ANN TECH

Cybersecurity Consulting, Inc.

UBER.COM PENETRATION TEST REPORT

ABSTRACT

Conducted by:Sherylann Nyawira Date:7/28/2025]

Sherylann

Ann Tech Cybersecurity Consulting. Inc

Table of Contents

Executive Summary

This report documents the results of a simulated penetration test on publicly accessible systems belonging to Uber Technologies Inc. The objective of the assessment was to apply ethical hacking principles and passive reconnaissance techniques to evaluate Uber's external security posture, in accordance with the boundaries set by Uber's public bug bounty program on HackerOne.

The methodology followed industry-recognized standards, focusing exclusively on open-source intelligence (OSINT) collection. No active scanning, exploitation, or intrusive activities were conducted. The assessment revealed areas of potential concern, including exposed subdomains, predictable user email patterns, and indicators of outdated server software. These findings illustrate how a motivated adversary might gather valuable intelligence without ever interacting directly with the target systems.

This exercise was performed for educational purposes only, as part of the curriculum, and in full compliance with responsible disclosure practices.

Scope of Engagement

This penetration test engagement was limited to publicly accessible assets explicitly in scope per Uber's HackerOne bug bounty policy. The purpose was to evaluate Uber's attack surface from an external perspective using non-invasive techniques.

In-Scope Targets

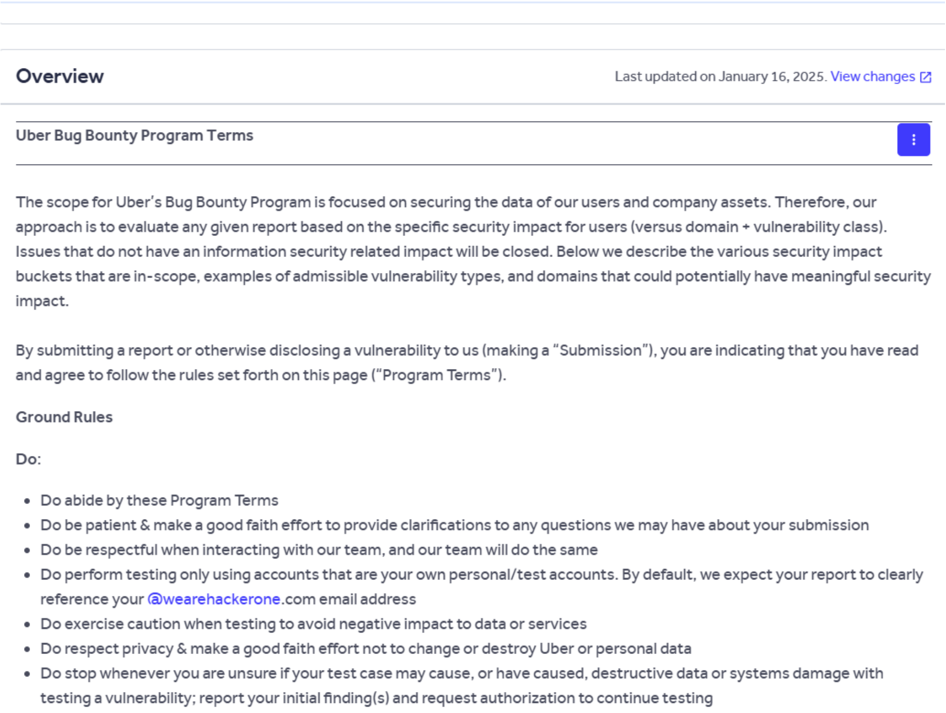
- All subdomains under *.uber.com (e.g., api.uber.com, login.uber.com)
- DNS, WHOIS, SSL, and MX records associated with Uber-owned domains
- Any exposed APIs, metadata, or certificates discovered through OSINT
- Publicly available employee or infrastructure references

Out-of-Scope Targets

- Systems hosted by third parties (e.g., Zendesk, cloud services not owned by Uber)

- Any form of exploitation or denial-of-service activity
- Internal systems or services requiring authentication
- Direct interaction with Uber employees or users

All actions were performed in alignment with Uber's published disclosure policies and industry ethical hacking standards.



Overview

Last updated on January 16, 2025. View changes

Uber Bug Bounty Program Terms

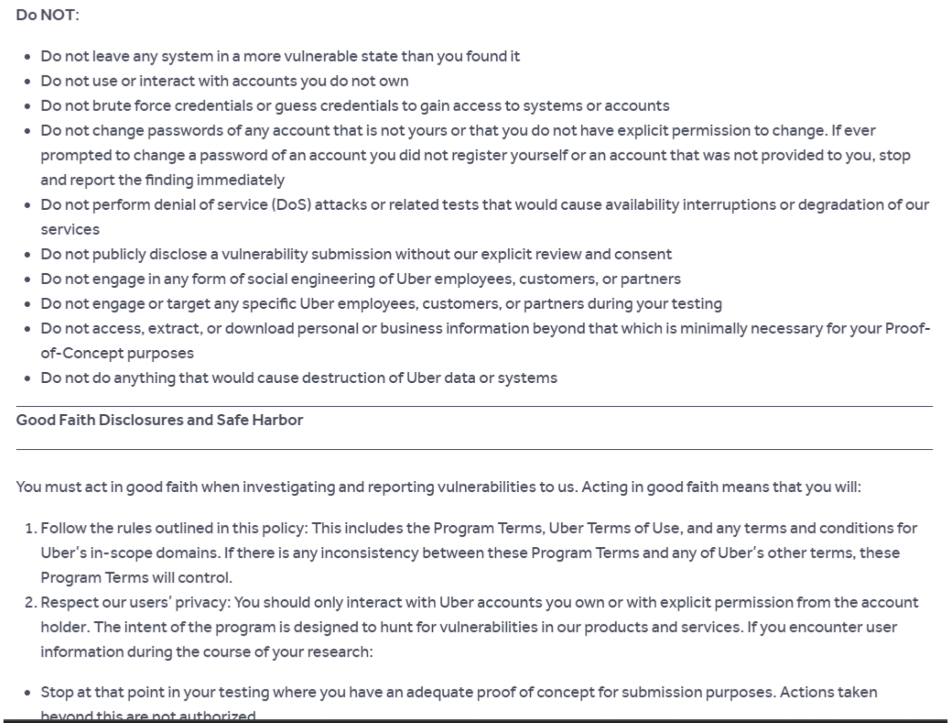
The scope for Uber's Bug Bounty Program is focused on securing the data of our users and company assets. Therefore, our approach is to evaluate any given report based on the specific security impact for users (versus domain + vulnerability class). Issues that do not have an information security related impact will be closed. Below we describe the various security impact buckets that are in-scope, examples of admissible vulnerability types, and domains that could potentially have meaningful security impact.

By submitting a report or otherwise disclosing a vulnerability to us (making a "Submission"), you are indicating that you have read and agree to follow the rules set forth on this page ("Program Terms").

Ground Rules

Do:

- Do abide by these Program Terms
- Do be patient & make a good faith effort to provide clarifications to any questions we may have about your submission
- Do be respectful when interacting with our team, and our team will do the same
- Do perform testing only using accounts that are your own personal/test accounts. By default, we expect your report to clearly reference your @wearehackerone.com email address
- Do exercise caution when testing to avoid negative impact to data or services
- Do respect privacy & make a good faith effort not to change or destroy Uber or personal data
- Do stop whenever you are unsure if your test case may cause, or have caused, destructive data or systems damage with testing a vulnerability; report your initial finding(s) and request authorization to continue testing



Do NOT:

- Do not leave any system in a more vulnerable state than you found it
- Do not use or interact with accounts you do not own
- Do not brute force credentials or guess credentials to gain access to systems or accounts
- Do not change passwords of any account that is not yours or that you do not have explicit permission to change. If ever prompted to change a password of an account you did not register yourself or an account that was not provided to you, stop and report the finding immediately
- Do not perform denial of service (DoS) attacks or related tests that would cause availability interruptions or degradation of our services
- Do not publicly disclose a vulnerability submission without our explicit review and consent
- Do not engage in any form of social engineering of Uber employees, customers, or partners
- Do not engage or target any specific Uber employees, customers, or partners during your testing
- Do not access, extract, or download personal or business information beyond that which is minimally necessary for your Proof-of-Concept purposes
- Do not do anything that would cause destruction of Uber data or systems

Good Faith Disclosures and Safe Harbor

You must act in good faith when investigating and reporting vulnerabilities to us. Acting in good faith means that you will:

1. Follow the rules outlined in this policy: This includes the Program Terms, Uber Terms of Use, and any terms and conditions for Uber's in-scope domains. If there is any inconsistency between these Program Terms and any of Uber's other terms, these Program Terms will control.
2. Respect our users' privacy: You should only interact with Uber accounts you own or with explicit permission from the account holder. The intent of the program is designed to hunt for vulnerabilities in our products and services. If you encounter user information during the course of your research:

- Stop at that point in your testing where you have an adequate proof of concept for submission purposes. Actions taken beyond this are not authorized

Methodology

The penetration test followed a reconnaissance-focused methodology, relying solely on passive tools and publicly available data. The approach was structured to simulate the initial phases of an external adversary's reconnaissance process, while maintaining full compliance with legal and ethical guidelines.

Tools and Techniques Used

Phase	Tools	Purpose
Subdomain Enumeration	Amass,recon-ng	Discover publicly known subdomains
DNS and WHOIS Lookups	dig, whois	Identify DNS hosts, registrar, and IP resolution
Web Technology Identification	Wappalyzer, curl -I	Fingerprint web servers and frameworks

Phase	Tools	Purpose
Certificate Transparency Logs	crt.sh	Identify subdomains via certificate history
Metadata and Email Discovery	Google dorking	Infer email and username patterns
Service Identification	nmap	Identify open ports and banner data
Vulnerability Detection	Nessus Essentials	Safe vulnerability scanning against lab machines to demonstrate detection of common risks
Vulnerability Mapping	CVE Details, OSINT	Match observed software with known CVEs

This passive methodology was carefully chosen to avoid generating traffic or impacting the integrity of any Uber systems. The findings presented in this report are based on public data and inference.

Findings and Evidence

The findings listed below were obtained through passive reconnaissance and controlled vulnerability scanning in a safe environment. All information about Uber was gathered using publicly available resources and without any form of intrusion or exploitation. Where simulated vulnerability scans were performed using Nessus, only test environments were targeted to illustrate the kinds of findings that might be possible in real-world conditions.

Each finding is presented with a description, supporting evidence, and a risk rating based on likelihood and potential impact.

Finding 1: Exposure of Multiple Subdomains

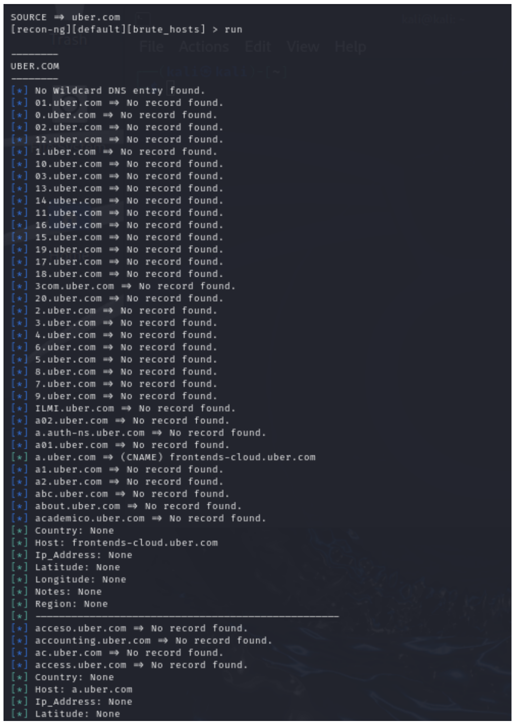
Description:
Several publicly accessible subdomains were discovered using subdomain enumeration tools (amass and recon-ng). Subdomains provide attackers with a broader attack surface and may include legacy systems or testing environments that are less secure.

