NAME : SHREYAS PATEL
BRANCH : TE COMPS
UID : 2018130037

LABORATORY

CEL62: Cryptography and System Security
Winter 2021

Experiment 6:	**Creating and Reading QR Codes** **Ref:** **https://www.linux-magazine.com/Online/Features/Generating-QR-Codes-in-Linux**

Note: Students are advised to read through this lab sheet before doing experiment. On-the-spot evaluation may be carried out during or at the end of the experiment. Your performance, teamwork/Personal effort, and learning attitude will count towards the marks.

Experiment 6: Creating and Reading QR codes

1 OBJECTIVE

Creating and Reading QR Codes for given exercise

2 INTRODUCTION

With the right tools, you can create your own QR code squares with information you want to share on a business card, in a letter, or on your website.

Read errors of unreliable, one-dimensional bar codes often caused interruptions in industrial production, prompting companies like Toyota and its subsidiary Denso Wave to develop as early as 1994 a new code for acquiring stock data. The new matrix bar code was designed to store more information than the traditional bar code and to stay legible, even if the label was dirty, wrinkled, or partially destroyed.

The quick response code, or QR code, comprises a matrix of square dots instead of the usual lines. Measuring up to 177 by 177 dots, the QR code encodes up to 4,296 characters, compared with a bar code that encodes just 13.

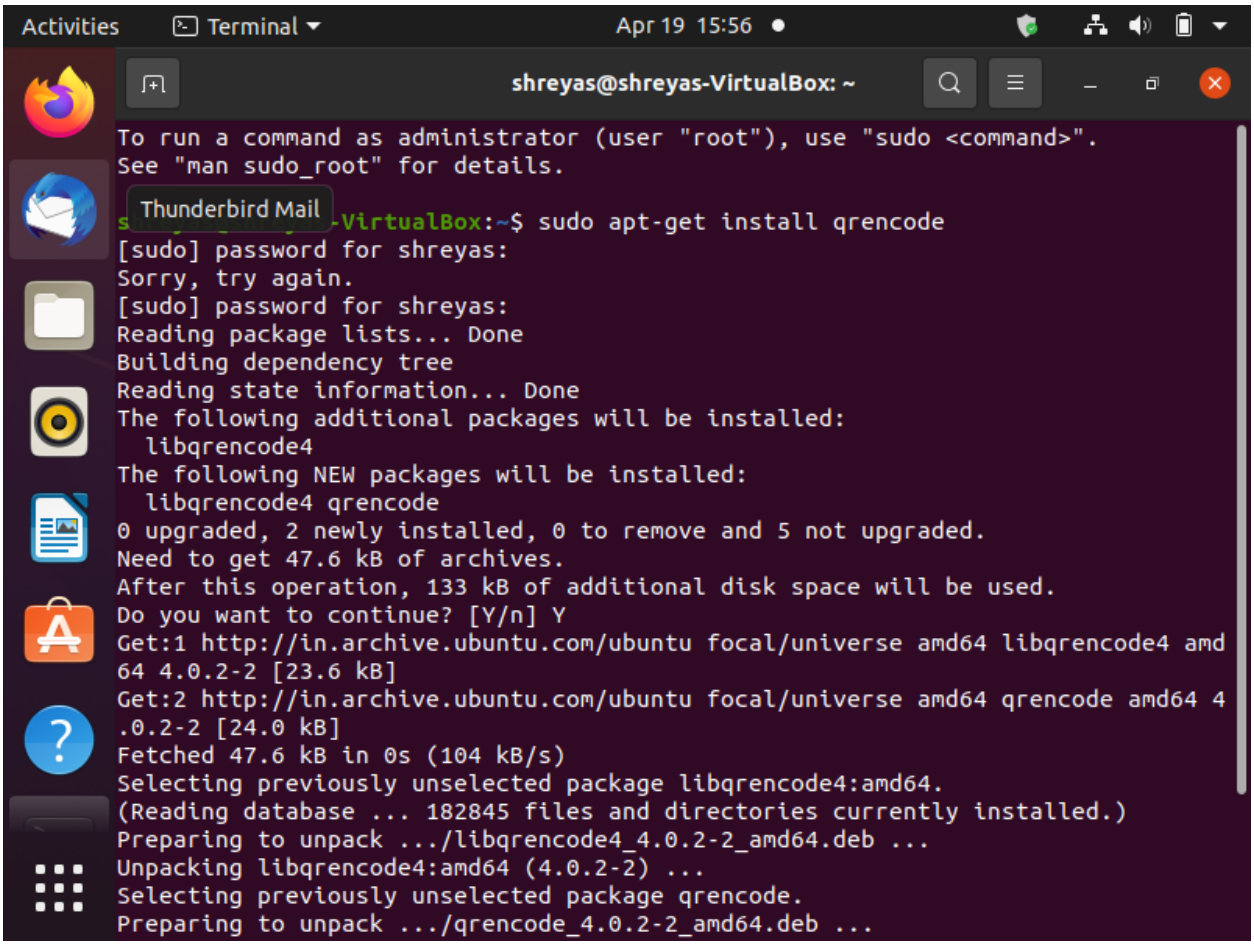
Thanks to numerous free reader apps for smartphones, QR codes have gained in popularity in recent years. Posters, catalogs, magazines, business cards, and even television screens display the small squares, offering additional information or URLs for microsites.

