ASSIGNMENT 1.3
HTTP PROTOCOL, TCP-UDP PROTOCOL & ICMP PROTOCOL

HTTP PROTOCOL

HTTP is an application layer protocol in the Open Systems Interconnection (OSI) network communication model. It lies in the Application Layer. It defines several types of requests and responses. For example, when you want to view some data from a website, you send the *HTTP GET* request. If you want to send some information, like filling out a contact form, you send the *HTTP PUT* request.

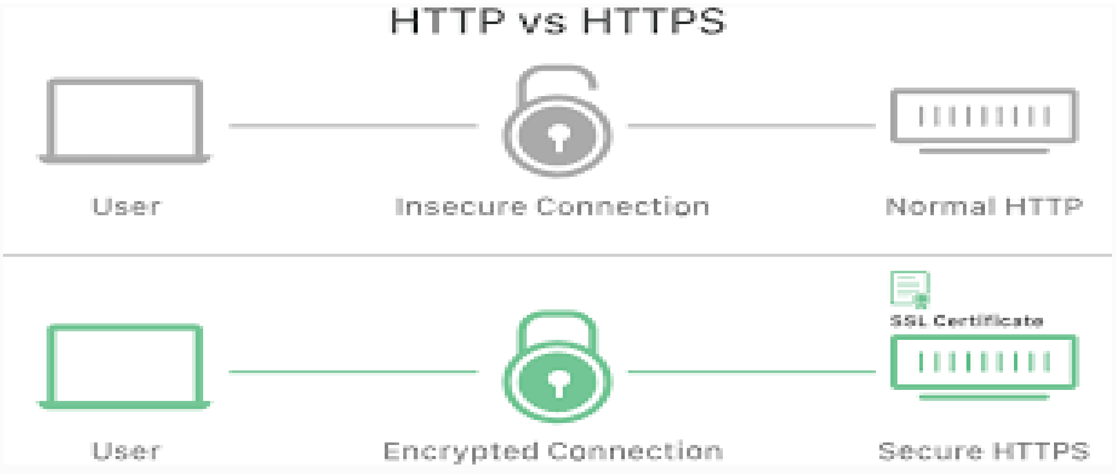
HTTPS PROTOCOL

HTTP transmits unencrypted data, which means that information sent from a browser can be intercepted and read by third parties. This wasn’t an ideal process, so it was extended into HTTPS to add another layer of security to communication. HTTPS combines HTTP requests and responses with SSL and TLS technology.

