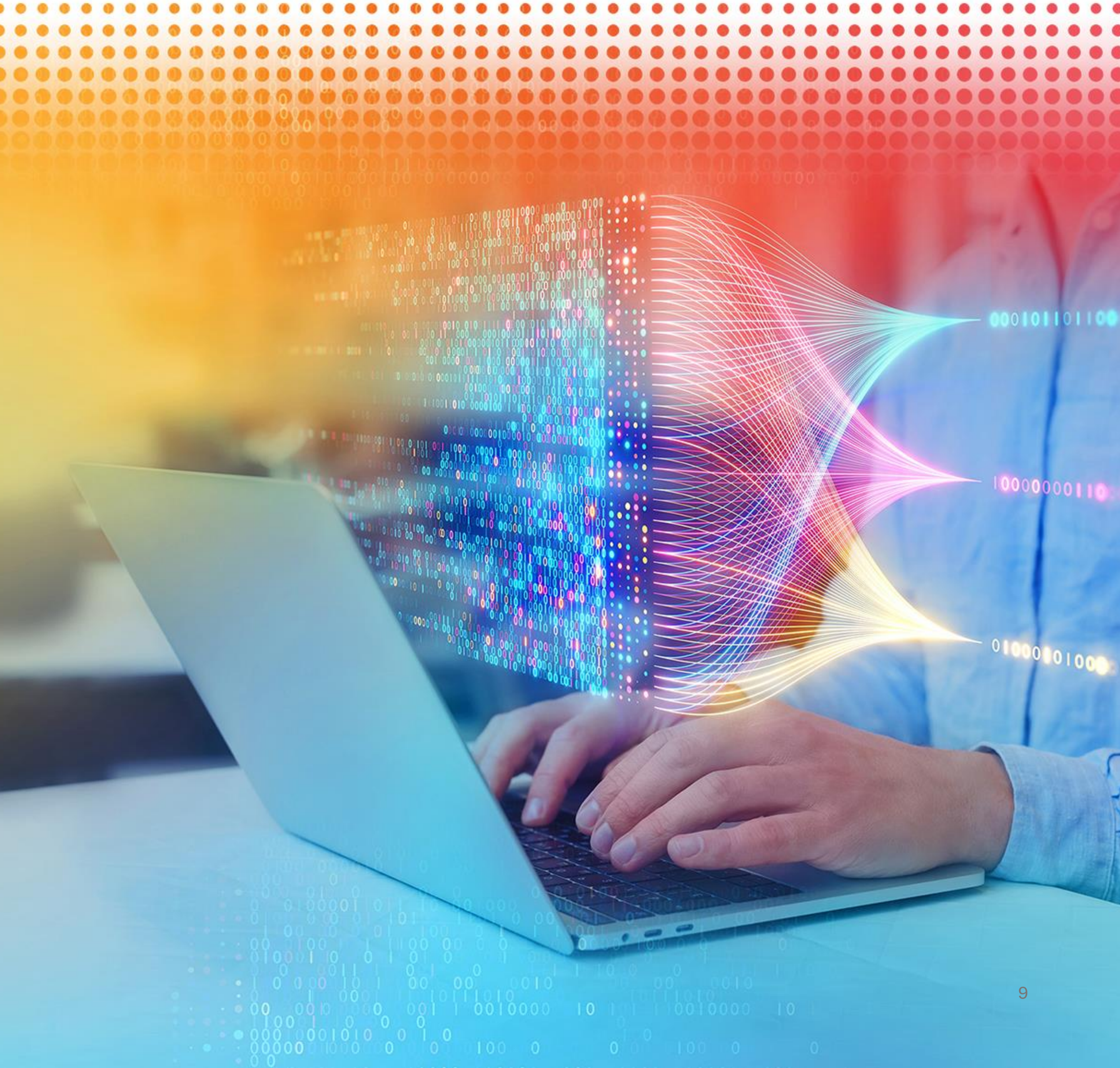
Attack D Overview:

URL Attempt:
https://<URL_1/en/it/login
Check your Server Logs/Time:
2025-01-28 11:49:21 UTC
Total Requests Made: 150 Requests
Requests Penetrated: 150 Requests