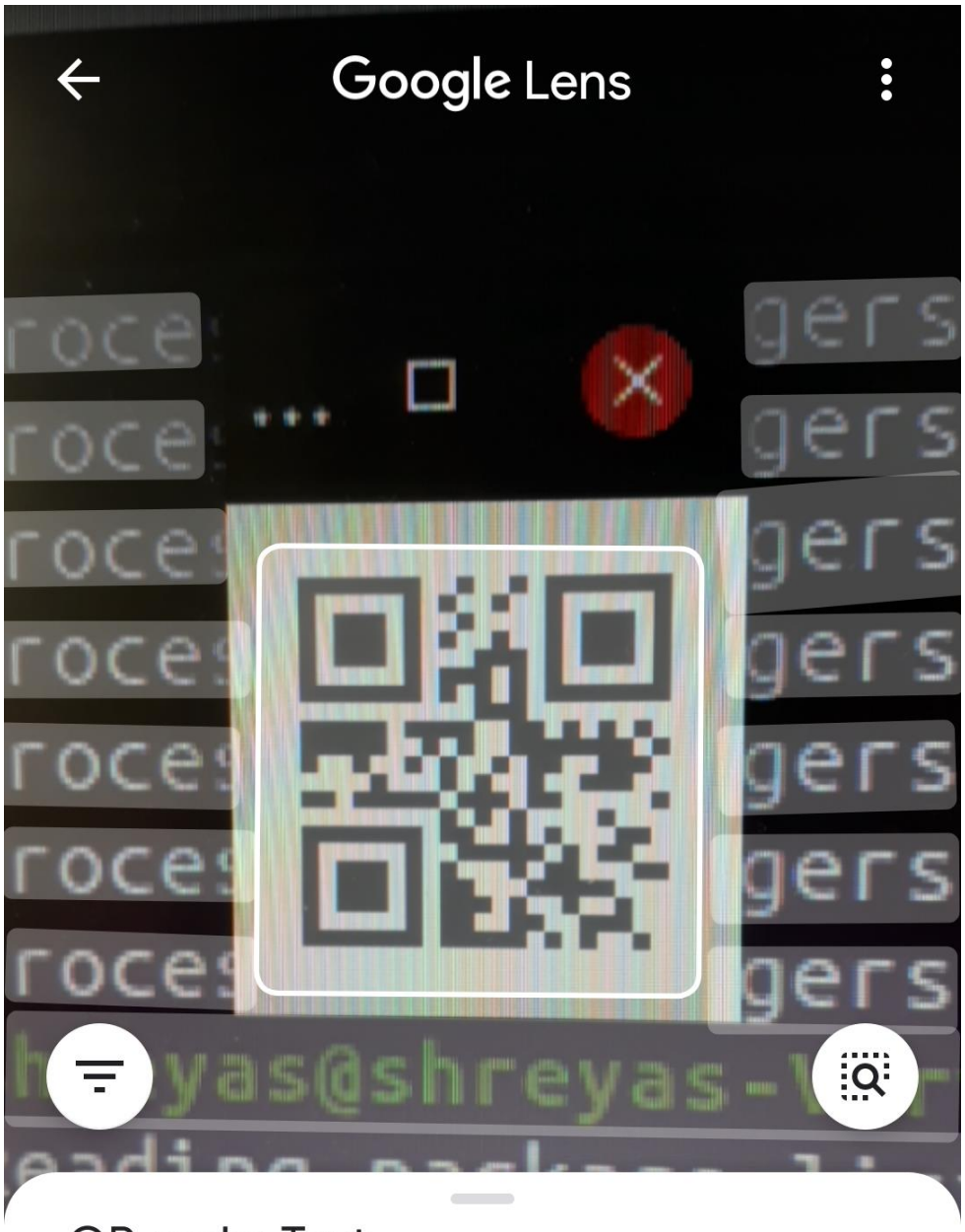
Data Grabber

If you want to encode your address into a QR code for a business card, you could try one of the many services on the Internet. However, restrictions typically apply. For example, some services allow only QR codes in a certain size or in limited quantities. Also, you never know for sure, what the services do with your data. For example, some sites explicitly allow sharing and selling the data to third parties in their terms of use, more or less pre-programming an increase in junk mail in the future. Fortunately, several QR code tools can help you avoid these problems in Linux.