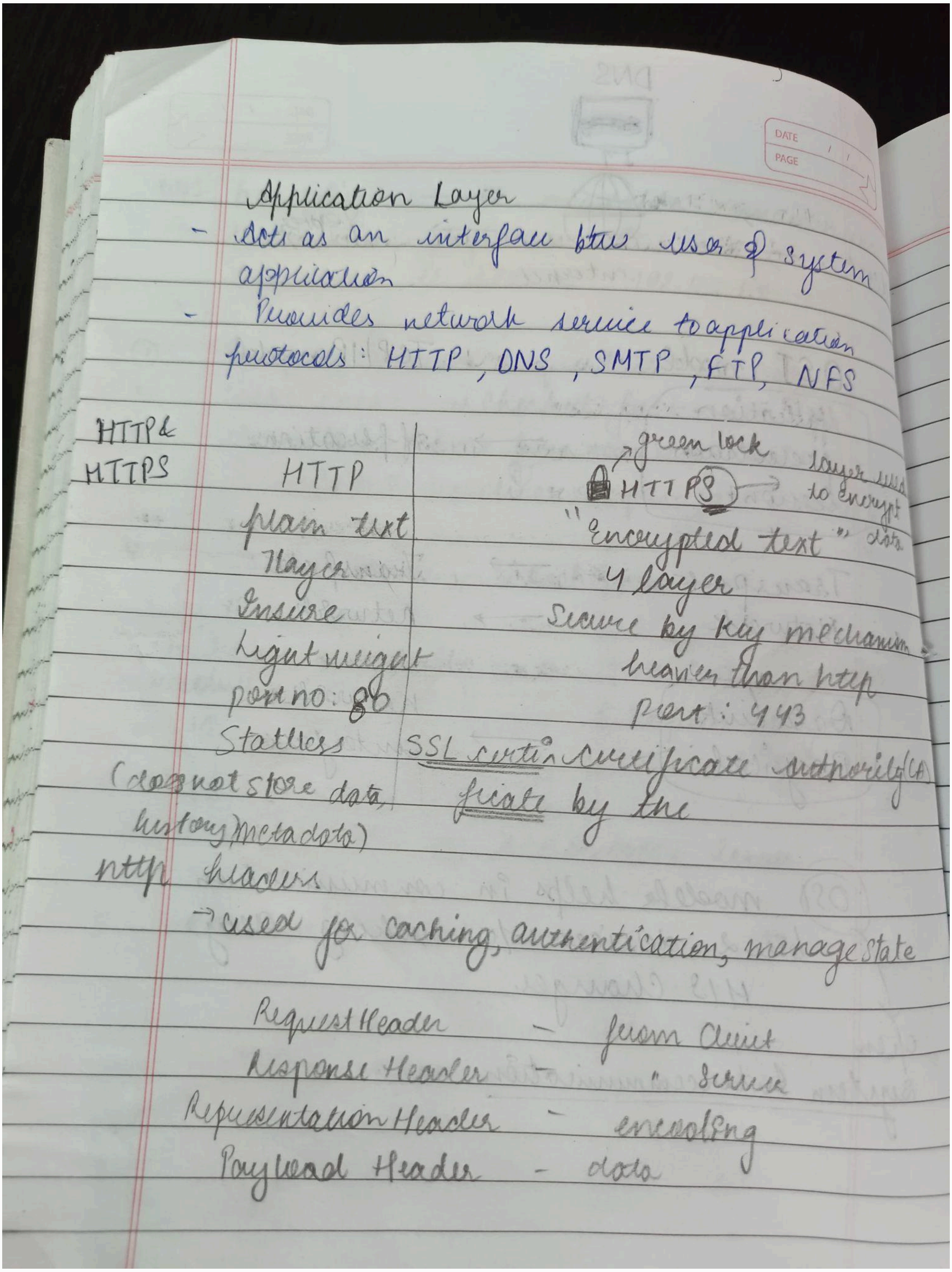
HTTPS websites must obtain an SSL/TLS certificate from an independent certificate authority (CA). These websites share the certificate with the browser before exchanging data to establish trust. The SSL certificate also contains cryptographic information, so the server and web browsers can exchange encrypted or scrambled data.

Working of HTTPS Protocol:

1. You visit an HTTPS website by typing the *https://* URL format in your browser's address bar.
2. The browser attempts to verify the site's authenticity by requesting the server's SSL certificate.
3. The server sends the SSL certificate that contains a public key as a reply.
4. The website's SSL certificate proves the server identity. Once the browser is satisfied, it uses the public key to encrypt and send a message that contains a secret session key.
5. The web server uses its private key to decrypt the message and retrieve the session key. It then encrypts the session key and sends an acknowledgment message to the browser.
6. Now, both browser and web server switch to using the same session key to exchange messages safely.



ASSIGNMENT 1.3
HTTP PROTOCOL, TCP-UDP PROTOCOL & ICMP PROTOCOL



Application Layer

- Acts as an interface btw user of system application
- Provides network service to application protocols : HTTP , DNS , SMTP , FTP, NFS

HTTP & HTTPS

HTTP	HTTPS
plain text	"Encrypted text"
7layer	4 layer
Insure	Secure by key mechanism
Light weight	heavier than http
port no: 80	port : 443
Statless (does not store data, history, metadata)	SSL certificate by the certificate authority(CA)

HTTPS → green lock

S → layer used to encrypt data

http headers

→ used for caching, authentication, manage state

Request Header	-	from Client
Response Header	-	" Server
Representation Header	-	encoding
Payload Header	-	data

TCP PROTOCOL

TCP, short for Transmission Control Protocol, is a connection-oriented protocol that ensures reliable and ordered delivery of data packets between devices. It provides error detection, flow control, and congestion control mechanisms to guarantee the successful transmission of data.