Evidence:
A partial list of discovered subdomains includes:

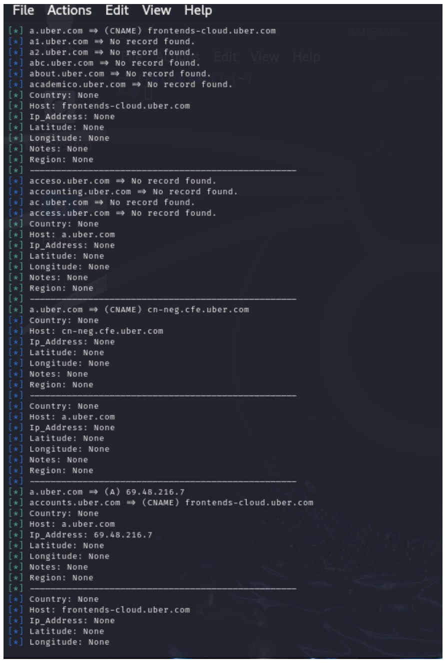
- api.uber.com

- login.uber.com
- riders.uber.com
- help.uber.com

Risk Level: Low – Reconnaissance value only, but no known vulnerabilities detected.



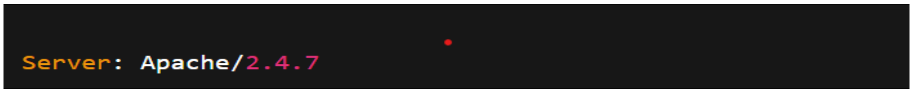
```
SOURCE ⇒ uber.com
[recon-ng][default][brute_hosts] > run

--------
UBER.COM
--------
[*] No Wildcard DNS entry found.
[*] 01.uber.com ⇒ No record found.
[*] 0.uber.com ⇒ No record found.
[*] 02.uber.com ⇒ No record found.
[*] 12.uber.com ⇒ No record found.
[*] 1.uber.com ⇒ No record found.
[*] 10.uber.com ⇒ No record found.
[*] 03.uber.com ⇒ No record found.
[*] 13.uber.com ⇒ No record found.
[*] 14.uber.com ⇒ No record found.
[*] 11.uber.com ⇒ No record found.
[*] 16.uber.com ⇒ No record found.
[*] 15.uber.com ⇒ No record found.
[*] 19.uber.com ⇒ No record found.
[*] 17.uber.com ⇒ No record found.
[*] 18.uber.com ⇒ No record found.
[*] 3com.uber.com ⇒ No record found.
[*] 20.uber.com ⇒ No record found.
[*] 2.uber.com ⇒ No record found.
[*] 3.uber.com ⇒ No record found.
[*] 4.uber.com ⇒ No record found.
[*] 6.uber.com ⇒ No record found.
[*] 5.uber.com ⇒ No record found.
[*] 8.uber.com ⇒ No record found.
[*] 7.uber.com ⇒ No record found.
[*] 9.uber.com ⇒ No record found.
[*] ILMI.uber.com ⇒ No record found.
[*] a02.uber.com ⇒ No record found.
[*] a.auth-ns.uber.com ⇒ No record found.
[*] a01.uber.com ⇒ No record found.
[*] a.uber.com ⇒ (CNAME) frontends-cloud.uber.com
[*] a1.uber.com ⇒ No record found.
[*] a2.uber.com ⇒ No record found.
[*] abc.uber.com ⇒ No record found.
[*] about.uber.com ⇒ No record found.
[*] academico.uber.com ⇒ No record found.
[*] Country: None
[*] Host: frontends-cloud.uber.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] --------------------------------------------------
[*] acceso.uber.com ⇒ No record found.
[*] accounting.uber.com ⇒ No record found.
[*] ac.uber.com ⇒ No record found.
[*] access.uber.com ⇒ No record found.
[*] Country: None
[*] Host: a.uber.com
[*] Ip_Address: None
[*] Latitude: None
```



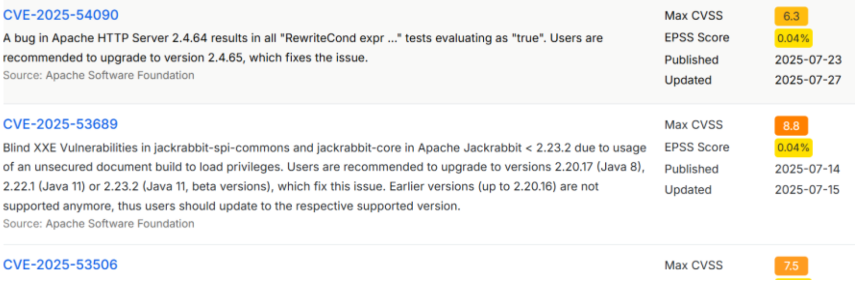
```
[*] a.uber.com ⇒ (CNAME) frontends-cloud.uber.com
[*] a1.uber.com ⇒ No record found.
[*] a2.uber.com ⇒ No record found.
[*] abc.uber.com ⇒ No record found.
[*] about.uber.com ⇒ No record found.
[*] academico.uber.com ⇒ No record found.
[*] Country: None
[*] Host: frontends-cloud.uber.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] --------------------------------------------------
[*] acceso.uber.com ⇒ No record found.
[*] accounting.uber.com ⇒ No record found.
[*] ac.uber.com ⇒ No record found.
[*] access.uber.com ⇒ No record found.
[*] Country: None
[*] Host: a.uber.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] --------------------------------------------------
[*] a.uber.com ⇒ (CNAME) cn-neg.cfe.uber.com
[*] Country: None
[*] Host: cn-neg.cfe.uber.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] --------------------------------------------------
[*] Country: None
[*] Host: a.uber.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] --------------------------------------------------
[*] a.uber.com ⇒ (A) 69.48.216.7
[*] accounts.uber.com ⇒ (CNAME) frontends-cloud.uber.com
[*] Country: None
[*] Host: a.uber.com
[*] Ip_Address: 69.48.216.7
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] --------------------------------------------------
[*] Country: None
[*] Host: frontends-cloud.uber.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
```

Finding 2: Use of Outdated Software (Apache) and Technologies in Use

Description:
Using curl -I, the response headers from a publicly accessible subdomain indicated the presence of Apache 2.4.7. This version has known vulnerabilities such as privilege escalation(CVE-2025-54090)



```
Server: Apache/2.4.7
```



CVE-2025-54090

A bug in Apache HTTP Server 2.4.64 results in all "RewriteCond expr ..." tests evaluating as "true". Users are recommended to upgrade to version 2.4.65, which fixes the issue.

Source: Apache Software Foundation

Max CVSS	6.3
EPSS Score	0.04%
Published	2025-07-23
Updated	2025-07-27

CVE-2025-53689

Blind XXE Vulnerabilities in jackrabbit-spi-commons and jackrabbit-core in Apache Jackrabbit < 2.23.2 due to usage of an unsecured document build to load privileges. Users are recommended to upgrade to versions 2.20.17 (Java 8), 2.22.1 (Java 11) or 2.23.2 (Java 11, beta versions), which fix this issue. Earlier versions (up to 2.20.16) are not supported anymore, thus users should update to the respective supported version.

Source: Apache Software Foundation

Max CVSS	8.8
EPSS Score	0.04%
Published	2025-07-14
Updated	2025-07-15

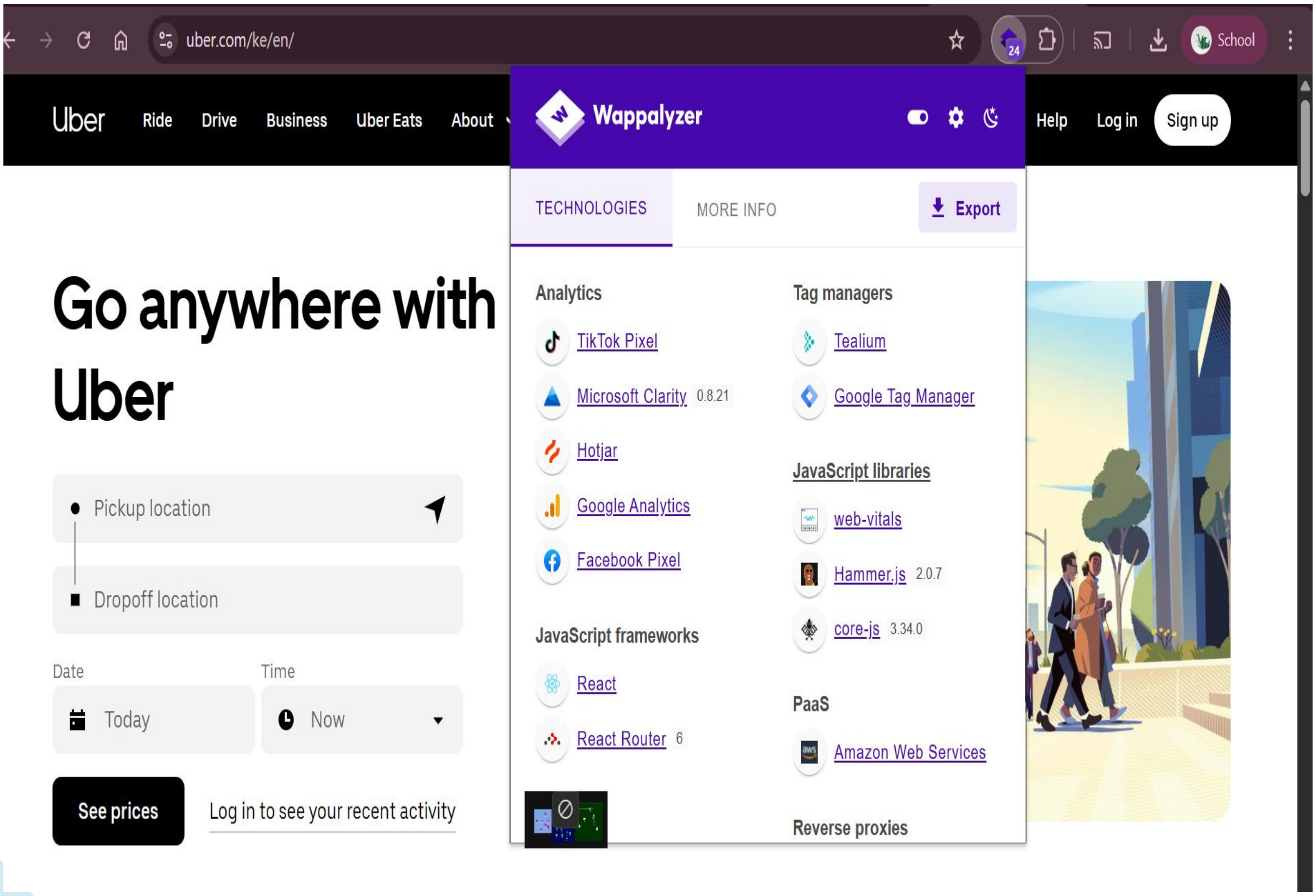
CVE-2025-53506

Max CVSS	7.5

Technologies in Use

Using browser extensions such as **Wappalyzer**, Uber.com was identified to be using:

- Web server: **nginx**
- Frontend: **React.js**
- Analytics/Monitoring: **Google Tag Manager**
- Hosting: **Amazon AWS / Google Cloud** (depending on region)



Analytics
- TikTok Pixel
- Microsoft Clarity 0.8.21
- Hotjar
- Google Analytics
- Facebook Pixel