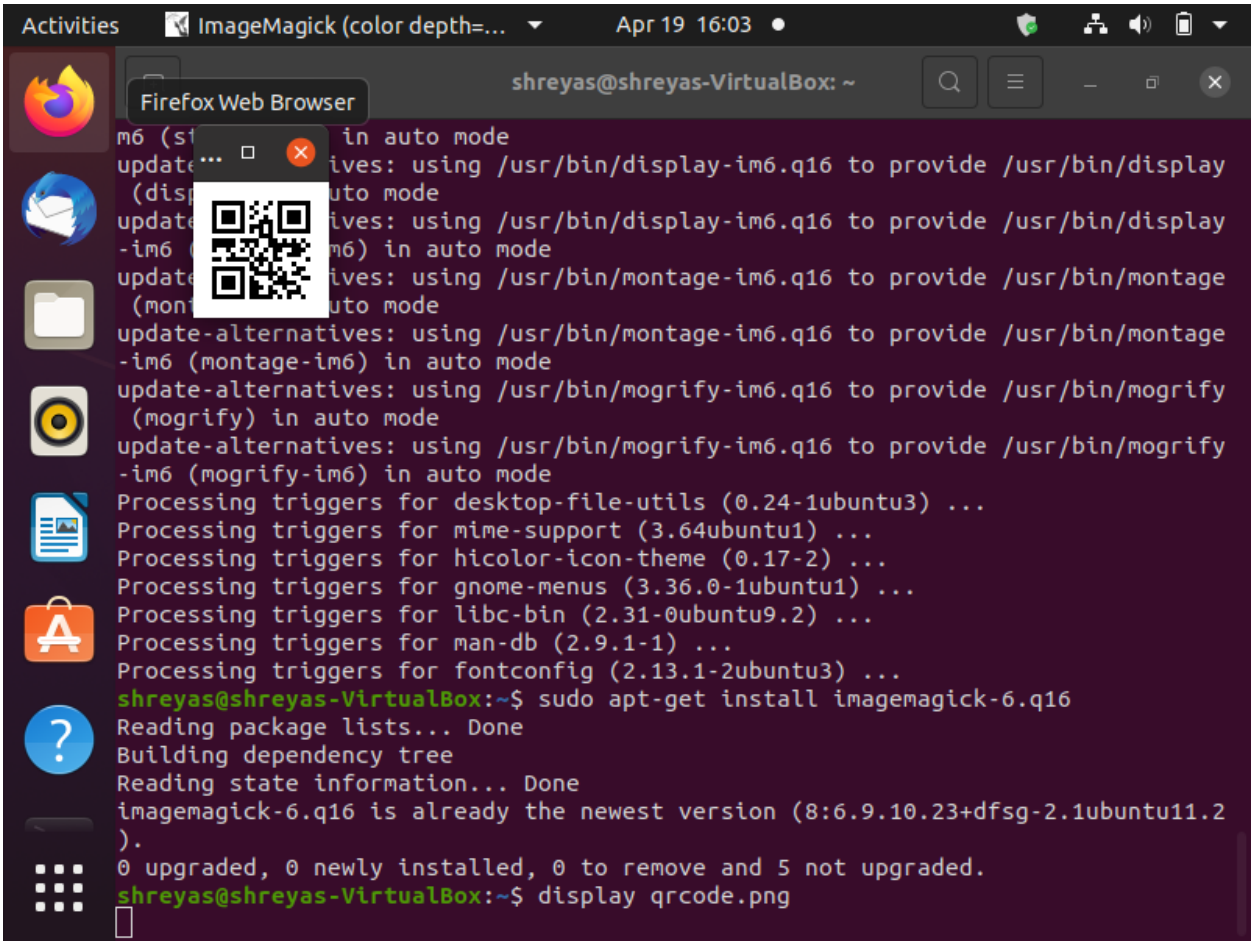
3 PROCEDURE/TASK

The quickest way to create a QR code is with the Qrencode command-line utility. Any major distribution can install Qrencode via the package manager. The following command then creates a QR code containing the text “Hello World!”:


Activities
Terminal
Apr 19 15:56
shreyas@shreyas-VirtualBox: ~
Thunderbird Mail
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
-VirtualBox:~$ sudo apt-get install qrencode
[sudo] password for shreyas:
Sorry, try again.
[sudo] password for shreyas:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
libqrencode4
The following NEW packages will be installed:
libqrencode4 qrencode
0 upgraded, 2 newly installed, 0 to remove and 5 not upgraded.
Need to get 47.6 kB of archives.
After this operation, 133 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 libqrencode4 amd
64 4.0.2-2 [23.6 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 qrencode amd64 4
.0.2-2 [24.0 kB]
Fetched 47.6 kB in 0s (104 kB/s)
Selecting previously unselected package libqrencode4:amd64.
(Reading database ... 182845 files and directories currently installed.)
Preparing to unpack .../libqrencode4_4.0.2-2_amd64.deb ...
Unpacking libqrencode4:amd64 (4.0.2-2) ...
Selecting previously unselected package qrencode.
Preparing to unpack .../qrencode_4.0.2-2_amd64.deb ...


Google Lens
QR code: Text
Search
Copy text
Translate
Hello World!

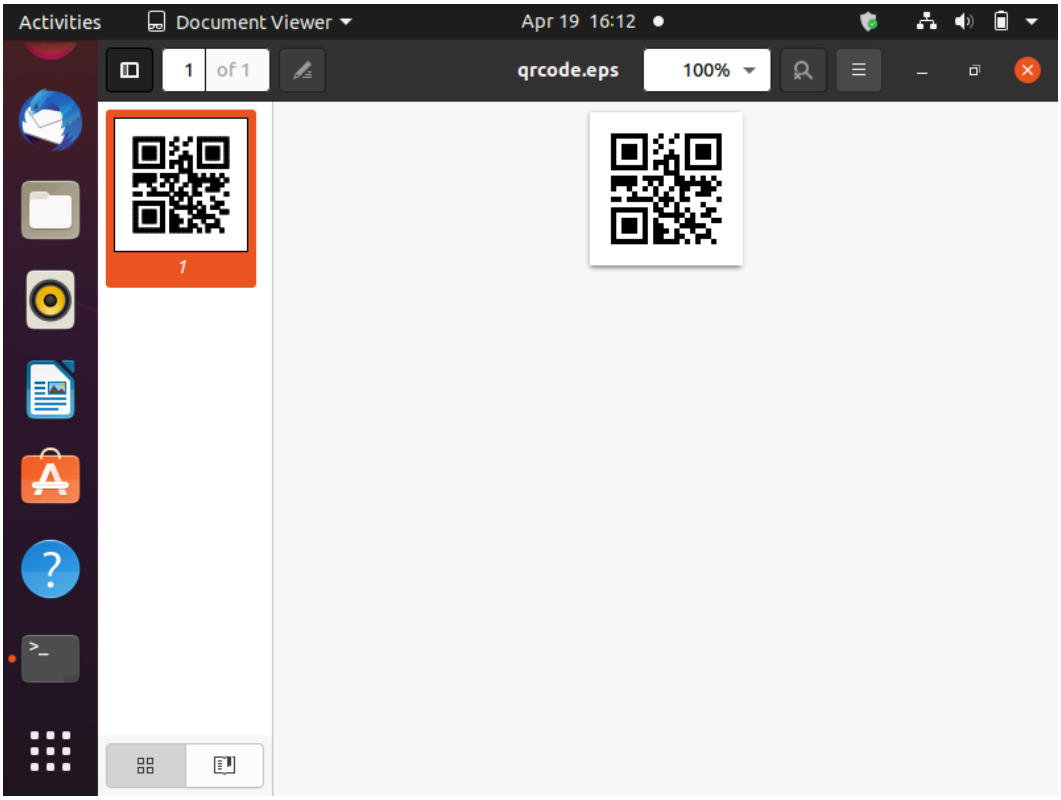
$ qrencode -o qrcode.png 'Hello World!'