Generation 4 Bots hit your servers at *periodic/timed intervals* (using Distributed IPs & User Agents) with minimum requests to ensure the requests are not captured/traced for bot signature creation. Also, Gen4 Bots exhibit '*human like behavior'* considering their website navigation and traversal.

radware

USE CASE ANALYSIS



ACCOUNT TAKEOVER ATTACK: (100 FAKE CREDENTIALS)

Targeted URL: https://<URL_1>/en-in/account/login
Time Period of Attack (Span): 28-01-2025 11:33:54 UTC -> 28-01-2025 12:41:48 UTC

OWASP Automated Threats exposed:
OAT-007 Credential Cracking , OAT-008 Credential Stuffing

X-Path Details
Username: //input[@id='loginForm-email']
Password: //input[@id='loginForm-password']
Submit Button: //button[@data-test='loginForm-submitButton']

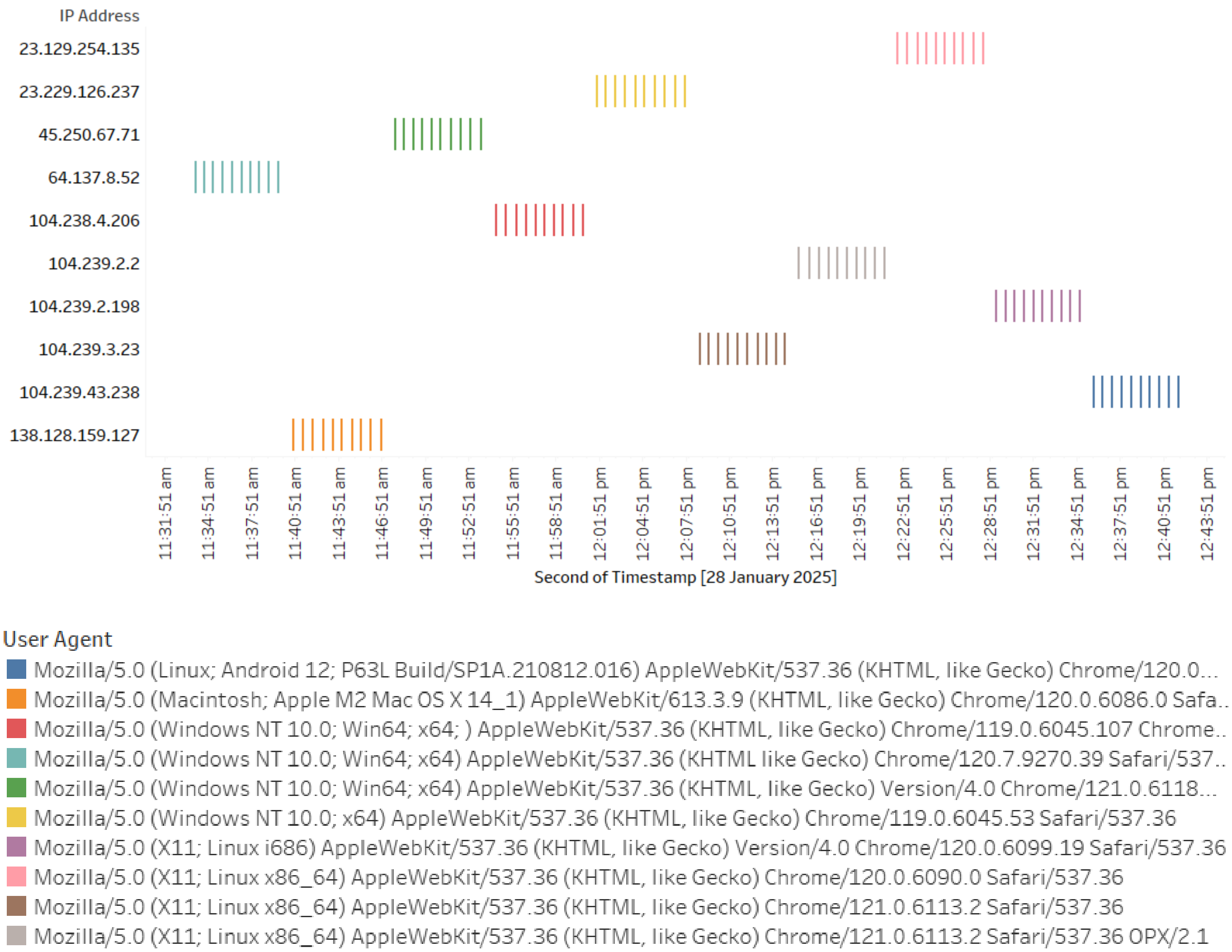
Combination of User Agents used: 10 User Agents (Spoofed User Agent)
Combination of Distributed IPs used: 10 IP Address
No of Hits Bypassed / Requests Made: **100 Bypassed/ 100 Requests Made**