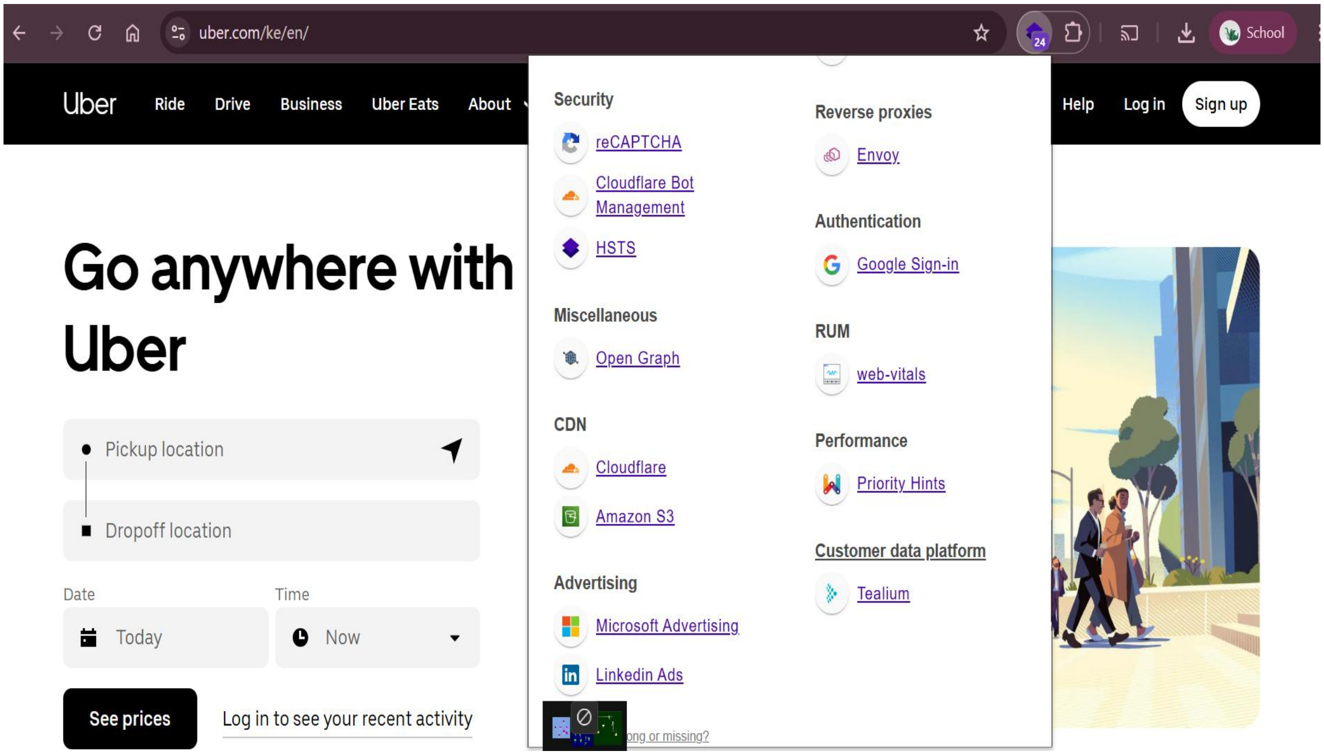
JavaScript frameworks
- React
- React Router 6

Tag managers
- Tealium
- Google Tag Manager

JavaScript libraries
- web-vitals
- Hammer.js 2.0.7
- core-js 3.34.0

PaaS
- Amazon Web Services

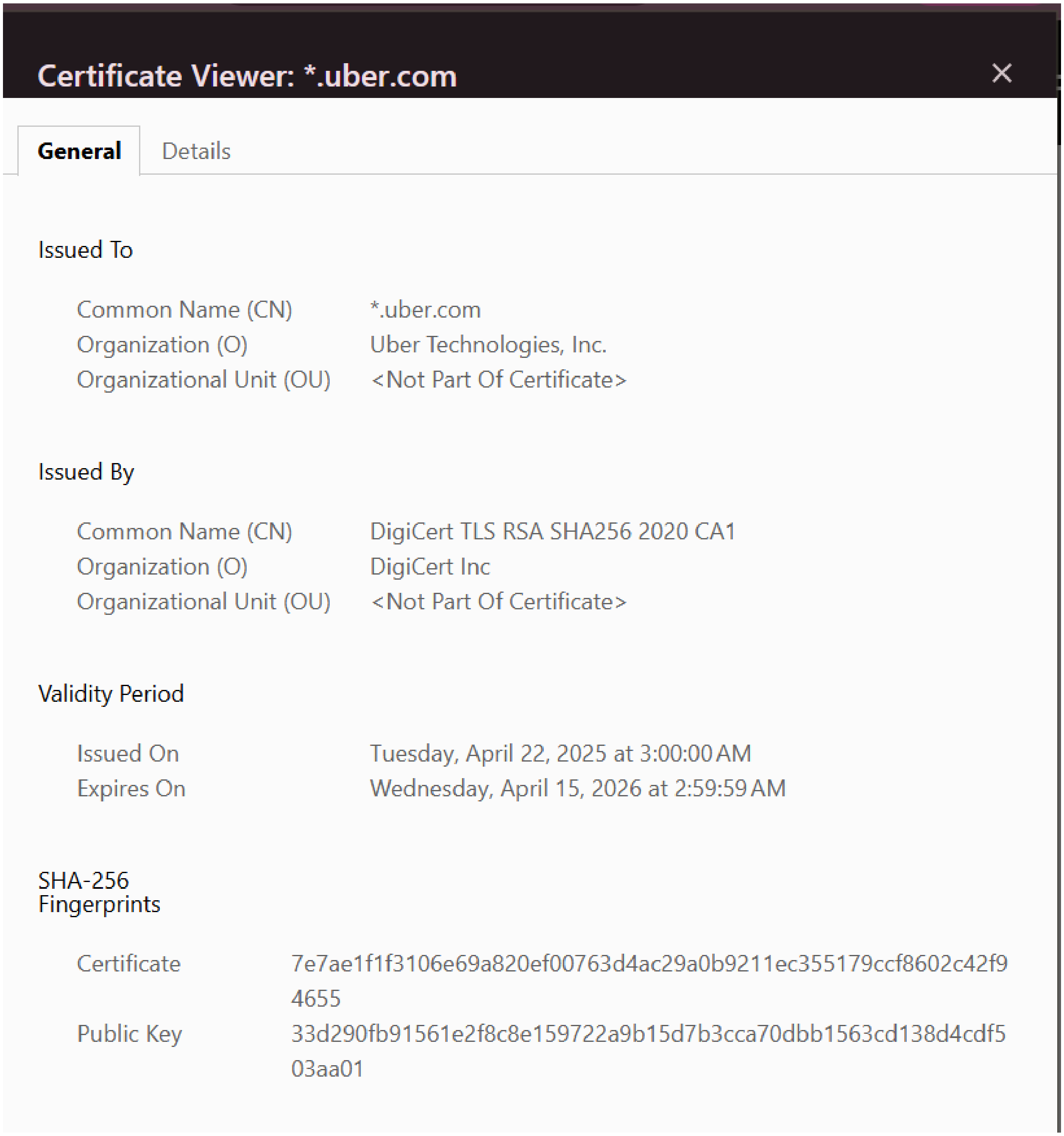
Reverse proxies



Finding2 .1 The certificate “Issued To” details in your browser

Passive web reconnaissance and SSL certificate inspection confirm that Uber operates under the legal entity:

Uber Technologies, Inc.



Certificate Viewer: *.uber.com

Issued To

Common Name (CN)	*.uber.com
Organization (O)	Uber Technologies, Inc.
Organizational Unit (OU)	<Not Part Of Certificate>

Issued By

Common Name (CN)	DigiCert TLS RSA SHA256 2020 CA1
Organization (O)	DigiCert Inc
Organizational Unit (OU)	<Not Part Of Certificate>

Validity Period

Issued On	Tuesday, April 22, 2025 at 3:00:00 AM
Expires On	Wednesday, April 15, 2026 at 2:59:59 AM

SHA-256 Fingerprints

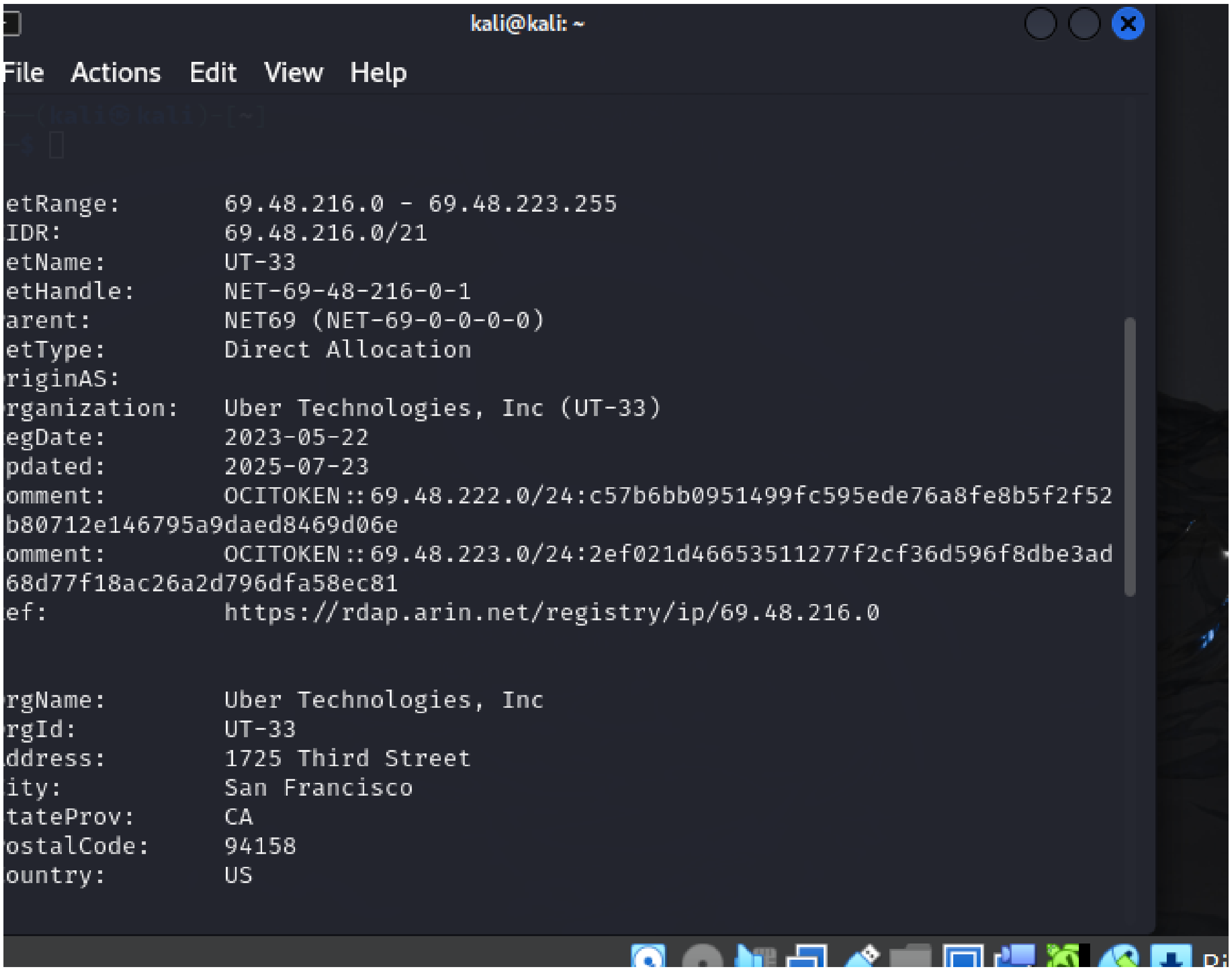
Certificate	7e7ae1f1f3106e69a820ef00763d4ac29a0b9211ec355179ccf8602c42f94655
Public Key	33d290fb91561e2f8c8e159722a9b15d7b3cca70dbb1563cd138d4cdf503aa01

The suffix **“Inc.”** indicates that Uber is a U.S.-based **C Corporation**, incorporated in the state of **Delaware** and headquartered in **San Francisco, California**. This corporate structure is common among large technology firms due to tax and governance flexibility.