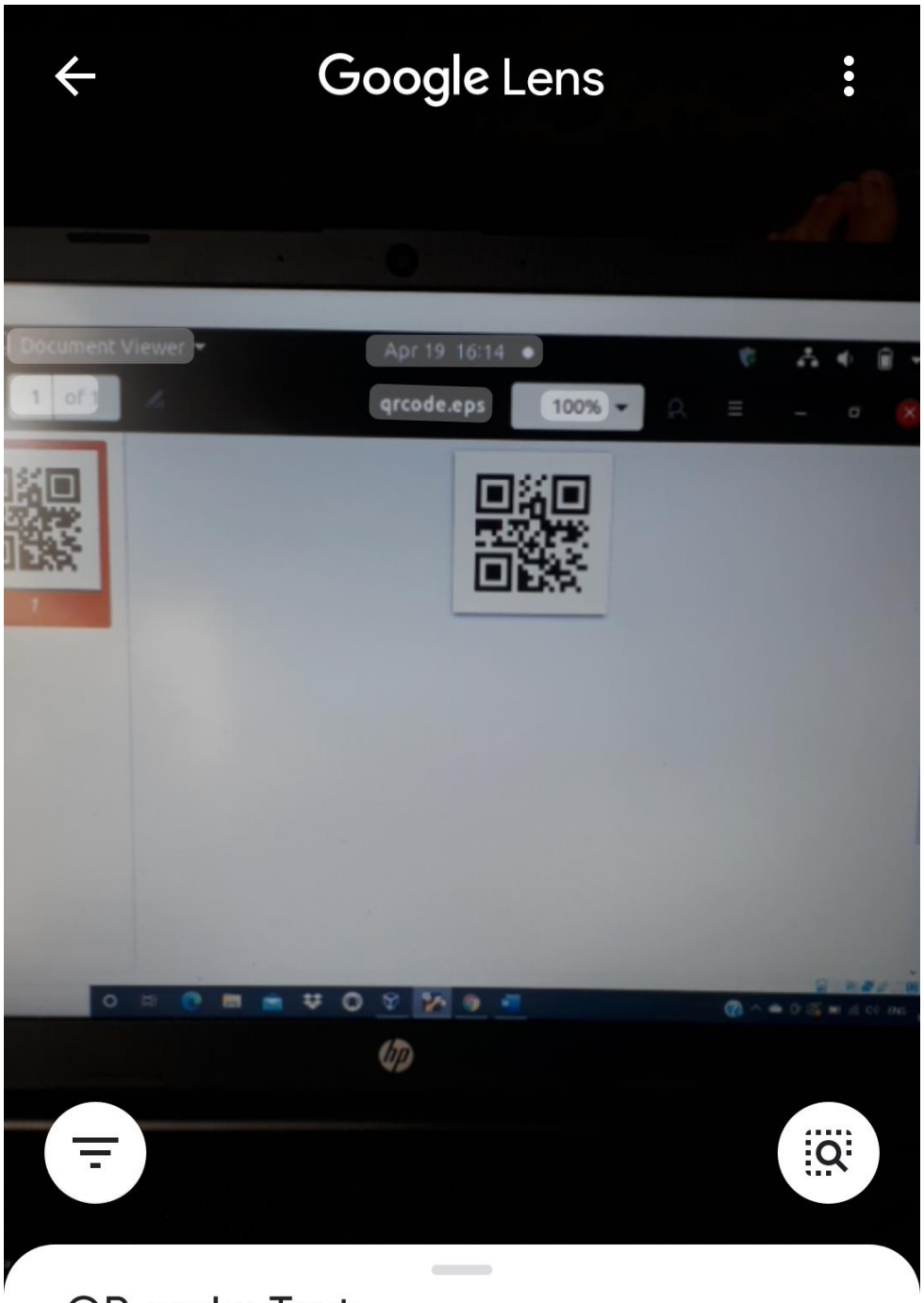


QR Code relevant Hello World will pop up: Qrencode encoding the string "Hello World!" as a QR code and storing it in the qrcode.png file.

The generated QR code ends up in the *qrcode.png* file. If the file already exists, it's overwritten without prompting. If you have at least version 3.3.0 (*qrencode -V*), Qrencode can generate an EPS graphic,

$ qrencode -t EPS -o qrcode.eps 'Hello World!'


Activities
Document Viewer
Apr 19 16:12
1
of 1
qrcode.eps
100%
1



QR code: Text

Search Copy text Translate

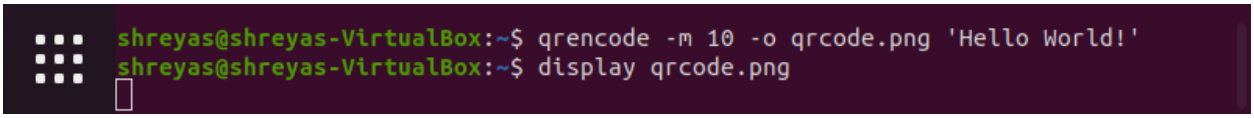
Hello World!

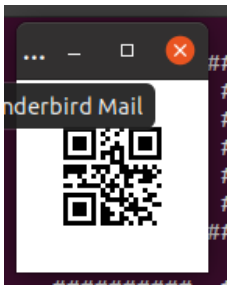
or ASCII output.

Outputting QR codes in ASCII characters just for fun. Each # corresponds to a dot.

In the QR code image, the software creates a white border the width of one dot. This facilitates the process of deciphering the code for programs or the smartphone later on. If desirable, you can increase or decrease the edge with the *-m* parameter; in the following example the border width would be 10 code pixels:

$ qrencode -m 10 -o qrcode.png 'Hello World!'





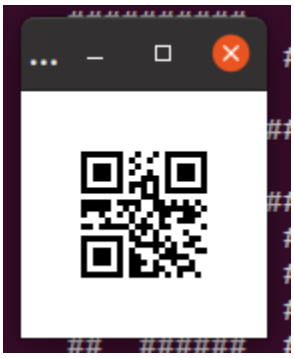
If you are saving the QR code in PNG format, the *-s* parameter specifies the height of a black QR code pixel. By default, Qrencode draws every black dot three by three pixels. The program creates a slightly smaller QR code with quite a wide margin with the following command:

$ qrencode -s 2 -m 10 -o qrcode.png 'Hello World!'

```
^Cshreyas@shreyas-VirtualBox:~$ qrencode -s 2 -m 10 -o qrcode.png 'Hello World!
shreyas@shreyas-VirtualBox:~$ display qrcode.png
```

```
shreyas@shreyas-VirtualBox:~$ qrencode -s 10 -m 10 -o qrcode.png 'Hello World!'
shreyas@shreyas-VirtualBox:~$ display qrcode.png
```

A white border appears around the QR code that is exactly 20 screen pixels wide (10 QR code dots in width, with each dot two pixels). If you are creating EPS images, the only parameter available is *-m*; *-s* moves the entire QR code out of the image in this case. To control the resolution of PNGs further, use the *-d* parameter to define the dpi.



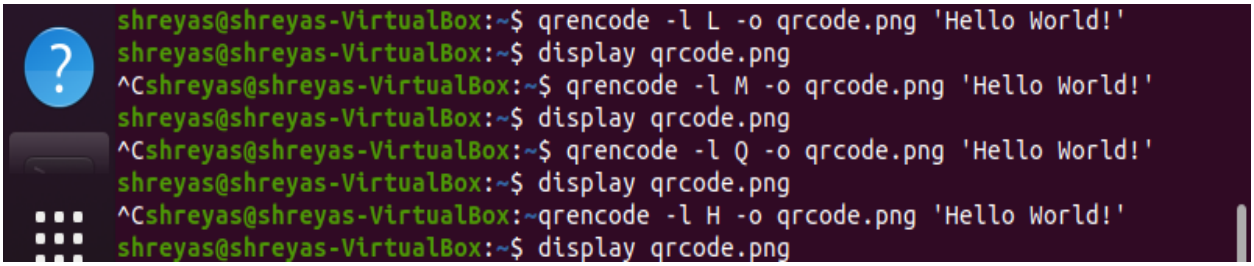
Tolerance

Besides the specified data, the QR code contains additional error correction information. If a portion of the image is damaged, it allows you to reconstruct the missing or illegible data. The more additional information the QR code contains, the more heavily damaged it can be without becoming useless.