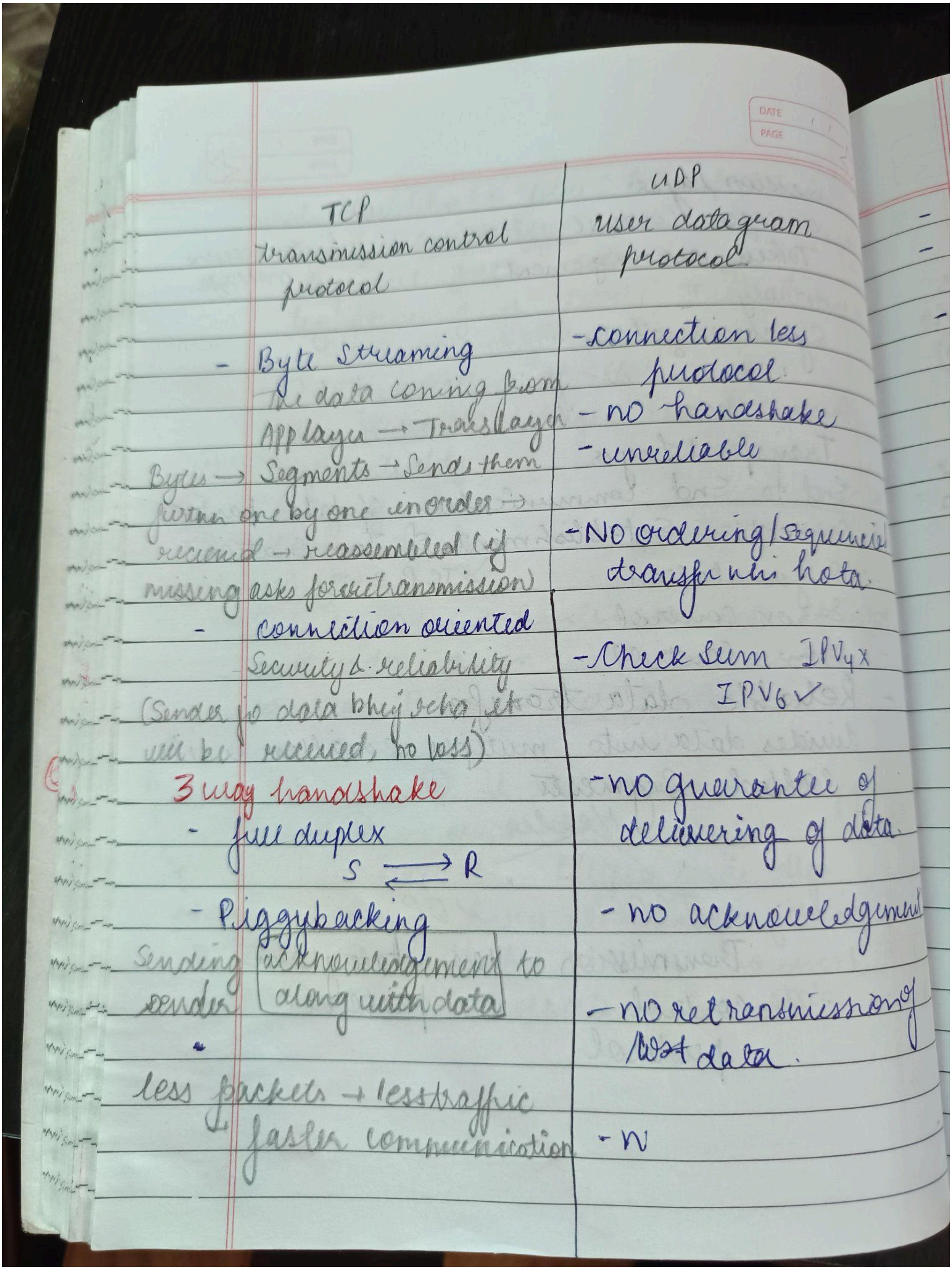
TCP establishes a connection between a sender and a receiver before transferring data, ensuring that packets arrive in the correct order and without errors. It retransmits lost or corrupted packets and acknowledges the receipt of data, making it highly reliable.

UDP PROTOCOL

User Datagram Protocol is a transport layer protocol used to transmit data. UDP, unlike TCP, has less overhead for establishing, maintaining, or terminating a connection; hence, it is faster than TCP. In UDP, the data is continuously sent to the recipient irrespective of whether it was received or not.

UDP can be considered a lightweight protocol because of its lesser responsibilities while delivering data. It is unreliable because there is no acknowledgment after receiving the data successfully, which means the sender won’t know if the data was lost during communication or received by the recipient.