SAMPLE ATTACK INSTANCE

Time	Username	Password
28-01-2025 11:33	yallison@gallagher.biz	Xm62Ztul#^
28-01-2025 11:34	dominique33@palmer-dickson.com	$KrVSxxnz1
28-01-2025 11:35	dylan70@mills-campbell.org	+z7V&o91^!
28-01-2025 11:35	paula76@davis-taylor.com	$kjd&LVF96
28-01-2025 11:36	vmitchell@silva.org	Hh6PFHg6k^
28-01-2025 11:37	christopher97@lewis-dominguez.org	GkG*6Emjf*
28-01-2025 11:37	lgraham@chavez-hunter.com	#3#(4YxO)D
28-01-2025 11:38	waltersanne@simpson.biz	^B#SoEhk66

ACCOUNT TAKEOVER ATTACK: IN-DEPTH

Account Takeover Attack – Overview (Attack Span: 52 Minutes)



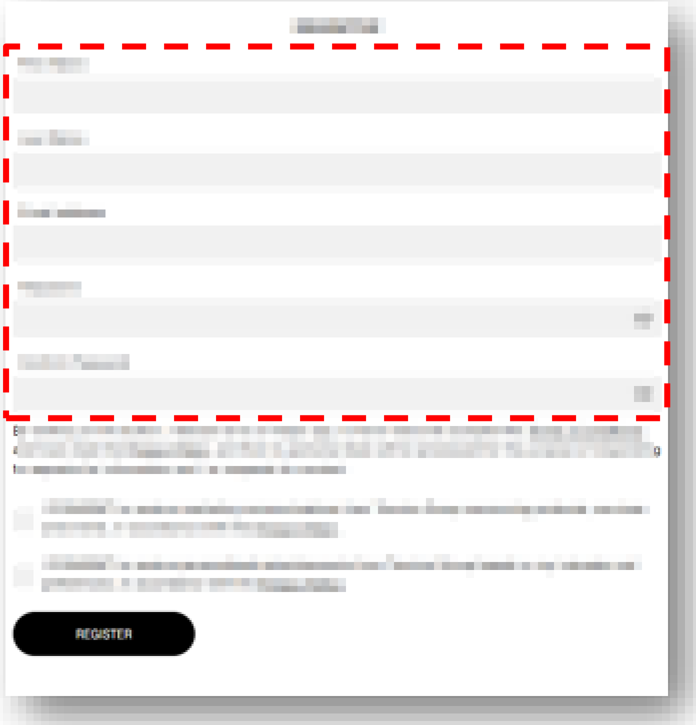
- Each line represents a Request/Hit to your server.
- Colour Code of the line indicates the User Agent used for Every Hit.

FAKE REGISTRATION ATTEMPT:
50 FAKE CREDENTIALS SUBMITTED!

The purpose of this attack is to demonstrate the potential impact on your infrastructure in the event of a sudden surge in bot-driven account creations, resulting in a massive influx of fake accounts.