Increasing the error tolerance increases the size of the image because you need more black dots

Both QR codes contain the text Hello World!; the one on the left uses the highest error correction level, H, and therefore is more robust – but also larger.

The QR code standard thus uses four levels of error correction: *H* level allows you to read all data if 30 percent of the QR code is destroyed, *Q* level if 25 percent is unreadable, and *M* level if just 15 percent is unintelligible. At the lowest level, *L*, only 7 percent of the data can be faulty. In Qrencode, the *-l* parameter selects the error correction level. The possible values are pretty much what you would expect: *L*, *M*, *Q*, and *H*.



L: M: Q: H:

Of the tested programs, Portable QR-Code Generator offers the greatest functionality. Data entry is also convenient in a variety of tabs. Qrencode creates QR codes more quickly, and it is easy to scripted so you generate QR codes in batches. KBarcode4-light is deprecated, but it is the only program that can generate PDF files; the EPS images created by Qrencode offer similar output.

EXERCISE

TPO of SPIT has asked you to submit simple QR code of your CV so that company can make off line process to create short list of potential candidates. Hence as a part of exercise create CV showing your Profile, Ability and Capability and generate QR code of CV. This QR code can only be shared between you and company. You will provide your digital signature as an authentication to company. Once authentication is successful company is able to scan this QR code of your CV.

Show Implementation steps that you will generate QR code and sign it with your digital signature. Share it with your colleague from same lab with who is capable to verify your digital signature. Successful verification makes scanning of QR code feasible. Show you creativity in terms of robustness of generating digital signature, generating of QR code and Scanning of QR code if relevant secret key is entered.

Sender Code:

```
import os
import base64
import pyqrcode
from Crypto.Hash import SHA256
from Crypto.Signature import PKCS1_v1_5
from Crypto.PublicKey import RSA
from Crypto.Cipher import AES

key = input('Enter 16 byte key: ')
key = key.encode()

with open("resume.txt", "r") as f:
    resume_t = f.readlines()

resume = ""
resume.join(resume_t)
```

```
img = pyqrcode.create(resume)
img.svg("qr_resume.svg", scale=8)
with open("qr_resume.svg", "rb") as image_file:
    message = base64.b64encode(image_file.read())

# Encrypt the image with AES
cipher = AES.new(key, AES.MODE_CFB)
ciphered_data = cipher.encrypt(message)

file_out = open('encrypted.bin', "wb")
file_out.write(cipher.iv)
file_out.write(ciphered_data)
file_out.close()


h = SHA256.new(ciphered_data)

# Load private key and sign message
private_key = False
with open ("priv.pem", "r") as myfile:
    private_key = RSA.importKey(myfile.read(), passphrase = "Harsh")

signature = PKCS1_v1_5.new(private_key).sign(h)

with open("signature.bin", "wb") as text_file:
    text_file.write(signature)
```

Receiver Code:

```
import os
import base64
from Crypto.Hash import SHA256
from Crypto.Signature import PKCS1_v1_5
from Crypto.PublicKey import RSA
from Crypto.Cipher import AES

key = input('Enter key: ')
key = key.encode()

with open("signature.bin", "rb") as signed:
   signature = signed.read()

with open("encrypted.bin", "rb") as file_in:
    iv = file_in.read(16)
    ciphered_data = file_in.read()
```

```
# Read public key from file
public_key = False
with open ("pub.pem", "r") as myfile:
    public_key = RSA.importKey(myfile.read())

verifier = PKCS1_v1_5.new(public_key)

h = SHA256.new(ciphered_data)

verified = verifier.verify(h, signature)
assert verified, 'Signature not verfied!'

try:
    cipher = AES.new(key, AES.MODE_CFB, iv=iv)
    original_data = cipher.decrypt(ciphered_data)
    with open("qr_decoded.svg", "wb") as fh:
        fh.write(base64.decodebytes(original_data))
except ValueError:
    print('The key entered is wrong!')
```

Output:

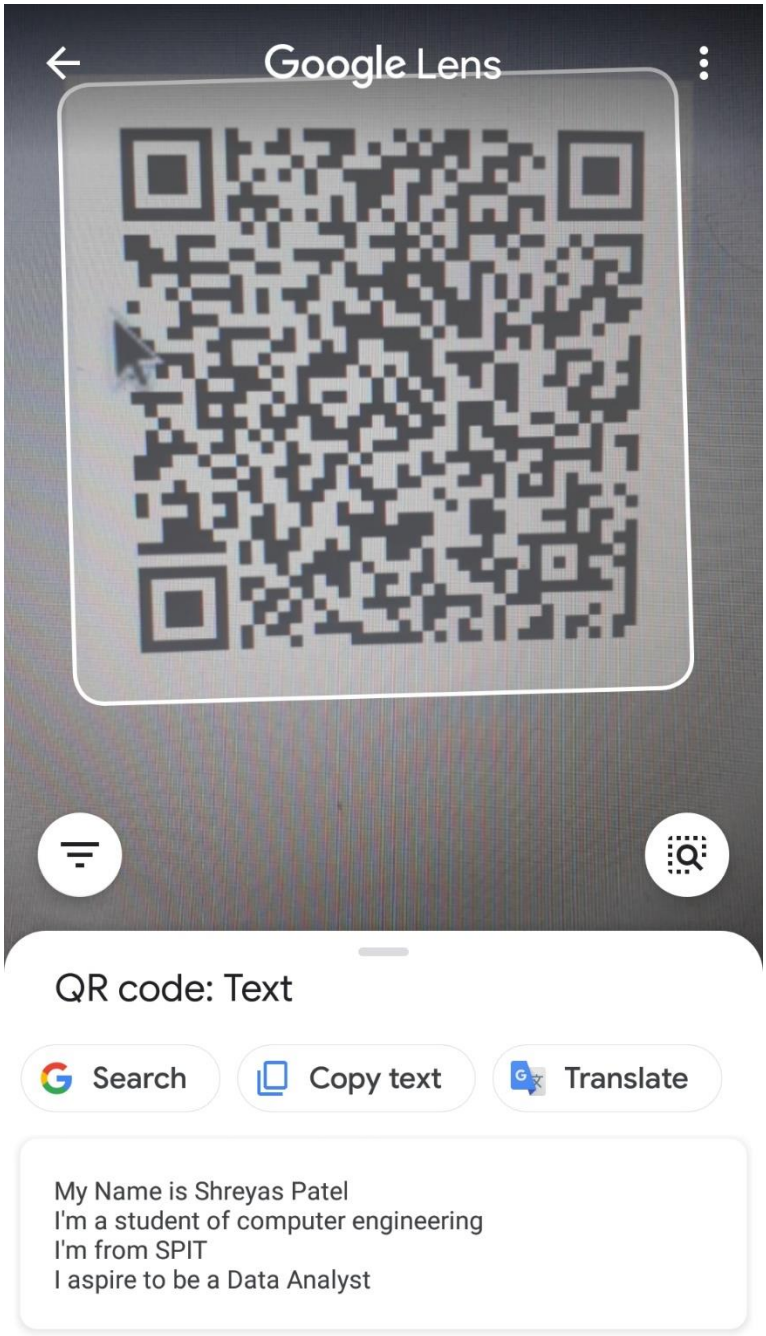
C:\Users\Shreyas\OneDrive\Desktop python sender.py
Enter 16 byte key : asdfghjklqwertui

