

Wireshark Experiment – 04

Transport Layer

Name: Pratik P Patil
USN: 02FE21BEC063
Roll No: 53
Div: A

Transport Layer

The Transport Layer is the fourth layer of the OSI (Open Systems Interconnection) model and plays a crucial role in end-to-end communication between devices in a network. Its primary purpose is to provide reliable data transfer, error recovery, and flow control, ensuring that data is delivered accurately and efficiently between applications on different devices.

Key Functions of the Transport Layer

- Breaks down large data streams into smaller segments for transmission.
- Provides direct communication between applications on different devices.
- Establishes a reliable connection before data transfer and ensures data integrity.
- Ensures data integrity by detecting and correcting errors during transmission.
- Regulates the data flow to prevent the sender from overwhelming the receiver.
- Allows multiple applications to use the network simultaneously by distinguishing between them using port numbers.

Protocols in the Transport Layer

1. Transmission Control Protocol (TCP):

- Connection-oriented protocol.
- Provides reliable data delivery, error checking, and retransmission.
- Used in applications like web browsing (HTTP), email (SMTP), and file transfers (FTP).

2. User Datagram Protocol (UDP):

- Connectionless protocol.
- Focuses on low latency and minimal overhead.
- Commonly used in applications like video streaming, online gaming, and VoIP.

1. For each of the first 8 Ethernet frames, specify the source of the frame (client or server), determine the number of SSL records that are included in the frame, and list the SSL record types that are included in the frame. Draw a timing diagram between client and server, with one arrow for each SSL record.

ANS:

No.	Time	Source	Destination	Protocol	Length	Info
106	21.805705	128.238.38.162	216.75.194.220	SSLv2	132	Client Hello
108	21.830201	216.75.194.220	128.238.38.162	SSLv3	1434	Server Hello
111	21.853520	216.75.194.220	128.238.38.162	SSLv3	790	Certificate, Server Hello Done
112	21.876168	128.238.38.162	216.75.194.220	SSLv3	258	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
113	21.945667	216.75.194.220	128.238.38.162	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
114	21.954189	128.238.38.162	216.75.194.220	SSLv3	806	Application Data
122	23.480352	216.75.194.220	128.238.38.162	SSLv3	272	Application Data
149	23.559497	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
158	23.560866	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
163	23.566451	128.238.38.162	216.75.194.220	SSLv3	156	Client Hello
165	23.586650	216.75.194.220	128.238.38.162	SSLv3	1329	Application Data
169	23.591590	216.75.194.220	128.238.38.162	SSLv3	200	Server Hello, Change Cipher Spec, Encrypted Handshake Message
171	23.599417	128.238.38.162	216.75.194.220	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
172	23.602696	128.238.38.162	216.75.194.220	SSLv3	470	Application Data
176	23.621694	128.238.38.162	216.75.194.220	SSLv3	156	Client Hello
178	23.627217	216.75.194.220	128.238.38.162	SSLv3	378	Application Data
184	23.646644	216.75.194.220	128.238.38.162	SSLv3	200	Server Hello, Change Cipher Spec, Encrypted Handshake Message
188	23.662642	128.238.38.162	216.75.194.220	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
189	23.665695	128.238.38.162	216.75.194.220	SSLv3	476	Application Data
190	23.666238	128.238.38.162	216.75.194.220	SSLv3	156	Client Hello

> Frame 106: 132 bytes on wire (1056 bits), 132 bytes captured (1056 bits)
> Ethernet II, Src: IBM_10:60:99 (00:09:6b:10:60:99), Dst: All-HSRP-routers_00 (00:00:0c:07:ac:00)
> Internet Protocol Version 4, Src: 128.238.38.162, Dst: 216.75.194.220
> Transmission Control Protocol, Src Port: 2271, Dst Port: 443, Seq: 1, Ack: 1, Len: 78
> Transport Layer Security

```
0000 00 00 0c 07 ac 00 00 09 6b 10 60 99 08 00 45 00 .....k-...E-
0010 00 76 48 28 40 00 80 06 6f a1 80 ee 26 a2 d8 4b ...vH(@...o...&K
0020 c2 dc 08 df 01 bb 56 d2 08 c5 4c 9e 64 9f 50 18 ....V...L d P
0030 ff ff e7 55 00 00 80 4c 01 03 00 00 33 00 00 00 ...U...L...3...
0040 10 00 00 04 00 00 05 00 00 0a 01 00 80 07 00 c0 .....@...d...b
0050 03 00 80 00 09 06 00 40 00 00 64 00 00 62 00 .....@...d...b
0060 00 03 00 00 06 02 00 00 04 00 00 00 13 00 00 .....@...d...b
0070 12 00 00 63 66 df 78 4c 04 8c d6 04 35 dc 44 89 ...cf.xL....S.D
0080 89 46 99 09 .....f..
```

First 8 Frames

No.	Time	Source	Destination	Protocol	Length	Info
106	21.805705	128.238.38.162	216.75.194.220	SSLv2	132	Client Hello
108	21.830201	216.75.194.220	128.238.38.162	SSLv3	1434	Server Hello
111	21.853520	216.75.194.220	128.238.38.162	SSLv3	790	Certificate, Server Hello Done
112	21.876168	128.238.38.162	216.75.194.220	SSLv3	258	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
113	21.945667	216.75.194.220	128.238.38.162	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
114	21.954189	128.238.38.162	216.75.194.220	SSLv3	806	Application Data
122	23.480352	216.75.194.220	128.238.38.162	SSLv3	272	Application Data
149	23.559497	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data