Targeted URL: https://<URL_1>/en-in/account/login
Time Period of Attack (Span): 28-01-2025 11:45:09 UTC -> 28-01-2025 12:27:52 UTC

OWASP Automated Threats exposed: OAT-019 Account Creation



X-Path Details

First Name:
//input[@id='registerForm-firstName’]
Last Name:
//input[@id='registerForm-lastName’]
Email:
//input[@id='registerForm-email’]
Password:
//input[@id='registerForm-password’]
Confirm Password:
//input[@id='registerForm-passwordConfirmation']
Submit Button:
//button[@data-test='registerForm-submitButton']

SAMPLE ATTACK INSTANCE

First Name	Last Name	Email	Password
Linda	Martin	shannonhernandez@hotmail.com	%6Wkyrss78
Chad	Hernandez	schmidttracy@gmail.com	o2*8KlAd_)
Cody	Goodman	angelacosta@hill.com	hXjhYHGj#5
Gina	Clark	tsmith@hotmail.com	^*Q0GOnh+T

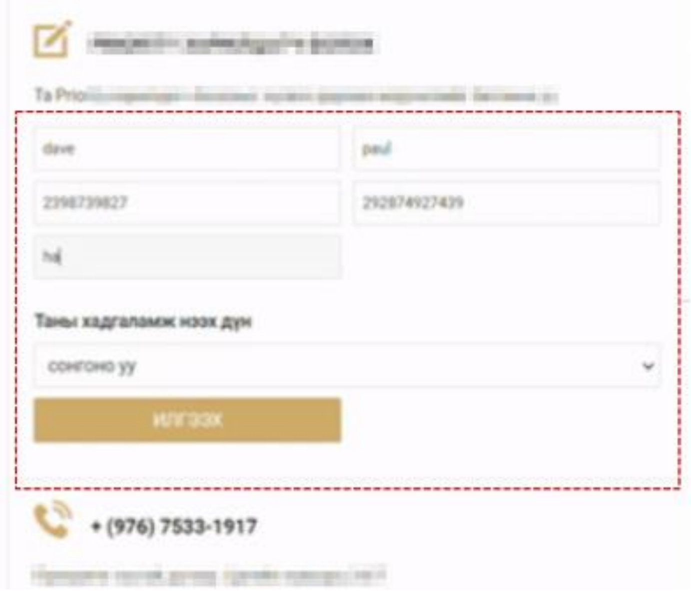
Attack Instances: 50 Requests
Attack Vectors: Generation 3 (Selenium - with Human Like Behavior)

Combination of User Agents used: 11 User Agents (Spoofed User Agents)
Combination of Distributed IPs used: 11 IP Address
No of Hits Bypassed / Requests Made: 50 Bypassed/ 50 Requests Made

FORM SPAM ATTEMPT: 50 FAKE REQUESTS SUBMITTED ON ‘CONTACT SUPPORT’ PAGE

Targeted URL: https://<url_1>/en-in/legal/contact-us
Time Period of Attack (Span): 28-01-2025 11:38:49 UTC -> 28-01-2025 12:19:52 UTC

OWASP Automated Threats exposed: OAT-017 Spamming



[redacted]

Ta Prio[redacted]

dave	paul
2398739827	292874927439
hd	

Таны хадгаламж нээх дүн

сонгоно уу

ИЛГЭЭХ

+ (976) 7533-1917

[redacted]

X-Path Details

Full Name:
//input[@id='contactForm-name’]

Email:
//input[@id='registerForm-email’]

Enquiry reason (Product Information):
//select[@id='contactForm-subject']/option[2]

Message
//textarea[@id='contactForm-message’]

Submit Button:
//button[@id='contactForm-submitButton']

Attack Instances: 50 Requests
Attack Vectors: Generation 3 (Selenium - with Human Like Behavior)

Combination of User Agents used: 10 User Agents (Spoofed User Agents)
Combination of Distributed IPs used: 10 IP Address
No of Hits Bypassed / Requests Made: **50 Bypassed/ 50 Requests Made**

SAMPLE ATTACK INSTANCE

Full Name	Email	Message
ssanchez	phillipmcbride@hotmail.com	Economic activity scene admit else.
justinmoreno	ogregory@king.com	Set board look hour everybody require often.
ashley36	jackcampbell@wong.info	These while drive area visit cell glass.
brian74	vincentrita@arnold-ford.net	Board decade hot those.
wgray	anthony92@hotmail.com	Article trouble parent environment look.