C:\Users\Shreyas\OneDrive\Desktop python receiver.py
Enter key : asdfghjklqwertyu
The key entered is wrong!

C:\Users\Shreyas\OneDrive\Desktop python receiver.py
Enter key : asdfghjklqwertui

Sent QR Code:

Received QR Code:



Following is for your information only, but if you would like to try you can give a go….

INTRUDERS

QR codes are quite robust against destructive behavior, which allows you put a logo or image in the middle. If the "foreign body" is small enough, the code still works, and it looks pretty, too.

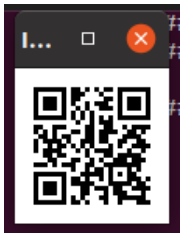
None of the generators available for Linux can add images like this, however, which only leaves you with the option of switching to a web service or editing the image in another application. In the latter case, you will want to create the QR code with error correction level H as a precaution and then check that it works with a smartphone.

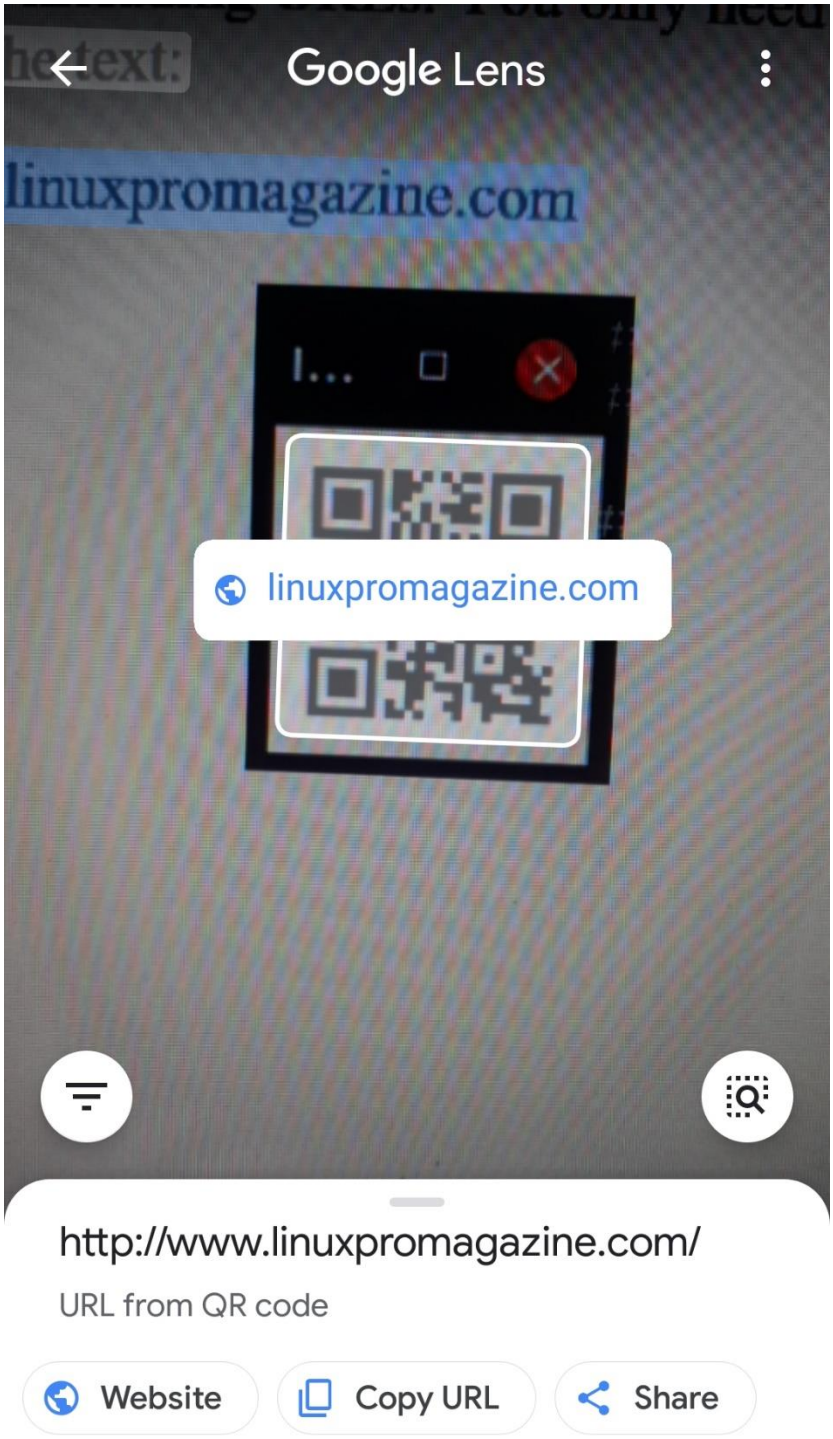
The level you choose depends on the proposed use. If you expect the QR code to be exposed to external influences (e.g., wind, weather, graffiti) or if you are planning to manipulate the image deliberately (see the "Intruders" box), you will want the highest level, *H*. On a business card, the lowest level, *L*, is sufficient. If in doubt, simply omit the *-l* parameter. The program then uses level *L* by default.