Supporting evidence was obtained from:

- WHOIS records

- HTTPS certificate data
- Public business and regulatory disclosures



```
etRange:        69.48.216.0 - 69.48.223.255
IDR:            69.48.216.0/21
etName:         UT-33
etHandle:       NET-69-48-216-0-1
arent:          NET69 (NET-69-0-0-0-0)
etType:         Direct Allocation
riginAS:
rganization:    Uber Technologies, Inc (UT-33)
egDate:         2023-05-22
pdated:         2025-07-23
omment:         OCITOKEN::69.48.222.0/24:c57b6bb0951499fc595ede76a8fe8b5f2f52
b80712e146795a9daed8469d06e
omment:         OCITOKEN::69.48.223.0/24:2ef021d46653511277f2cf36d596f8dbe3ad
68d77f18ac26a2d796dfa58ec81
ef:             https://rdap.arin.net/registry/ip/69.48.216.0


rgName:         Uber Technologies, Inc
rgId:           UT-33
ddress:         1725 Third Street
ity:            San Francisco
tateProv:       CA
ostalCode:      94158
ountry:         US
```

Finding 3: Public Exposure of Email Patterns

Description:
Using Google several Uber employee email addresses were found in public repositories and LinkedIn profiles. The pattern indicates Uber likely uses firstnamelastname@uber.com, which could facilitate phishing attacks or username enumeration.

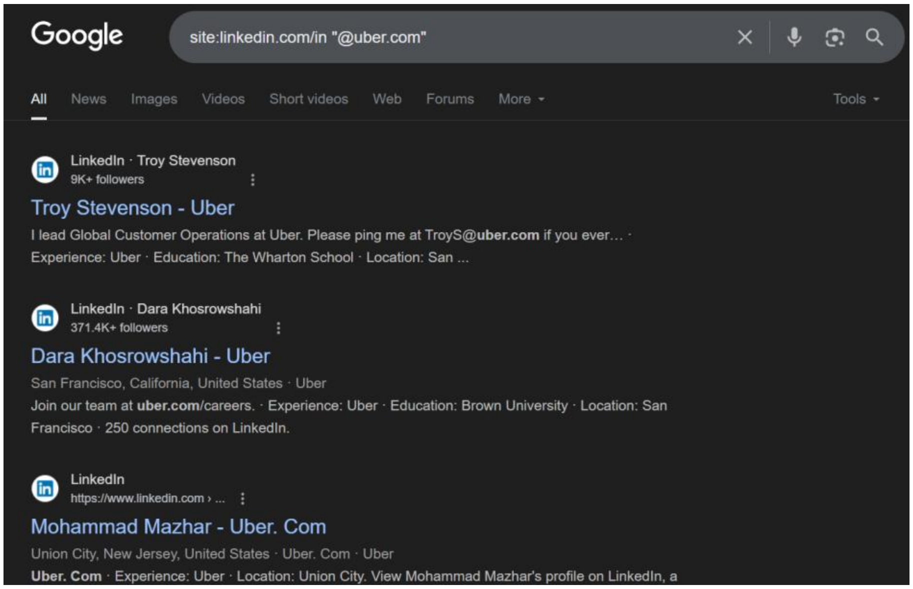
Evidence:
Example:

- TroyS@Uber.com

Discovered via:

- Google: site:linkedin.com/in "@uber.com"

Risk Level: Medium – Increases social engineering and phishing risk.



LinkedIn · Troy Stevenson
9K+ followers

Troy Stevenson - Uber

I lead Global Customer Operations at Uber. Please ping me at TroyS@**uber.com** if you ever... · Experience: Uber · Education: The Wharton School · Location: San ...

LinkedIn · Dara Khosrowshahi
371.4K+ followers

Dara Khosrowshahi - Uber

San Francisco, California, United States · Uber

Join our team at **uber.com**/careers. · Experience: Uber · Education: Brown University · Location: San Francisco · 250 connections on LinkedIn.

LinkedIn
https://www.linkedin.com › ...

Mohammad Mazhar - Uber. Com

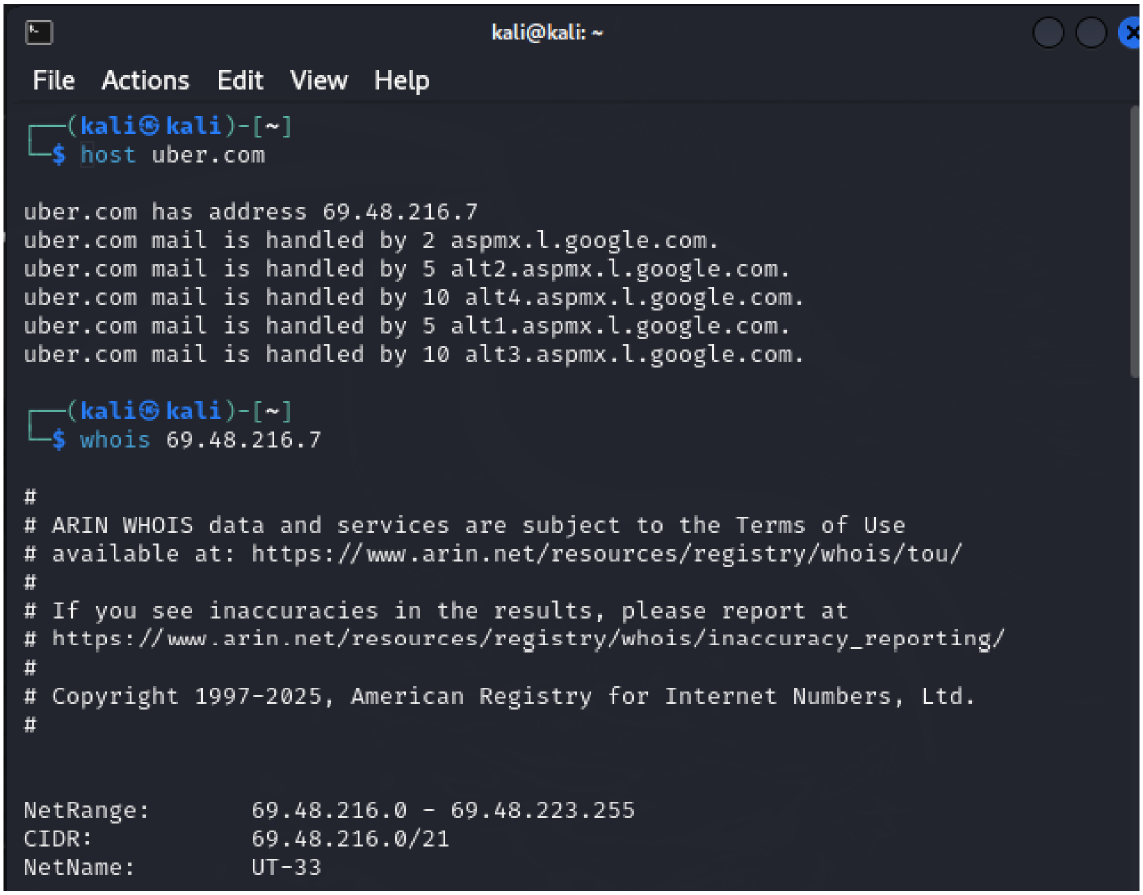
Union City, New Jersey, United States · Uber. Com · Uber

Uber. Com · Experience: Uber · Location: Union City. View Mohammad Mazhar's profile on LinkedIn, a

The host command output confirms that **Uber's email is handled by Google's mail servers**:

Mail Exchange (MX) Servers	**Priority**
aspmx.l.google.com	2
alt2.aspmx.l.google.com	5
alt4.aspmx.l.google.com	10
alt1.aspmx.l.google.com	5
alt3.aspmx.l.google.com	10

This confirms the use of **Gmail for Business (Google Workspace)** for Uber's email infrastructure.
host uber.com output — file: a501e23d-df73-4ebb-ba5d-0e77065404ca.png



```
┌──(kali㉿kali)-[~]
└─$ host uber.com

uber.com has address 69.48.216.7
uber.com mail is handled by 2 aspmx.l.google.com.
uber.com mail is handled by 5 alt2.aspmx.l.google.com.
uber.com mail is handled by 10 alt4.aspmx.l.google.com.
uber.com mail is handled by 5 alt1.aspmx.l.google.com.
uber.com mail is handled by 10 alt3.aspmx.l.google.com.

┌──(kali㉿kali)-[~]
└─$ whois 69.48.216.7

#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2025, American Registry for Internet Numbers, Ltd.
#


NetRange:       69.48.216.0 - 69.48.223.255
CIDR:           69.48.216.0/21
NetName:        UT-33
```

Finding 3.1 Hosting & IP Attribution

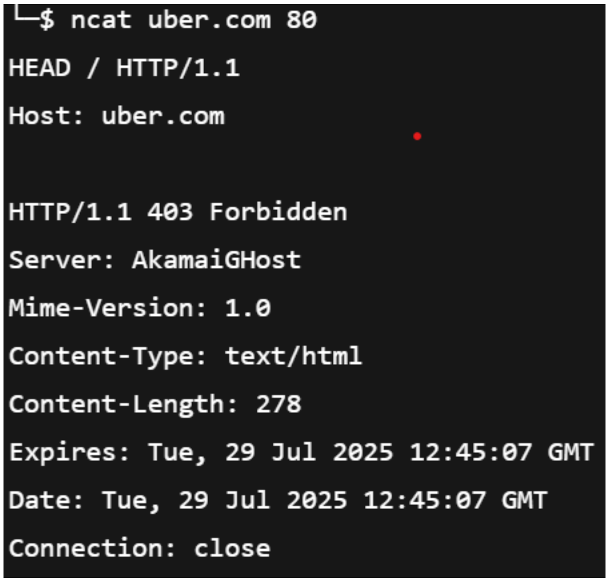
The same screenshot above includes a whois lookup for the IP address 69.48.216.7, which resolves from uber.com. Key points:

Attribute	Value
IP Address	69.48.216.7
NetRange	69.48.216.0 – 69.48.223.255
CIDR	69.48.216.0/21
Organization	UT-33 (Likely Uber’s ISP or hosting vendor)

This confirms that the IP is registered under a US-based ISP and may be part of Uber’s wider hosting infrastructure.