ASSIGNMENT 1.3
HTTP PROTOCOL, TCP-UDP PROTOCOL & ICMP PROTOCOL



TCP transmission control protocol	UDP user datagram protocol
- Byte Streaming the data coming from App layer → Trans layer Bytes → Segments → Sends them further one by one in order → recieved → reassembled (if missing asks for retransmission)	-connection less protocol - no handshake - unreliable
	-No ordering/sequencie transfer nhi hota.
- connection oriented Security & reliability (Sender jo data bhej raha it will be received, no loss)	-Check sum IPV4 x IPV6 ✓
3 way handshake	-no guarantee of delivering of data.
- full duplex $S \rightleftarrows R$	
- Piggybacking Sending acknowledgement to sender along with data	- no acknowledgement
•	-no retransmission of ~~lost~~ lost data.
less packets → less traffic ↓ faster communication	- N

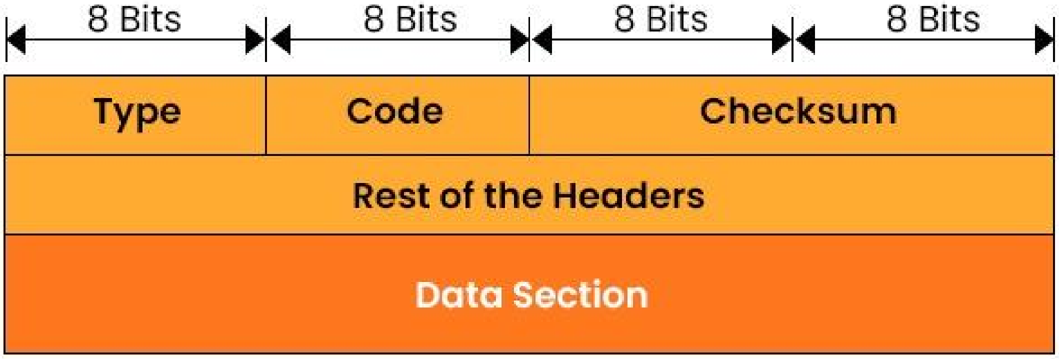
ICMP PROTOCOL

ICMP or Internet Control Message Protocol is a kind of protocol that operates at the network layer on top of the Internet Protocol (IP). IP handles the transportation of data packets between sources and destinations, whereas ICMP is responsible for transmitting control and error messages among network devices. For instance, if a router encounters an issue while forwarding an IP packet, it can utilize ICMP to send an error message to the source host. Similarly, if a host wishes to assess connectivity or latency with another host, it can employ ICMP to transmit an echo request and await an echo reply.

The two main functions of ICMP Protocol are:

1. Error reporting
2. Querying

ICMP FORMAT

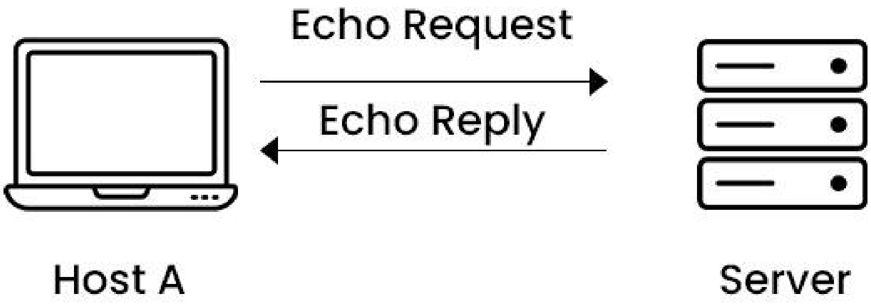


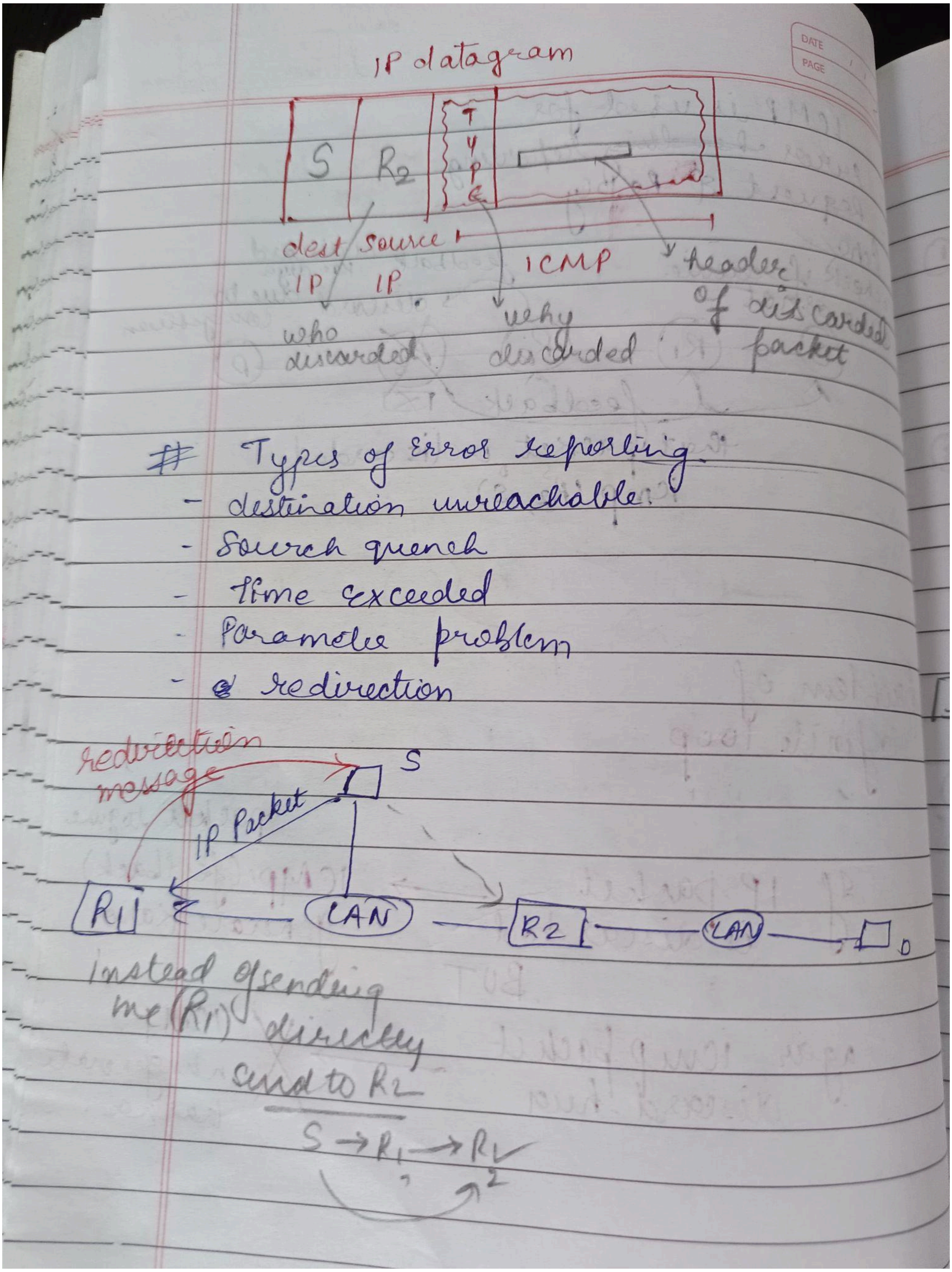
ICMP Message Format

Working of ICMP Protocol:

ICMP operates by exchanging messages between network devices using IP datagrams. An ICMP message comprises an IP header followed by an ICMP header and data. The IP header contains details like source and destination addresses, protocol number (1 for ICMP) and a checksum. The ICMP header includes information such as message type, code, another checksum and optional identifier.The data section carries information depending on the type and code of the message.

When a network device sends an ICMP message, it packages it within an IP datagram. Forward it to the specified destination address mentioned in the IP header. Upon receiving an ICMP message, a network device unpacks it from the IP datagram. It then checks the type and code fields in the ICMP header. Based on these values, different actions or responses may be triggered.





IP datagram

S
R2
T
y
p
e
dest IP
source IP
who discarded
ICMP
why discarded
header of discarded packet