Variety

Qrencode lets you encode any text, including URLs. You only need to use single quotes if blanks and non-standard characters appear in the text:

```
$ qrencode -o lpm.png http://www.linuxpromagazine.com
```

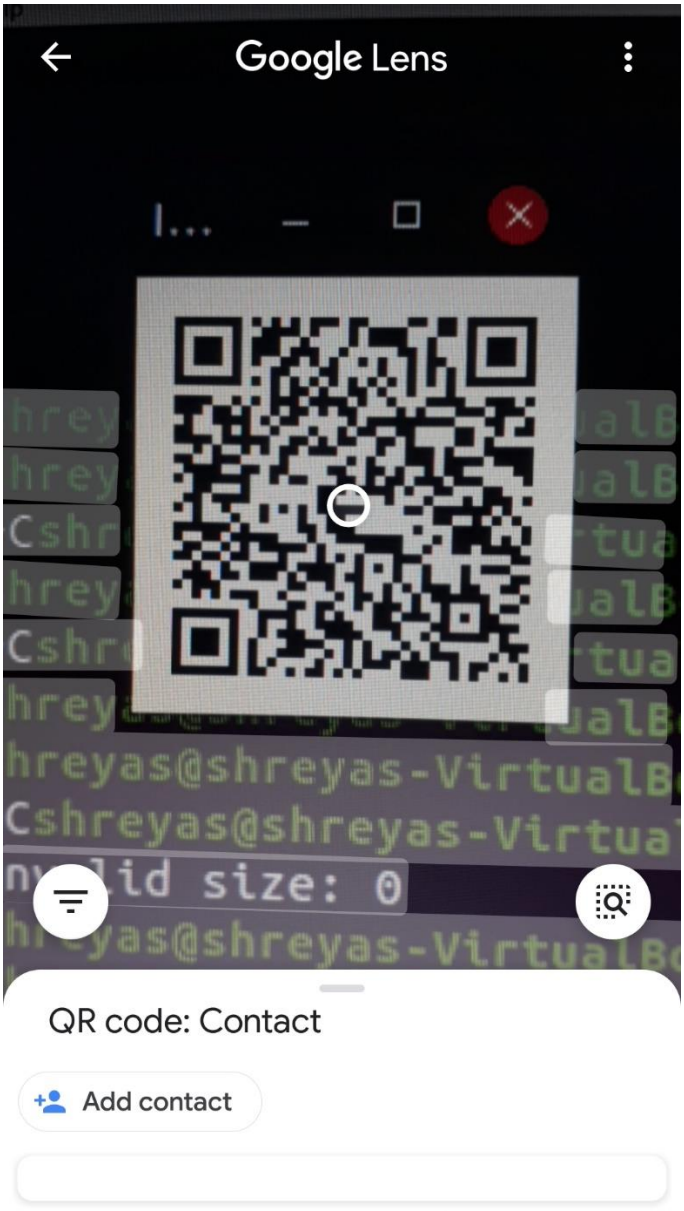




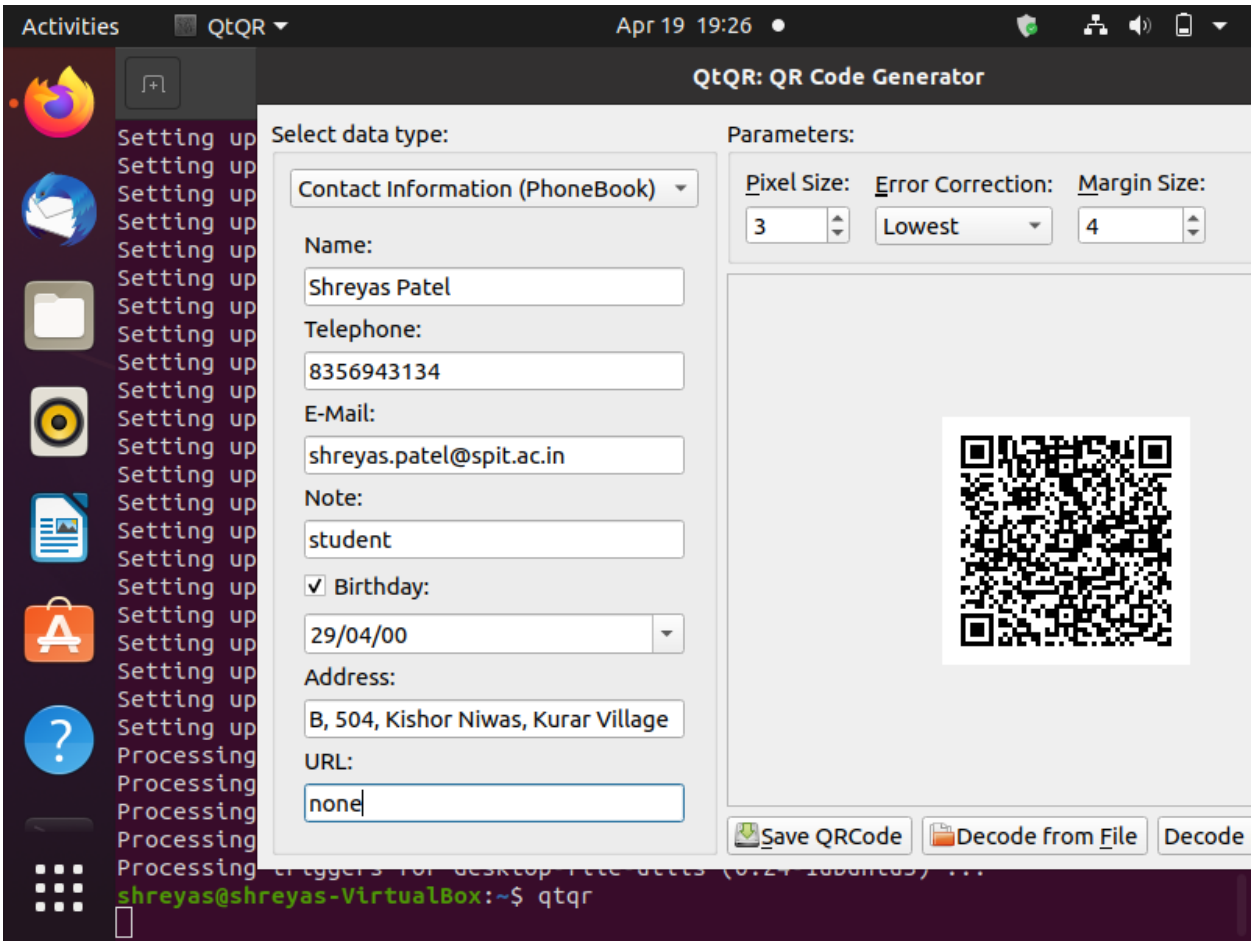
All modern QR code readers recognize that this is a web address and will offer to open it in the browser.