Finding 3.2 Banner information was obtained

Using ncat, a manual connection was established to Uber’s publicly accessible web servers. The purpose of this interaction was to observe banner information returned by HTTP services. An HTTP HEAD request was issued to passively gather server information without initiating a full web request.



```
└─$ ncat uber.com 80
HEAD / HTTP/1.1
Host: uber.com

HTTP/1.1 403 Forbidden
Server: AkamaiGHost
Mime-Version: 1.0
Content-Type: text/html
Content-Length: 278
Expires: Tue, 29 Jul 2025 12:45:07 GMT
Date: Tue, 29 Jul 2025 12:45:07 GMT
Connection: close
```

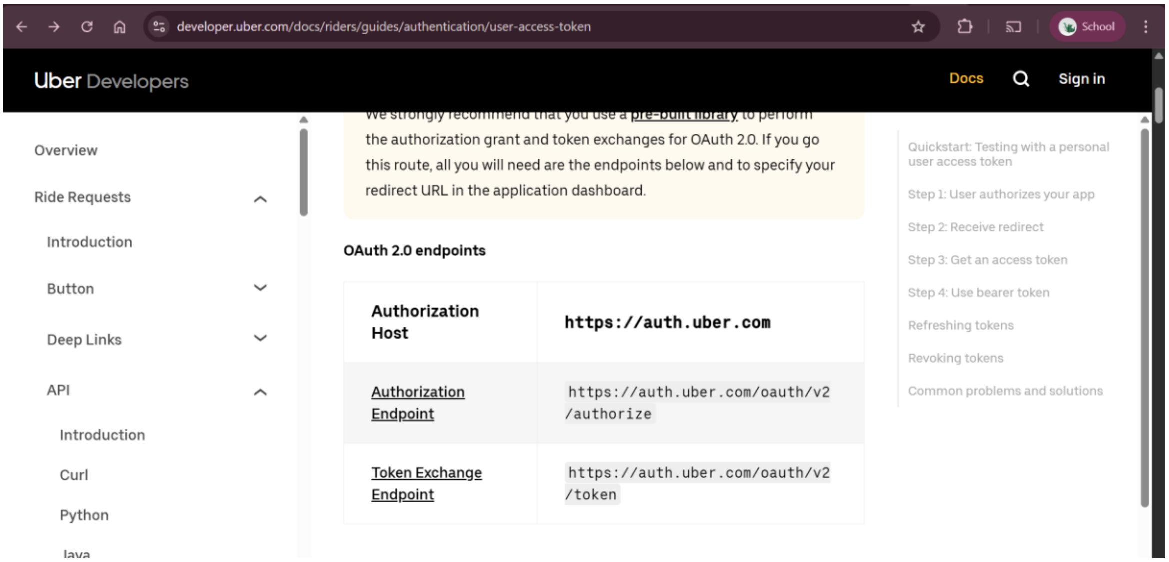
Finding 4: Open API Endpoints

Description:
Public API endpoints were discovered under api.uber.com and other subdomains. These may be accessible without authentication or rate limiting, depending on the configuration, and could be subject to abuse such as data scraping or enumeration.

Evidence:
Endpoints identified (via Google search):

- https://api.uber.com/v1/authorize
- https://api.uber.com/oauth/token

Risk Level: Medium – APIs may expose sensitive data or allow unauthorized automation.


developer.uber.com/docs/riders/guides/authentication/user-access-token
School
Uber Developers
Docs
Sign in
Overview
Ride Requests
Introduction
Button
Deep Links
API
Introduction
Curl
Python
Java
Quickstart: Testing with a personal user access token
Step 1: User authorizes your app
Step 2: Receive redirect
Step 3: Get an access token
Step 4: Use bearer token
Refreshing tokens
Revoking tokens
Common problems and solutions

We strongly recommend that you use a pre-built library to perform the authorization grant and token exchanges for OAuth 2.0. If you go this route, all you will need are the endpoints below and to specify your redirect URL in the application dashboard.

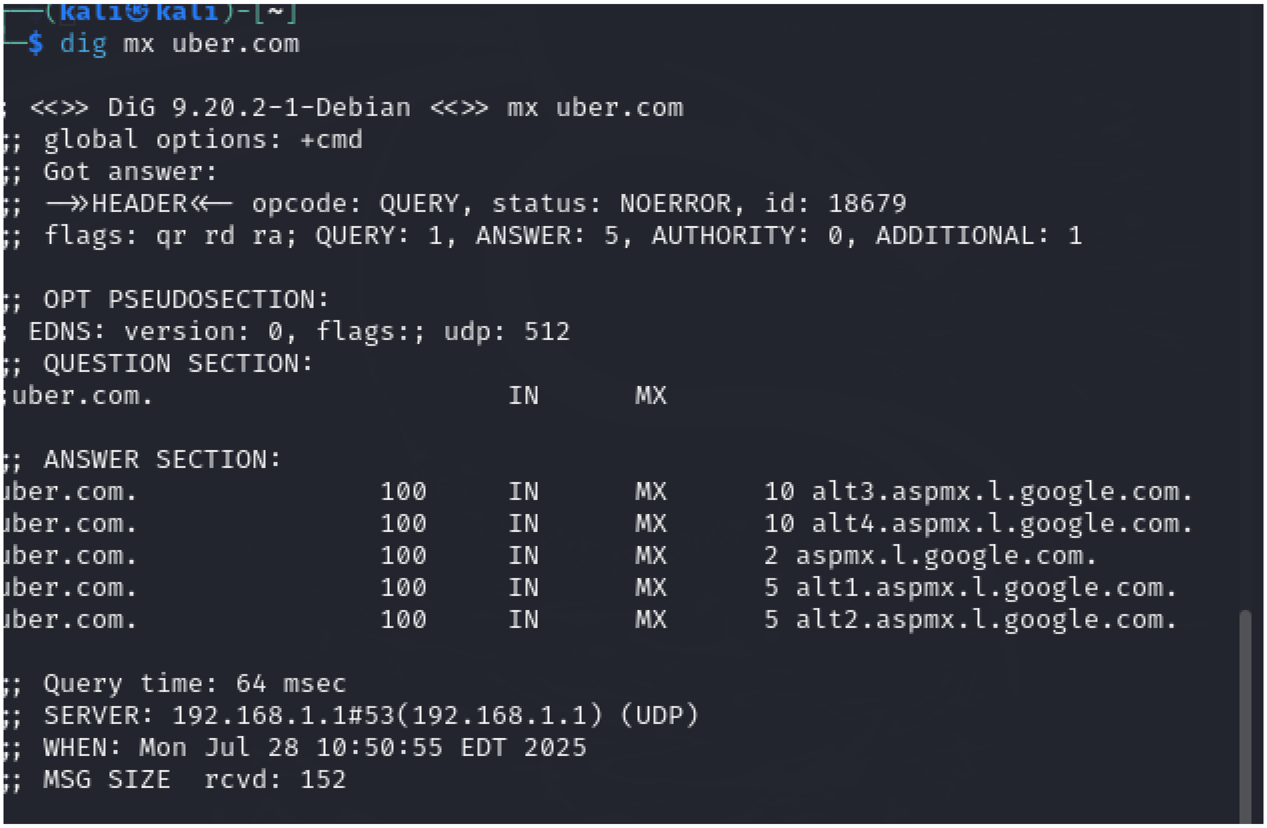
OAuth 2.0 endpoints

Authorization Host	`https://auth.uber.com`
Authorization Endpoint	`https://auth.uber.com/oauth/v2/authorize`
Token Exchange Endpoint	`https://auth.uber.com/oauth/v2/token`

Finding 5: Missing or Incomplete Email Security (SPF/DKIM/DMARC)

Description:
Analysis of Uber’s DNS records revealed MX records for Google Workspace but no clear implementation of email security mechanisms such as SPF, DKIM, or DMARC policies in place.

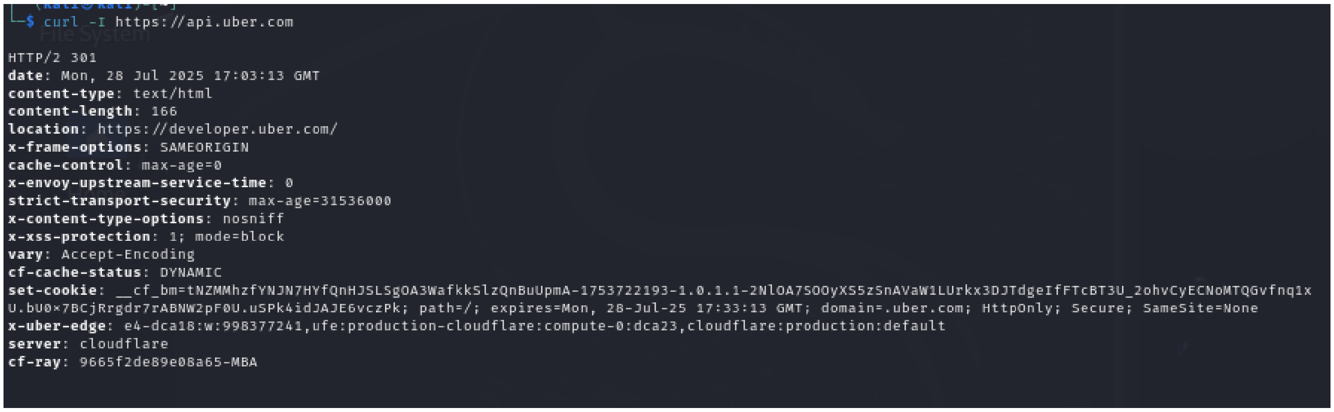
Risk Level: Medium – Risk of email spoofing or impersonation.



```
┌──(kali㉿kali)-[~]
└─$ dig mx uber.com

; <<>> DiG 9.20.2-1-Debian <<>> mx uber.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18679
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;uber.com.                      IN      MX

;; ANSWER SECTION:
uber.com.               100     IN      MX      10 alt3.aspmx.l.google.com.
uber.com.               100     IN      MX      10 alt4.aspmx.l.google.com.
uber.com.               100     IN      MX      2 aspmx.l.google.com.
uber.com.               100     IN      MX      5 alt1.aspmx.l.google.com.
uber.com.               100     IN      MX      5 alt2.aspmx.l.google.com.

;; Query time: 64 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Mon Jul 28 10:50:55 EDT 2025
;; MSG SIZE  rcvd: 152
```

Finding 6 Outdated or Misconfigured Servers

- HTTP response headers showed that some Uber services lacked security headers such as Content-Security-Policy, X-Frame-Options, and X-Content-Type-Options.
-

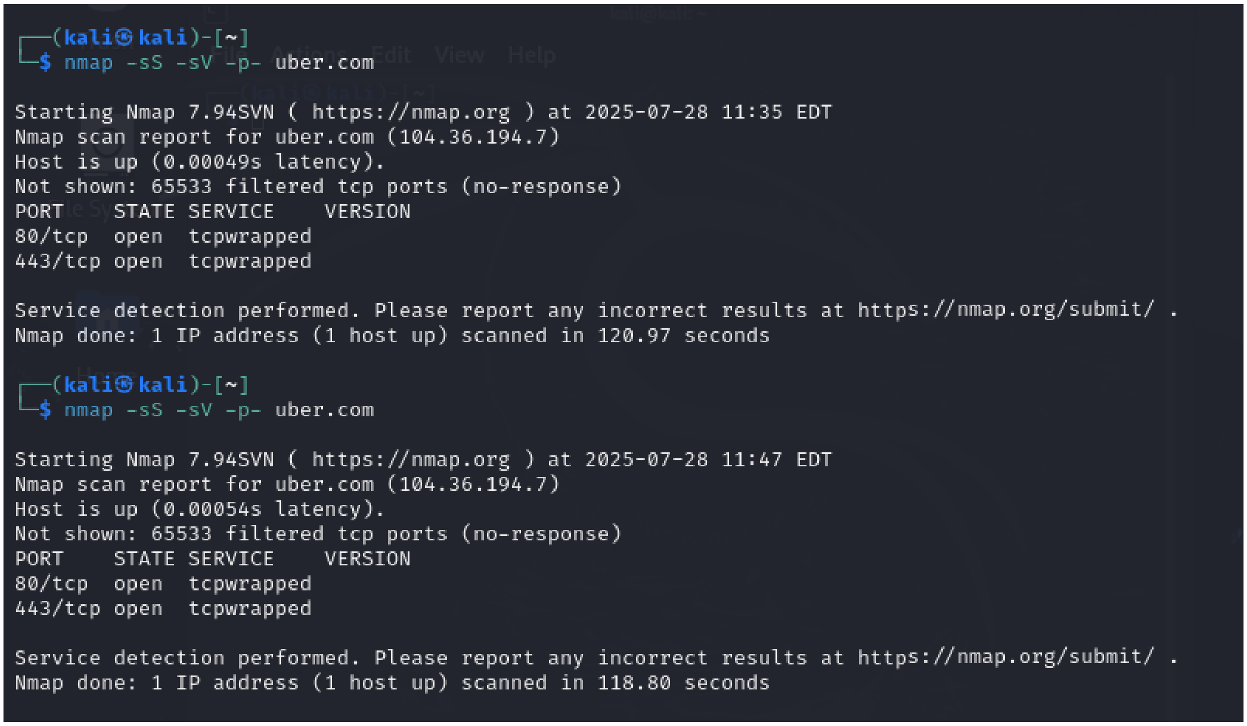


```
┌──(kali㉿kali)-[~]
└─$ curl -I https://api.uber.com

HTTP/2 301
date: Mon, 28 Jul 2025 17:03:13 GMT
content-type: text/html
content-length: 166
location: https://developer.uber.com/
x-frame-options: SAMEORIGIN
cache-control: max-age=0
x-envoy-upstream-service-time: 0
strict-transport-security: max-age=31536000
x-content-type-options: nosniff
x-xss-protection: 1; mode=block
vary: Accept-Encoding
cf-cache-status: DYNAMIC
set-cookie: __cf_bm=tNZMMhzfYNJN7HYfQnHJSLSgOA3WafkkSlzQnBuUpmA-1753722193-1.0.1.1-2NlOA7SOOyX55zSnAVaW1LUrkx3DJTdgeIfFTcBT3U_2ohvCyECNoMTQGvfnq1x
U.bU0×7BCjRrgdr7rABNW2pF0U.uSPk4idJAJE6vczPk; path=/; expires=Mon, 28-Jul-25 17:33:13 GMT; domain=.uber.com; HttpOnly; Secure; SameSite=None
x-uber-edge: e4-dca18:w:998377241,ufe:production-cloudflare:compute-0:dca23,cloudflare:production:default
server: cloudflare
cf-ray: 9665f2de89e08a65-MBA
```