Types of Error reporting

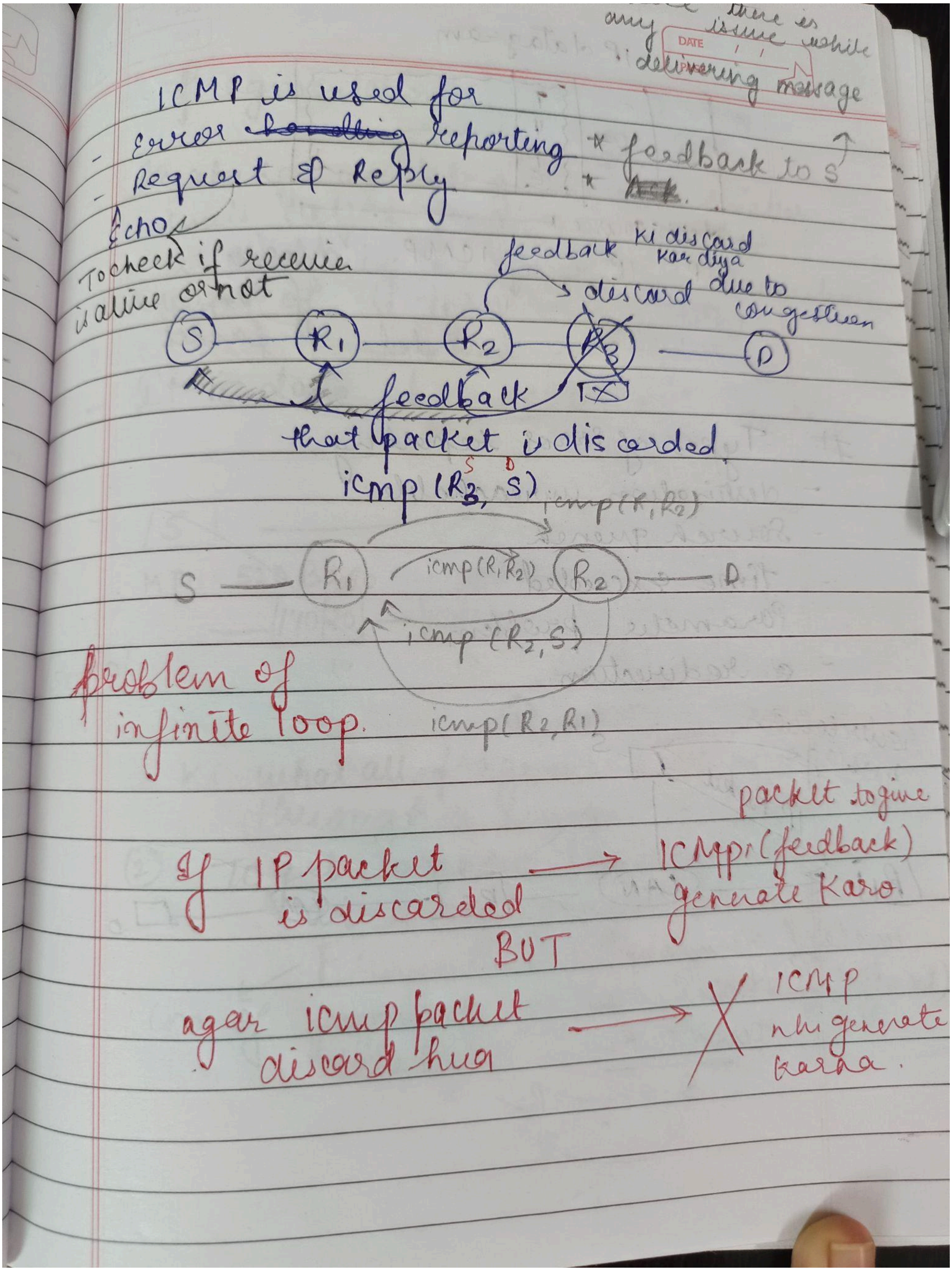
- destination unreachable
- Sourch quench
- time exceeded
- Paramete problem
- redirection

redirection message
S
IP Packet
R1
LAN
R2
LAN
D

instead of sending me(R1) directly send to R2

$S \rightarrow R_1 \rightarrow R_2$

ASSIGNMENT 1.3
HTTP PROTOCOL, TCP-UDP PROTOCOL & ICMP PROTOCOL



there is any issue while delivering message

ICMP is used for
- Error ~~handling~~ reporting
- Request & Reply

* feedback to S

* ~~Ack~~.

Echo

To check if receiver is alive or not

S
R1
R2
R3
D
feedback
discard
ki discard kar diya
due to congestion
feedback

that packet is discarded.

icmp (R3, S)

S
D
S
R1
R2
D
icmp(R1,R2)
icmp(R1R2)
icmp (R2,S)
icmp(R2,R1)

problem of infinite loop.

If IP packet is discarded → ICMP (feedback) packet to give generate Karo

BUT

agar icmp packet discard hua → X ICMP nhi generate karna.