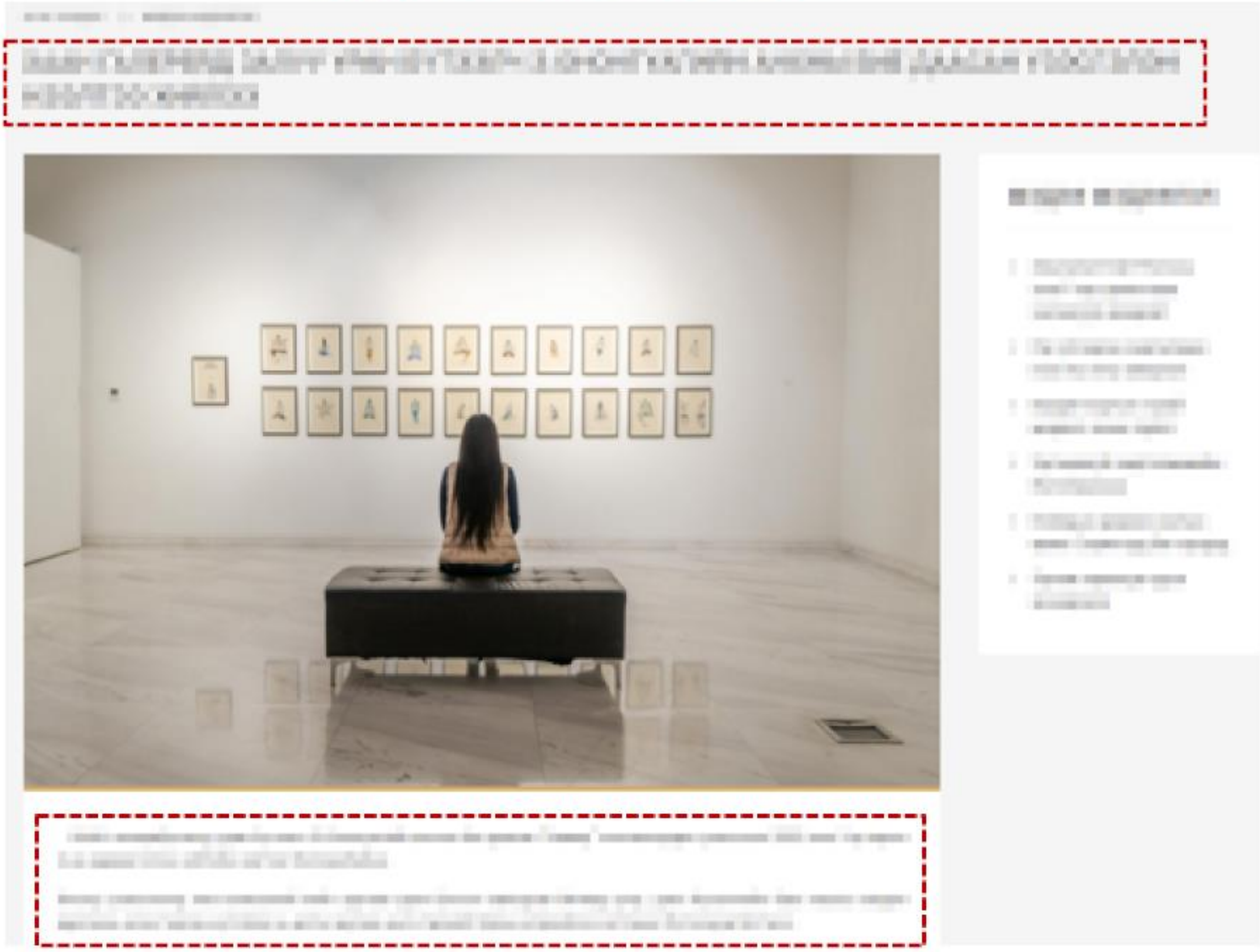
CONTENT SCRAPING:
517 PRODUCT DETAILS SCRAPED!