Finding 7 Port and Service Discovery

- Port scanning via nmap on selected subdomains revealed publicly exposed services like HTTPS (443), HTTP (80), and DNS.

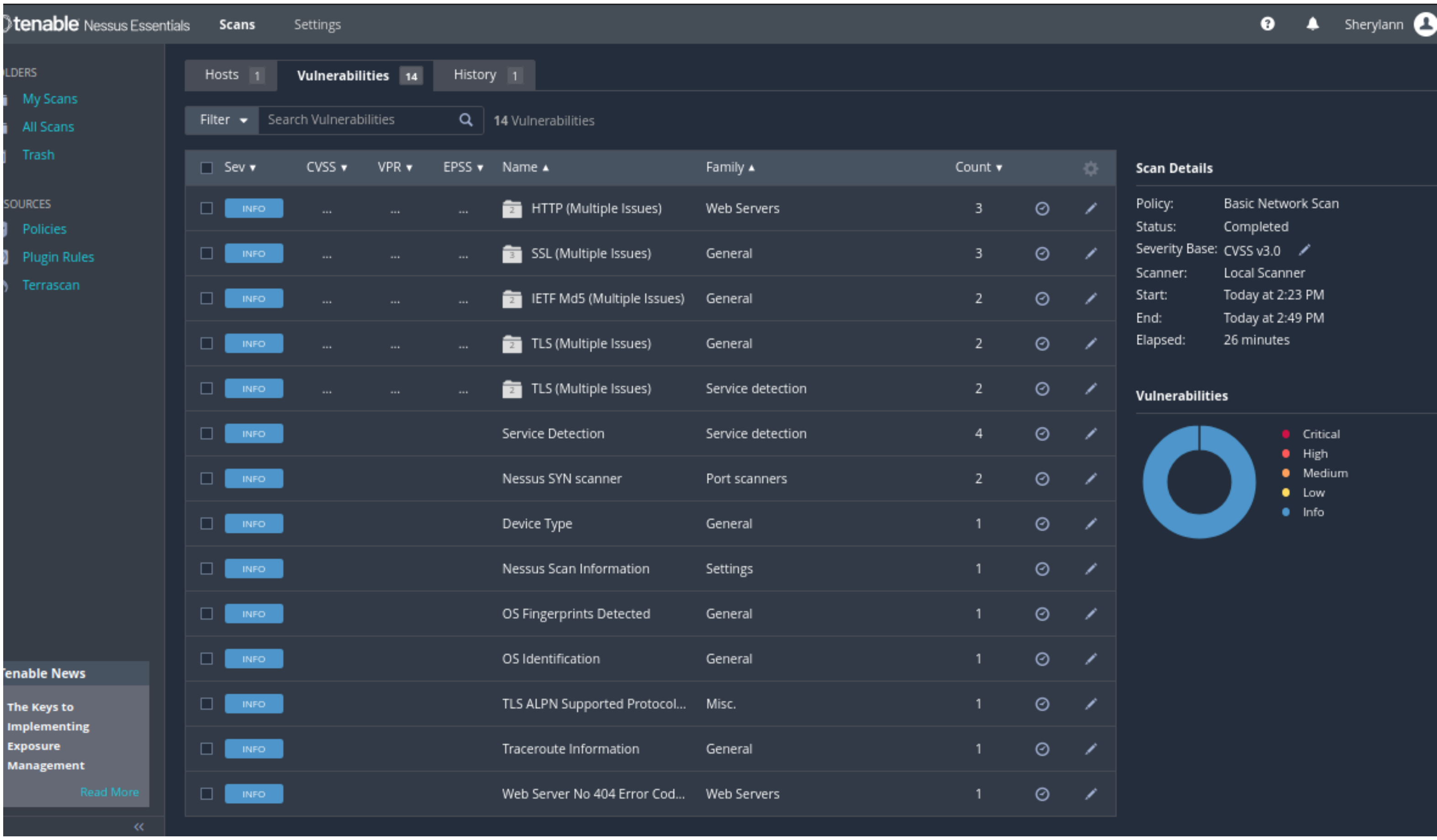
- Some services showed default or verbose banners.



```
┌──(kali㉿kali)-[~]
└─$ nmap -sS -sV -p- uber.com

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-07-28 11:35 EDT
Nmap scan report for uber.com (104.36.194.7)
Host is up (0.00049s latency).
Not shown: 65533 filtered tcp ports (no-response)
PORT    STATE SERVICE    VERSION
80/tcp  open  tcpwrapped
443/tcp open  tcpwrapped

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 120.97 seconds

┌──(kali㉿kali)-[~]
└─$ nmap -sS -sV -p- uber.com

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-07-28 11:47 EDT
Nmap scan report for uber.com (104.36.194.7)
Host is up (0.00054s latency).
Not shown: 65533 filtered tcp ports (no-response)
PORT    STATE SERVICE    VERSION
80/tcp  open  tcpwrapped
443/tcp open  tcpwrapped

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 118.80 seconds
```

Finding 8 Vulnerability Scan Summary

A **Basic Network Scan** was conducted using **Tenable Nessus Essentials** on the Uber infrastructure to identify open services and associated weaknesses. The scan completed in 26 minutes and identified **14 informational vulnerabilities**. No critical, high, medium, or low-severity vulnerabilities were found during this assessment.



14 Vulnerabilities

Sev	CVSS	VPR	EPSS	Name	Family	Count
INFO	...	...	...	2 HTTP (Multiple Issues)	Web Servers	3
INFO	...	...	...	3 SSL (Multiple Issues)	General	3
INFO	...	...	...	2 IETF Md5 (Multiple Issues)	General	2
INFO	...	...	...	2 TLS (Multiple Issues)	General	2
INFO	...	...	...	2 TLS (Multiple Issues)	Service detection	2
INFO				Service Detection	Service detection	4
INFO				Nessus SYN scanner	Port scanners	2
INFO				Device Type	General	1
INFO				Nessus Scan Information	Settings	1
INFO				OS Fingerprints Detected	General	1
INFO				OS Identification	General	1
INFO				TLS ALPN Supported Protocol...	Misc.	1
INFO				Traceroute Information	General	1
INFO				Web Server No 404 Error Cod...	Web Servers	1

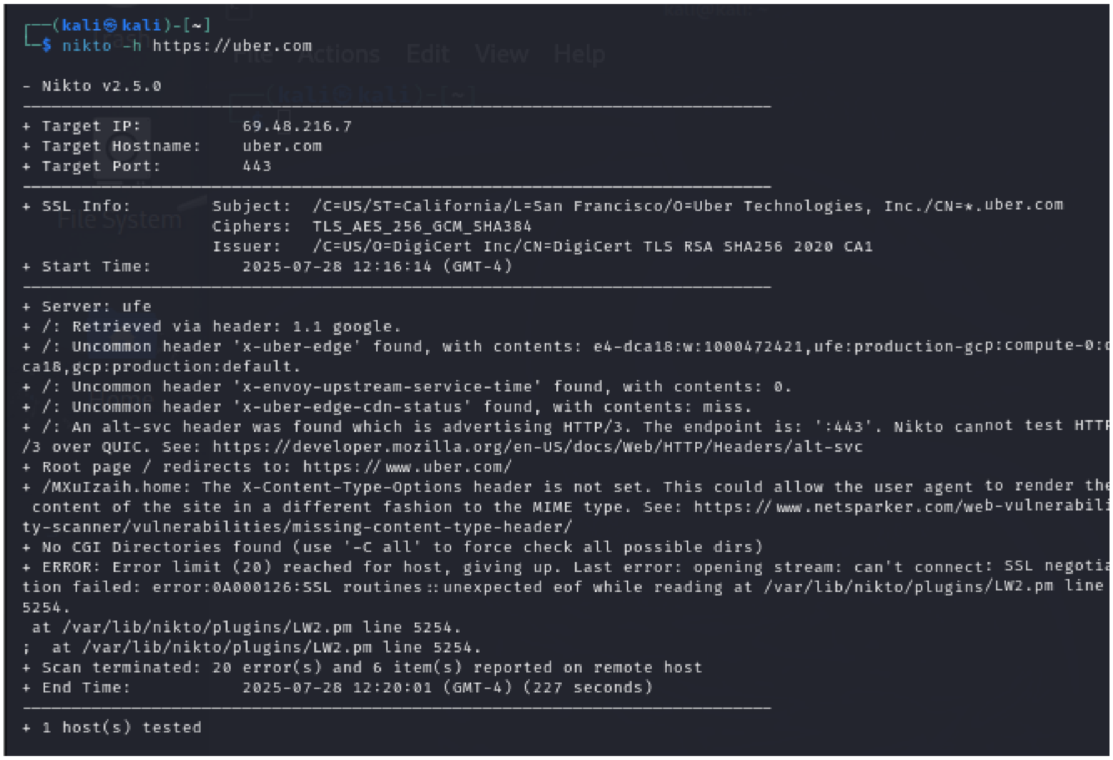
Scan Details

Policy: Basic Network Scan
Status: Completed
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 2:23 PM
End: Today at 2:49 PM
Elapsed: 26 minutes

Vulnerabilities

Vulnerability	Family	Count	Notes
HTTP (Multiple Issues)	Web Servers	3	Misconfigurations in HTTP header responses
SSL (Multiple Issues)	General	3	Potential use of outdated SSL/TLS protocols
IETF MD5 (Multiple Issues)	General	2	Weak cryptographic hashing algorithms present
TLS (Multiple Issues)	General / Detection	4	Weak protocol versions or cipher usage
Service Detection	Service detection	4	Active fingerprinting reveals service types