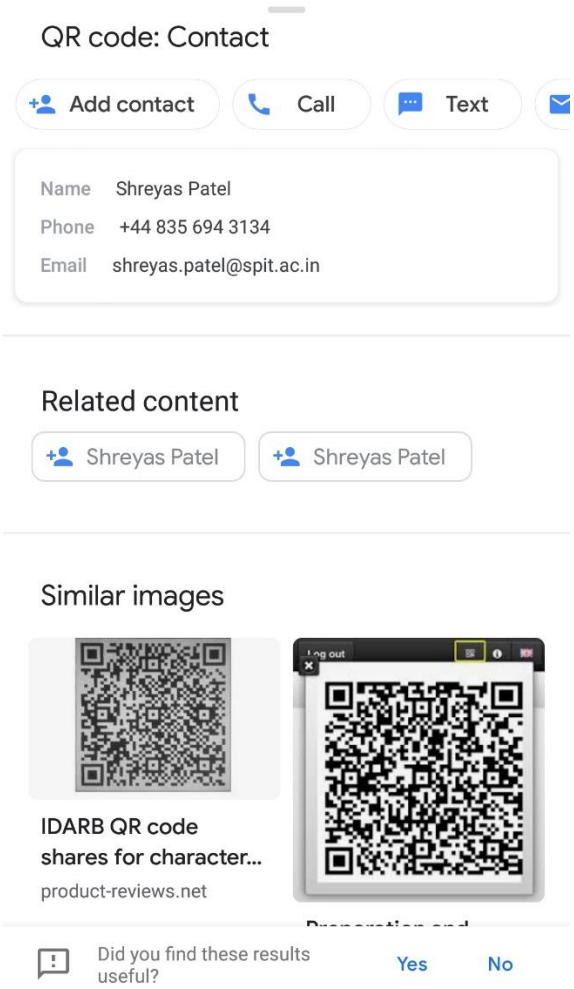
Getting an QR-encoded address automatically to enter a smartphone owner's address book is more difficult. In this case, you need to enter your address and some additional, cryptic information:

$ qrencode -o address.png 'BEGIN:VCARD VERSION:4.0 FN: N:Schuermann;Tim;;; ADR:;;Putzbrunner Str. 71;Munich;;81739; END:VCARD'



Typing this is not exactly easy, so to make life easier, you might prefer a QR code generator with a graphical user interface, such as QtQR.


Activities
QtQR
Apr 19 19:26
QtQR: QR Code Generator
Select data type:
Contact Information (PhoneBook)
Name:
Shreyas Patel
Telephone:
8356943134
E-Mail:
shreyas.patel@spit.ac.in
Note:
student
Birthday:
29/04/00
Address:
B, 504, Kishor Niwas, Kurar Village
URL:
none
Parameters:
Pixel Size:
3
Error Correction:
Lowest
Margin Size:
4
Save QRCode
Decode from File
Decode
Setting up
Processing
shreyas@shreyas-VirtualBox:~$ qtqr



Of the tested programs, Portable QR-Code Generator offers the greatest functionality. Data entry is also convenient in a variety of tabs. Qrencode creates QR codes more quickly, and it is easy to scripted so you generate QR codes in batches. KBarcode4-light is deprecated, but it is the only program that can generate PDF files; the EPS images created by Qrencode offer similar output.

Vulnerabilities: https://analyticsindiamag.com/25-years-of-qr-codes-a-look-at-vulnerabilities-of-this-popular-payment-method/

Currently, over 23% of Trojans and viruses are transmitted via QR codes. On the 25th anniversary of QR codes, its creator, Masahiro Hara wants to make QR scanning more secure.

Usually, in the case of QR scanning, possible scenarios of attacks can be summarised as follows:

- QR codes cannot be hacked. One way hackers to infiltrate this system by changing the QR code added in the poster. These fake posters can be circulated in the public domains and clueless customers scan these fake QR codes and end up visiting phishing websites.

- This usually happens because of the increase in the number of mobile users. Mobiles make it hard to verify the full link in the address bar. This makes users more vulnerable. When they use this phishing page to login, their passwords are compromised.
- An attacker might set up a fake website and redirect users by changing the QR Code. This is dangerous if some form of credentials are needed to access the website. The user has no possibility to verify that the link is not modified.
- SQL injection is another form of attack that occurs when SQL queries are made with user input text inserted into the query string. QR code readers are subject to data injection into their structured objects when they attempt to interpret the data of a QR code.
- A malicious party can create a QR code that injects arbitrary strings into a user's data structures potentially causing harm to the user.
- Criminals can simply prepare malicious QR codes and affix them over legitimate codes which may result in victims inadvertently making payments to a criminal rather a legitimate service provider.
- QRLJacking or Quick Response Code Login Jacking is a simple social engineering attack vector capable of session hijacking affecting all applications that rely on "Login with QR code" feature as a secure way to login to accounts.
- QRLJacking attack gives attackers the ability to apply a full account hijacking scenario on the vulnerable Login with QR Code feature resulting in accounts stealing and reputation affection.

CONCLUSION : 1. QR code are a very good way to encrypt and decrypt messages very quickly.

2. The message can be of any type like text, address information, image, phone number, etc.
3. Google lens can be used to decrypt the data from the QRCODE.
4. Through this experiment I learnt the concept of QRCODEs and also executed them effectively.