URL Scraped

- https://<URL_1>/en-in/sets/new-in,
- https://<URL_1>/en-in/shopping/woman,
- https://<URL_1>/en-in/shopping/man,
- https://<URL_1>/en-in/shopping/kid

OWASP Automated Threats exposed: OAT-011 Scraping

Highlighted Items were scraped!



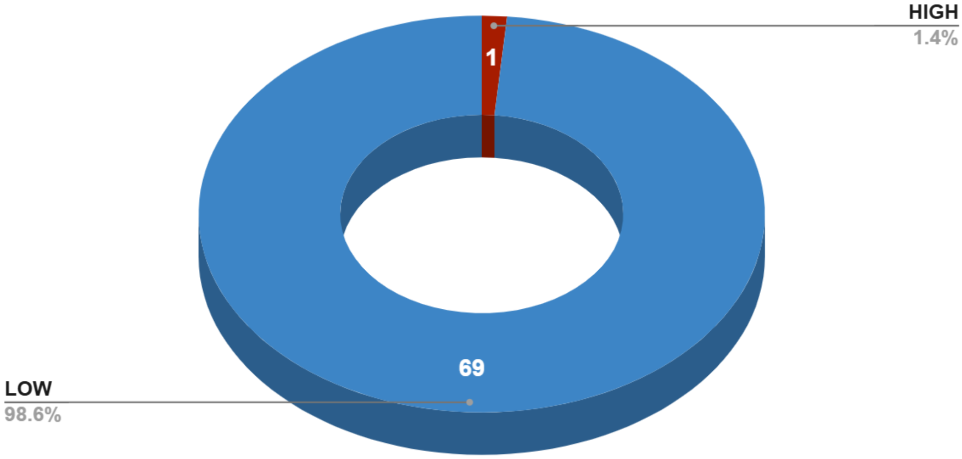
API VULNERABIILITY SCAN

SECURITY RISK IDENTIFIED ON YOUR PLATFORM

API Security Attacks performed

Highlighted risks were API-related security risks found on your platform, as defined by the Open Web Application Security Project (OWASP).

70 VULNERABLE EVENTS FOUND FROM YOUR API's BASED ON SEVERITY



Identified Vulnerabilities as per OWASP Top 10 API Security Risks (2023)

- ❑ API1:2023 - Broken Object Level Authorization
- ❑ API2:2023 - Broken Authentication
- ❑ API3:2023 - Broken Object Property Level Authorization
- ❑ API4:2023 - Unrestricted Resource Consumption
- ❑ API5:2023 - Broken Function Level Authorization
- ❑ API6:2023 - Unrestricted Access to Sensitive Business Flow
- ❑ API7:2023 – Server-Side Request Forgery
- ❑ API8:2023 - Security Misconfiguration
- ❑ API9:2023 - Improper Inventory Management
- ❑ API10:2023 - Unsafe Consumption of APIs

ESSENTIAL RESPONSE HEADERS

(CWE-693: Protection Mechanism Failure)

SECURITY IMPACT:
LOW

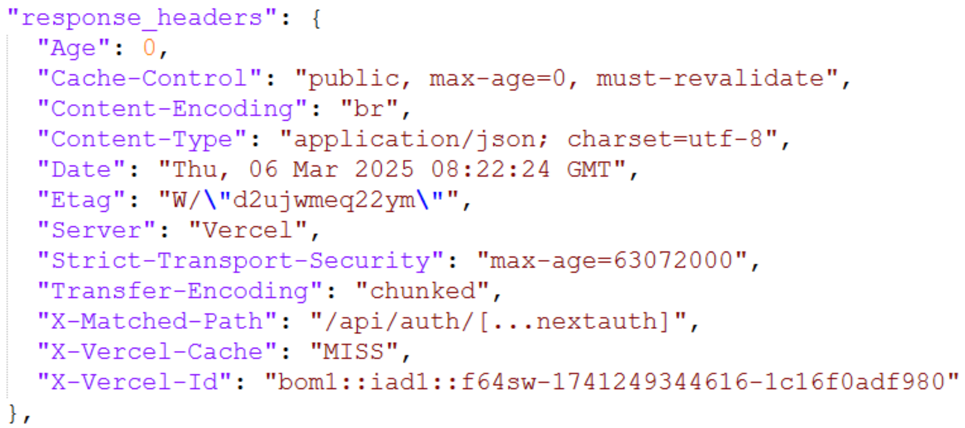
IMPACTED OWASP TOP 10 API SECURITY:

API8:2023 - Security Misconfiguration

“absence of certain standard HTTP headers in the response sent by a web server to a client's request. These headers provide important information about the content, security, and behavior of the response. The absence of critical headers can lead to incorrect rendering of content, caching problems etc.”

Test Performed: Check if Response Headers have important Information about content, Security and behaviour.

RESPONSE RECEIVED WITH MISSING STANDARD HEADERS



```
"response_headers": {
  "Age": 0,
  "Cache-Control": "public, max-age=0, must-revalidate",
  "Content-Encoding": "br",
  "Content-Type": "application/json; charset=utf-8",
  "Date": "Thu, 06 Mar 2025 08:22:24 GMT",
  "Etag": "W/\"d2ujwmeq22ym\"",
  "Server": "Vercel",
  "Strict-Transport-Security": "max-age=63072000",
  "Transfer-Encoding": "chunked",
  "X-Matched-Path": "/api/auth/[...nextauth]",
  "X-Vercel-Cache": "MISS",
  "X-Vercel-Id": "bom1::iad1::f64sw-1741249344616-1c16f0adf980"
},
```

Missing Standard Headers in Response Headers

- ***Clear-Site-Data***
- ***Content-Security-Policy***
- ***Cross-Origin-Embedder-Policy***
- ***Cross-Origin-Opener-Policy***
- ***Cross-Origin-Resource-Policy***
- ***Permissions-Policy***
- ***Pragma***
- ***Referrer-Policy***
- ***Strict-Transport-Security***
- ***X-Content-Type-Options***
- ***X-Frame-Options***
- ***X-Permitted-Cross-Domain-Policies***

ESSENTIAL RESPONSE HEADERS

(CWE-693: Protection Mechanism Failure)

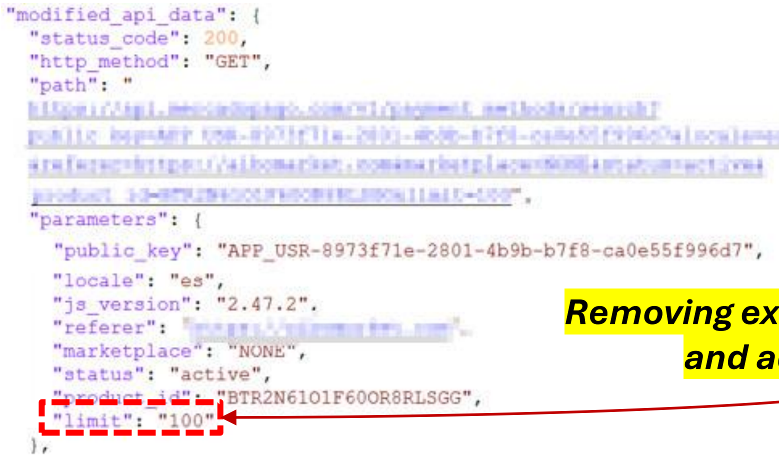
SECURITY IMPACT: HIGH

IMPACTED OWASP TOP 10 API SECURITY: API4:2023 - Unrestricted Resource Consumption

Pagination misconfiguration occurs when a web application fails to allocate adequate resources or enforce restrictions on pagination requests, leading to excessive server load. This can result in resource exhaustion, degraded performance, or service disruptions due to uncontrolled pagination activity.’

Test Performed: Modify request payload with *“Modified payload with limit:100"* Checking if request parameter containing higher payload is processed.