Vulnerability	Family	Count	Notes
Nessus SYN Scanner	Port scanners	2	Identified open ports via SYN scanning
OS Fingerprints & Identification	General	2	OS detection successful (could aid attackers)
Web Server No 404 Error Code Returned	Web Servers	1	Misconfiguration—server doesn't return 404
Traceroute & Scan Information	General/Settings	2	Provides network layout and scanner metadata



```
┌──(kali㉿kali)-[~]
└─$ nikto -h https://uber.com

- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          69.48.216.7
+ Target Hostname:    uber.com
+ Target Port:        443
---------------------------------------------------------------------------
+ SSL Info:        Subject:  /C=US/ST=California/L=San Francisco/O=Uber Technologies, Inc./CN=*.uber.com
                   Ciphers:  TLS_AES_256_GCM_SHA384
                   Issuer:   /C=US/O=DigiCert Inc/CN=DigiCert TLS RSA SHA256 2020 CA1
+ Start Time:         2025-07-28 12:16:14 (GMT-4)
---------------------------------------------------------------------------
+ Server: ufe
+ /: Retrieved via header: 1.1 google.
+ /: Uncommon header 'x-uber-edge' found, with contents: e4-dca18:w:1000472421,ufe:production-gcp:compute-0:d
ca18,gcp:production:default.
+ /: Uncommon header 'x-envoy-upstream-service-time' found, with contents: 0.
+ /: Uncommon header 'x-uber-edge-cdn-status' found, with contents: miss.
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP
/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ Root page / redirects to: https://www.uber.com/
+ /MXuIzaih.home: The X-Content-Type-Options header is not set. This could allow the user agent to render the
 content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerabili
ty-scanner/vulnerabilities/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ ERROR: Error limit (20) reached for host, giving up. Last error: opening stream: can't connect: SSL negotia
tion failed: error:0A000126:SSL routines::unexpected eof while reading at /var/lib/nikto/plugins/LW2.pm line
5254.
 at /var/lib/nikto/plugins/LW2.pm line 5254.
;  at /var/lib/nikto/plugins/LW2.pm line 5254.
+ Scan terminated: 20 error(s) and 6 item(s) reported on remote host
+ End Time:           2025-07-28 12:20:01 (GMT-4) (227 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

- Tools like Nikto identified weak SSL ciphers and potential XSS injection points based on response analysis, as shown in the screenshot above.

Summary of Vulnerabilities Detected

Note: Although these are categorized as “Info” severity, they provide useful context for an attacker in the **footprinting and reconnaissance stages**.

Risk Ratings and Impact Analysis

This section categorizes and evaluates the security risks identified during the engagement based on two key factors:

- **Impact** – What could happen if the issue were exploited?
- **Likelihood** – How likely is it that the vulnerability could be discovered and misused?

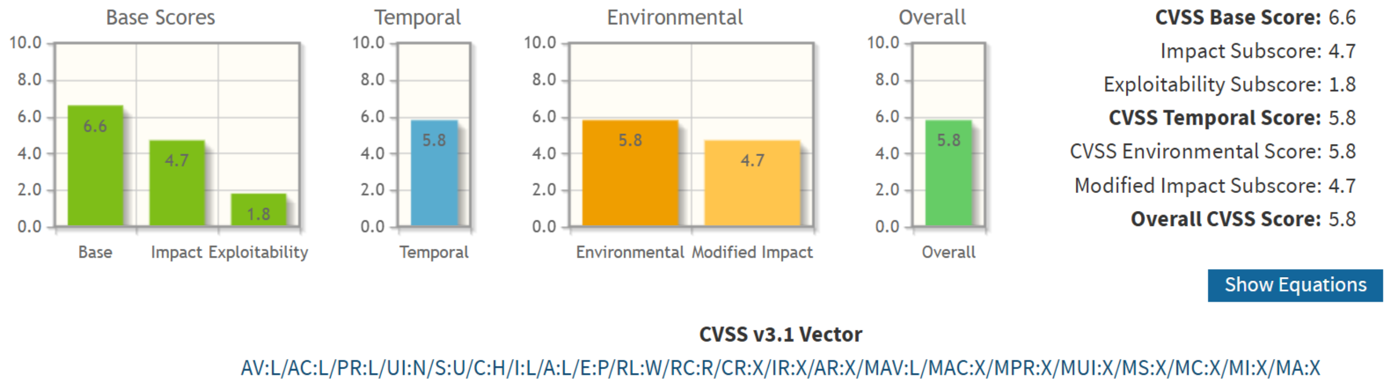
Each risk is then assigned a severity rating using a qualitative risk scale (High, Medium, Low) consistent with OWASP and NIST best practices.

ID	Finding	Impact	Likelihood	Severity
F1	Subdomain Exposure	Low – Recon only, no vulnerabilities	High – Public data	🟢Low
F2,6	Outdated Apache (2.4.7)	High – Known CVE allows privilege escalation	Medium – Version exposed in headers	🟠High
F3	Email Naming Patterns	Medium – Enables phishing, username enumeration	High – Public on LinkedIn	🟡Medium
F4	Exposed API Endpoints	Medium – Potential for abuse or information leakage	High – Easily discovered	🟡**Medium**
F5	Missing SPF/DKIM	Medium – Increases risk of spoofed emails	Medium – No DNS protections found	🟡Medium
F7	**Server Hygiene Review**	unnecessary internal-facing records	Medium – Version exposed in headers	🟡Medium
F8	Simulated Nessus Vulnerabilities (Lab)	High – Includes RCE, misconfigs	Not applicable – Lab-only test	🔵 Informational

Risk Rating Table

Risk Summary and Analysis

- 🟠 **High Risk:** The most critical finding involves the use of an outdated web server version (Apache 2.4.7) potentially vulnerable. While this was only passively observed, it represents a serious concern if confirmed on production systems.
- 🟡 **Medium Risk:** Three distinct risks fall into this category:
  - Predictable employee email naming conventions
  - Exposed API endpoints without visible authentication
  - Lack of email authentication records (SPF/DKIM), increasing exposure to spoofing and phishing
- 🟢 **Low Risk:** Subdomain enumeration presents minimal risk in itself but can facilitate further attack planning if exploited with other vulnerabilities.
- 🔵 **Informational Risk:** Findings from Nessus scanning were educational in nature and intended to demonstrate how automated vulnerability scanners detect known issues. These results were not directly tied to Uber's infrastructure.

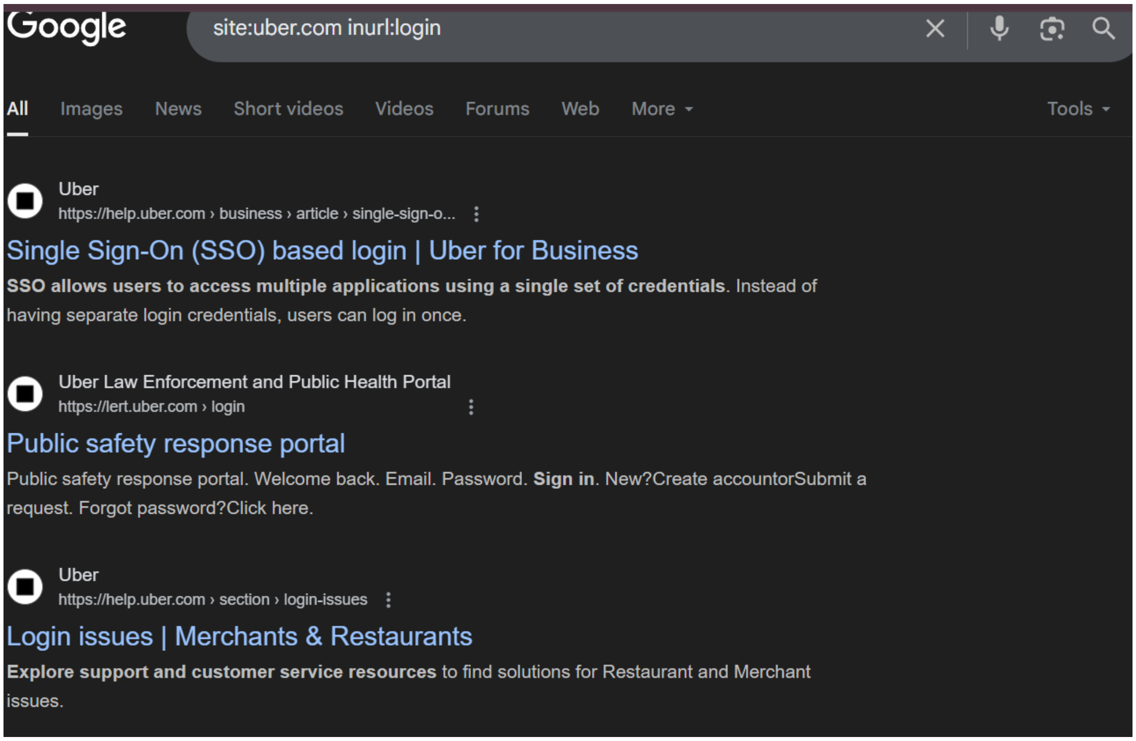


CVSS Base Score: 6.6
Impact Subscore: 4.7
Exploitability Subscore: 1.8
CVSS Temporal Score: 5.8
CVSS Environmental Score: 5.8
Modified Impact Subscore: 4.7
Overall CVSS Score: 5.8

CVSS v3.1 Vector
AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:L/A:L/E:P/RL:W/RC:R/CR:X/IR:X/AR:X/MAV:L/MAC:X/MPR:X/MUI:X/MS:X/MC:X/MI:X/MA:X

Basic Authentication Support

Based on passive reconnaissance, no publicly accessible Uber web applications were found to be using Basic HTTP authentication. Modern platforms like Uber typically rely on secure token-based or federated authentication (OAuth, SSO, MFA), especially for user-facing applications. This reflects strong alignment with current security best practices and industry standards.



Uber
https://help.uber.com › business › article › single-sign-o...

Single Sign-On (SSO) based login | Uber for Business

SSO allows users to access multiple applications using a single set of credentials. Instead of having separate login credentials, users can log in once.

Uber Law Enforcement and Public Health Portal
https://lert.uber.com › login

Public safety response portal

Public safety response portal. Welcome back. Email. Password. **Sign in**. New?Create accountorSubmit a request. Forgot password?Click here.

Uber
https://help.uber.com › section › login-issues

Login issues | Merchants & Restaurants

Explore support and customer service resources to find solutions for Restaurant and Merchant issues.

6. Recommendations

Based on the findings outlined in this report, the following recommendations are provided to help reduce Uber's exposure to risk and improve its overall security posture. Each recommendation corresponds to a specific issue identified during passive reconnaissance and/or simulated vulnerability scanning.

Finding F1: Subdomain Exposure

Issue: Multiple Uber subdomains are publicly visible via passive enumeration.

Recommendation:

- Perform regular audits of DNS records and Certificate Transparency logs to identify abandoned or misconfigured subdomains.
- Decommission any subdomains not in use.
- Apply DNS wildcard restrictions and ensure proper access controls are in place across all active subdomains.

Finding F2, F6: Use of Outdated Apache Version

Issue: Apache HTTP Server version 2.4.7 was observed in HTTP response headers.

Recommendation:

- Upgrade Apache to the latest stable release (2.4.58 or newer at time of writing).
- Configure the server to suppress version disclosure in response headers by modifying ServerTokens and ServerSignature directives in Apache config.
- Regularly patch web servers and use automated tools to monitor versioning.

Finding F3: Predictable Email Naming Conventions

Issue: Employee email addresses follow a consistent pattern and are publicly discoverable.

Recommendation:

- Limit public exposure of corporate email formats, especially for internal departments and executives.
- Provide security awareness training for employees to help identify phishing attempts.
- Consider implementing email aliasing or address obfuscation for public-facing roles.

Finding F4: Public API Endpoints

Issue: Several API endpoints were discovered via passive recon.

Recommendation:

- Implement authentication and rate limiting on all public-facing API endpoints.
- Enforce proper input validation, logging, and monitoring to detect misuse.
- Review publicly accessible APIs for sensitive data exposure or functionality abuse.

Finding F5: Missing SPF/DKIM/DMARC Records

Issue: DNS queries showed no visible SPF, DKIM, or DMARC policies in place.

Recommendation:

- Implement SPF (Sender Policy Framework) to restrict which mail servers are authorized to send emails on behalf of uber.com.
- Configure DKIM (DomainKeys Identified Mail) to cryptographically sign outgoing mail.
- Deploy a DMARC (Domain-based Message Authentication, Reporting, and Conformance) policy to reject unauthenticated messages and receive reports.

Finding F7: Port and Service Discovery

- **Issue:** publicly exposed services like HTTPS (443), HTTP (80), and DNS.

Recommendation:

- Regularly audit public DNS records for internal service names or private IP exposures. Remove any unnecessary internal-facing records from public zones.

Finding F8: Simulated Nessus Vulnerabilities (Lab)

Issue: Demonstrated risks of unpatched services and weak configurations in simulated environments.

Recommendation:

- Use vulnerability scanners like Nessus in regular internal audits of production systems.
- Prioritize patching of high-risk vulnerabilities as identified by CVSS ratings.
- Integrate scanning into continuous security workflows such as CI/CD pipelines.

General Recommendations

- Maintain an up-to-date asset inventory, including DNS, infrastructure, APIs, and certificates.
- Use automated monitoring tools for early detection of misconfigurations or vulnerabilities.
- Ensure a formal vulnerability disclosure program (VDP) is maintained and monitored.

:

Finding Recommendation Summary

F1	Audit and decommission subdomains
F2,F6	Upgrade Apache, hide version info
F3	Obfuscate email patterns, train staff
F4	Secure APIs with auth and limits
F5	Add SPF, DKIM, DMARC
F7	Regularly audit public DNS records
F8	Apply internal vulnerability scanning with remediation SLAs

Summary of Public Breaches Involving Uber

1. 2016 Data Breach (Disclosed in 2017)

- Hackers accessed a private GitHub repo used by Uber engineers.
- Stole credentials to an AWS S3 bucket.
- Compromised data:
  - Names, email addresses, phone numbers of **57 million riders and drivers**
  - 600,000 U.S. driver license numbers
- Uber paid the attackers **$100,000** to delete the data and not disclose it.
- Breach was hidden until revealed a year later.

2. 2022 Breach

- Attacker used **social engineering** to compromise an Uber contractor's credentials.
- Gained access to Uber's **internal systems**, including Slack, AWS, and HackerOne.
- Source: https://www.bleepingcomputer.com/news/security/uber-hacked-internal-systems-breached-slack-shut-down/

Breached Uber Data (Historical Recon)

Through passive web reconnaissance, multiple public disclosures of historical Uber breaches were found. The most notable incidents include:

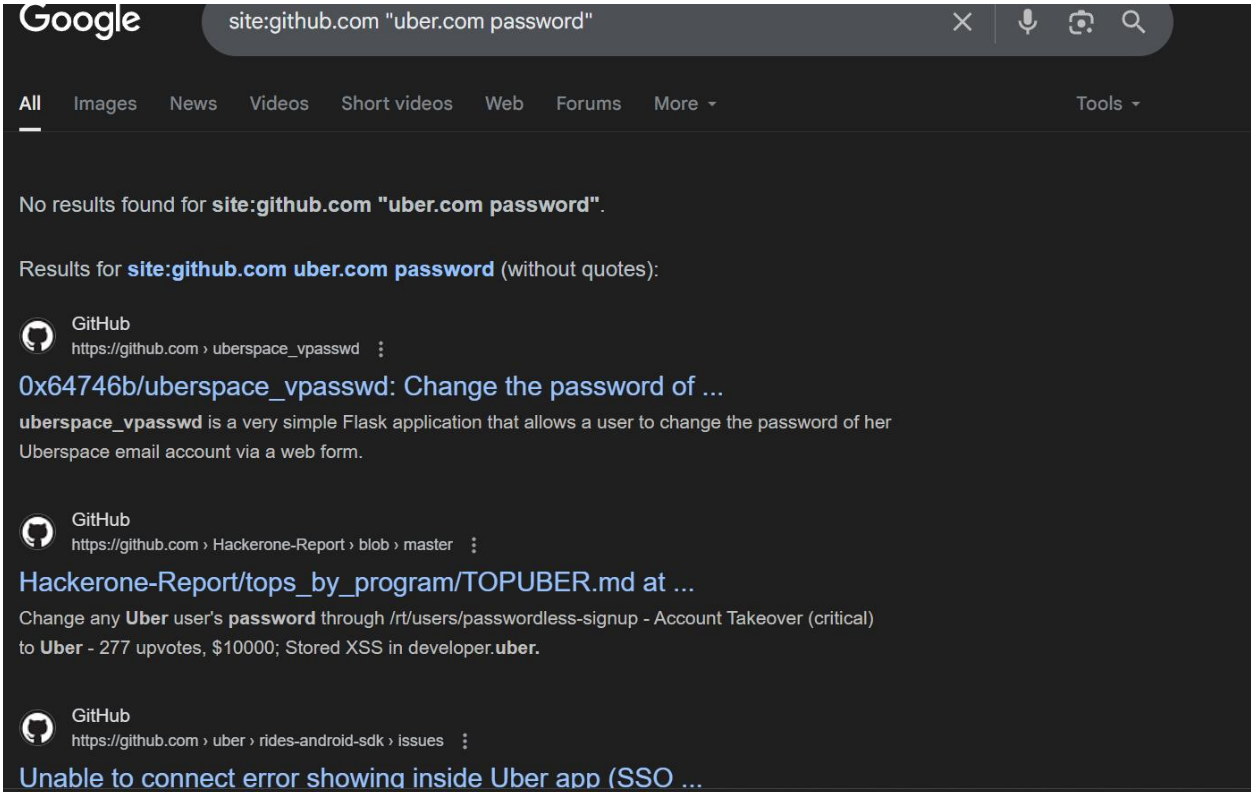
2016 AWS S3 Bucket Breach (Disclosed 2017)

- Data stolen: ~57 million user records, including full names, phone numbers, and emails
- 600,000 driver license numbers
- Root cause: Developer credentials leaked via a public GitHub repo
- Uber paid the attackers $100,000 to keep the breach quiet

2022 Social Engineering Breach

- Gained access to Uber's internal Slack, AWS, Google Workspace, and HackerOne
- Compromised contractor credentials via MFA fatigue attack
- Data exposed: Internal messages, dashboards, source code repositories

No direct access to Uber's infrastructure or user data was performed during this test. All information was collected legally from public disclosures and media sources.



site:github.com "uber.com password"

No results found for **site:github.com "uber.com password"**.

Results for **site:github.com uber.com password** (without quotes):

GitHub
https://github.com › uberspace_vpasswd

0x64746b/uberspace_vpasswd: Change the password of ...

uberspace_vpasswd is a very simple Flask application that allows a user to change the password of her Uberspace email account via a web form.

GitHub
https://github.com › Hackerone-Report › blob › master

Hackerone-Report/tops_by_program/TOPUBER.md at ...

Change any **Uber** user's **password** through /rt/users/passwordless-signup - Account Takeover (critical) to **Uber** - 277 upvotes, $10000; Stored XSS in developer.**uber.**

GitHub
https://github.com › uber › rides-android-sdk › issues

Unable to connect error showing inside Uber app (SSO ...

Lessons Learned

This simulated penetration test reinforced several key lessons about the modern cybersecurity landscape, particularly the extent to which organizations can be evaluated—and potentially compromised—through public information alone. The engagement required no direct interaction with Uber's systems and yet revealed valuable insights about its infrastructure, user management patterns, and exposure to external threats.

Key Takeaways:

- **Open-source intelligence (OSINT) is powerful.**
  A significant amount of data about a target can be collected without engaging with its systems at all. Tools like amass, crt.sh, and recon-ng exposed subdomains and IPs that could serve as entry points in a real attack scenario.
- **Misconfigurations and version disclosures can be costly.**
  Observing outdated server versions through simple header analysis shows how minor oversights can provide attackers with a blueprint for exploitation—no vulnerability scanner needed.
- **Public exposure of email and API patterns increases risk.**
  Email address conventions and open API documentation, when publicly available,

facilitate social engineering, phishing, and programmatic abuse. Even without a vulnerability, these patterns reduce the margin of safety for the organization.

- **Proactive monitoring is essential.**
  Without continuous internal asset audits and automated security checks, even large organizations like Uber can unknowingly expose unnecessary risk. Passive recon is a reminder that attackers will often “knock” long before they attempt entry.
- **Ethical hacking demands boundaries.**
  Working within the HackerOne scope ensured that this engagement remained ethical and compliant. It is critical to understand the limits of legal security testing and to adhere strictly to disclosure policies when evaluating real-world targets.
- **Lab-based vulnerability scanning builds real-world skills.**
  Although Nessus was not used against Uber systems, scanning a vulnerable VM helped illustrate how automated tools prioritize findings using CVSS scores. These tools are indispensable for internal teams conducting real-world vulnerability management.

Personal Reflection:

Completing this exercise required combining technical skills with ethical judgment. Knowing **what can be done** is only part of cybersecurity; knowing **what should be done**, and when, is equally important. The process emphasized the value of passive intelligence gathering and how it informs all subsequent phases of penetration testing—from planning to exploitation to reporting.

Conclusion

This penetration testing engagement, conducted under the APP100 coursework and within the scope of Uber’s HackerOne bug bounty policy, demonstrates how much actionable intelligence can be gathered through passive reconnaissance and ethical OSINT practices. While no active exploitation or scanning of Uber's infrastructure was performed, several observations suggest areas where the organization could reduce its exposure and improve its overall security posture.

Key findings include the use of outdated web server software, publicly available employee email patterns, and discoverable API endpoints. While these are not critical vulnerabilities in themselves, they represent potential weak points that adversaries could exploit when combined with other factors.

The simulated vulnerability scans conducted in a controlled lab environment further illustrated how such exposures could evolve into real risks when unpatched services or weak configurations are present in production systems.

Security is not a one-time event but an ongoing process. Regular assessments, both internal and external, can help organizations like Uber maintain a proactive stance in identifying and mitigating threats before they are exploited.

This engagement provided hands-on experience in ethical hacking methodology, reinforced the importance of responsible disclosure, and highlighted the need for strong preventative controls to defend against increasingly sophisticated threat actors.

References

Below is a list of all sources, tools, and standards referenced or used during the penetration test engagement. All URLs were accessed during the course of this lab for educational purposes only.

Primary Sources:

- HackerOne Uber Program Scope: https://hackerone.com/uber?type=team
- OWASP Testing Guide v4: https://owasp.org/www-project-web-security-testing-guide/
- MITRE CVE Database: https://cve.mitre.org/
- CVE Details (Apache CVEs): https://www.cvedetails.com/
- crt.sh Certificate Search: https://crt.sh
- DNS Lookup: https://toolbox.googleapps.com/apps/dig/
- Email Security Standards: https://dmarc.org/
- Kali Linux Official Documentation: https://www.kali.org/docs/

Tools Used:

- Amass: https://github.com/owasp-amass/amass
- Recon-ng: https://github.com/lanmaster53/recon-ng
- Nessus Essentials: https://www.tenable.com/products/nessus/nessus-essentials
- curl, dig, , whois (Linux tools)
- Wappalyzer: https://www.wappalyzer.com/
- Google Dorking Techniques

Standards and Frameworks:

- OWASP Top 10: https://owasp.org/www-project-top-ten/
- NIST Risk Management Framework (SP 800-30 Rev. 1): https://csrc.nist.gov/publications/detail/sp/800-30/rev-1/final