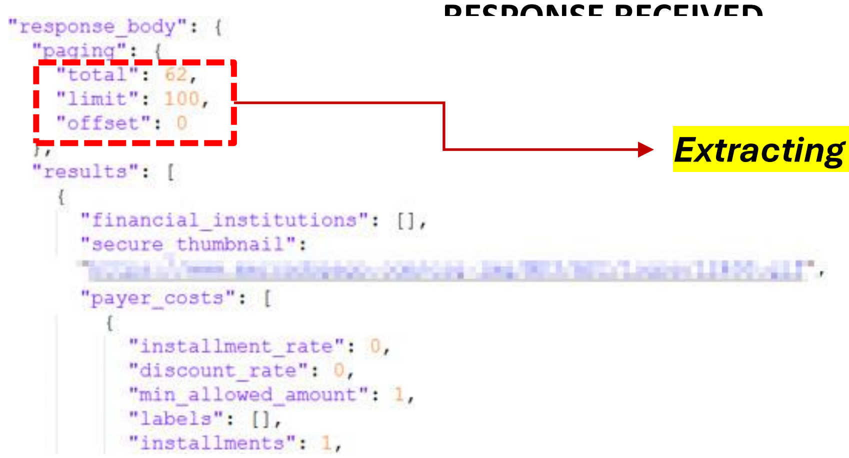
MODIFIED REQUEST



```
"modified_api_data": {
  "status_code": 200,
  "http_method": "GET",
  "path": "
  [redacted]",
  "parameters": {
    "public_key": "APP_USR-8973f71e-2801-4b9b-b7f8-ca0e55f996d7",
    "locale": "es",
    "js_version": "2.47.2".
    "referer": [redacted],
    "marketplace": "NONE",
    "status": "active",
    "product_id": "BTR2N61O1F60OR8RLSGG",
    "limit": "100"
  },
```

Removing existing payload(limit=1) and adding limit=100

RESPONSE RECEIVED



```
"response_body": {
  "paging": {
    "total": 62,
    "limit": 100,
    "offset": 0
  },
  "results": [
    {
      "financial_institutions": [],
      "secure thumbnail":
      [redacted],
      "payer_costs": [
        {
          "installment_rate": 0,
          "discount_rate": 0,
          "min_allowed_amount": 1,
          "labels": [],
          "installments": 1,
```

Extracting data from all 62 pages

radware

BENEFITS OF IMPLEMENTING RADWARE BOT MANAGER

Prevent Credential stuffing and brute force attacks that are used to gain unauthorized access to customer accounts

Prevent bad bots from generating fake accounts on a massive scale

Prevent malicious bots from deluging online marketplaces and community forums with spam leads and comments

Prevent bad bots from scraping valuable, proprietary content and illegally reproduced on ghost websites or repurposed by competitors

BENEFITS OF IMPLEMENTING RADWARE API SECURITY

Prevent server outages (Denial of Service) and associated monetary loss

Prevent undetected, unauthorized access which could provide administrative access to exploiters.

Prevent Data Theft & Sensitive Data Exposure

Protection against any type of API Attack on your platform.

Protection against account hijacking, PII Information exfiltration or credential leakage.

Improve Customer Trust and confidence.

ANALYSTS PRAISE US

GigaOm Radar for Application and API Security

2024 Application and API Security Leader

Quadrant Knowledge Solutions SPARK Matrix™

2024 DDoS Mitigation Leader

KuppingerCole WAF Leadership Compass

2024 WAF Leader

Gartner® Peer Insights™

Voice of the Customer for Cloud WAAP, 2024

Aite-Novarica

Best-in-Class Provider for Bot Management

INDUSTRY’S WIDEST SET OF COMPLIANCE STANDARDS

EU GDPR	EU General Data Protection Regulation
PCI-DSS	Payment Card Industry Data Security Standard
HIPAA	Health Insurance Portability and Accountability Act
US SSAE16	SOC-1 Type II, SOC-2 Type II
ISO 27001	Information Security Management Systems
ISO 27017	Information Security for Cloud Services
ISO 27018	Information Security Protection of PII in public clouds
ISO 27701	Privacy Information Management for PII controllers and processors
ISO 27032	Security Techniques -- Guidelines for Cybersecurity
ISO 28000	Specification for Security Management Systems for the Supply Chain

THANK YOU