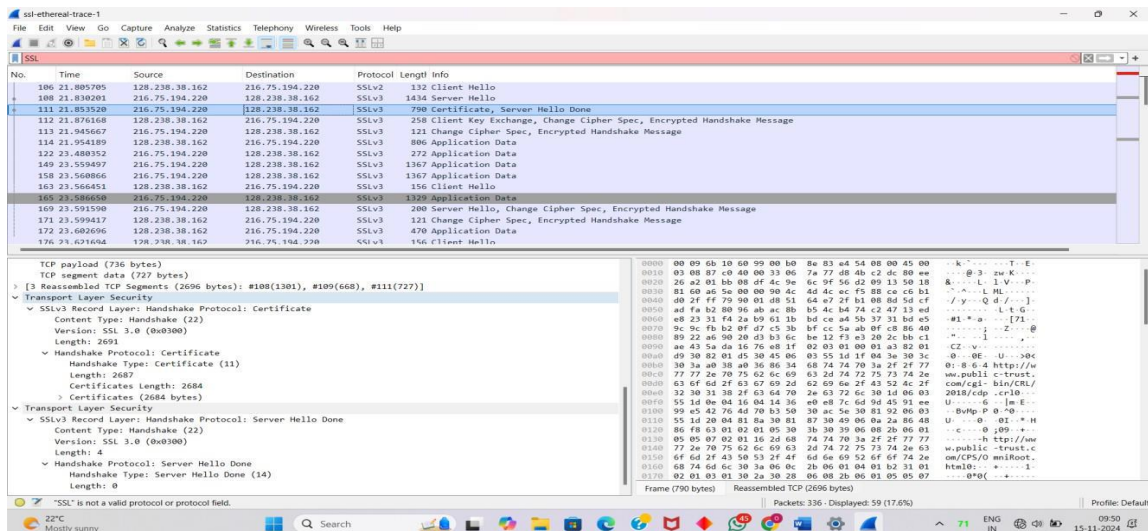
2.Each of the SSL records begins with the same three fields (with possibly different values). One of these fields is “content type” and has length of one byte. List all three fields and their lengths.

ANS:

No.	Time	Source	Destination	Protocol	Length	Info
106	21.805705	128.238.38.162	216.75.194.220	SSLv2	132	Client Hello
108	21.830201	216.75.194.220	128.238.38.162	SSLv3	1434	Server Hello
111	21.853520	216.75.194.220	128.238.38.162	SSLv3	790	Certificate, Server Hello Done
112	21.876168	128.238.38.162	216.75.194.220	SSLv3	258	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
113	21.945667	216.75.194.220	128.238.38.162	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
114	21.954189	128.238.38.162	216.75.194.220	SSLv3	806	Application Data
122	23.480352	216.75.194.220	128.238.38.162	SSLv3	272	Application Data
149	23.559497	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
158	23.560866	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
163	23.566451	128.238.38.162	216.75.194.220	SSLv3	156	Client Hello
165	23.586650	216.75.194.220	128.238.38.162	SSLv3	1329	Application Data
169	23.591590	216.75.194.220	128.238.38.162	SSLv3	200	Server Hello, Change Cipher Spec, Encrypted Handshake Message
171	23.599417	128.238.38.162	216.75.194.220	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
172	23.602696	128.238.38.162	216.75.194.220	SSLv3	470	Application Data
176	23.621694	128.238.38.162	216.75.194.220	SSLv3	156	Client Hello

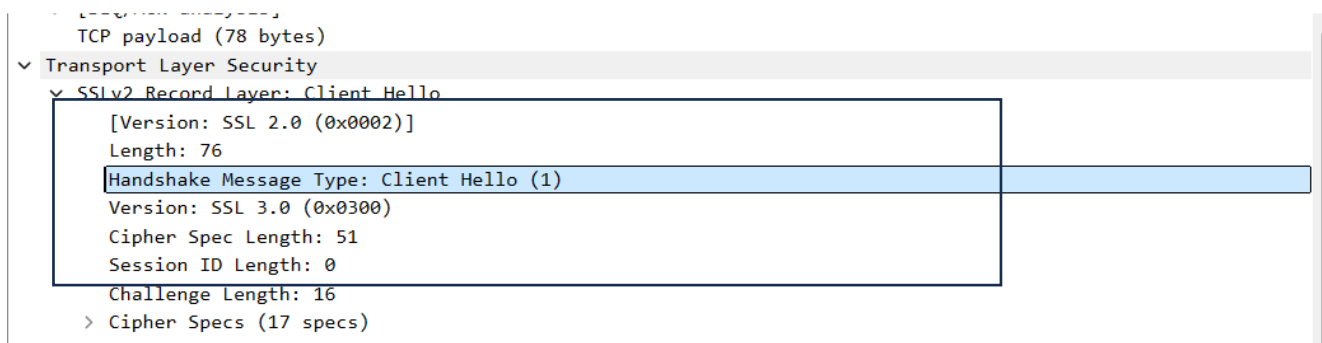
> Flags: 0x018 (PSH, ACK)
Window: 65535
[Calculated window size: 65535]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0xe755 [Unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
> [Timestamps]
> [SEQ/ACK analysis]
TCP payload (78 bytes)
Transport Layer Security
SSLv2 Record Layer: Client Hello
[Version: SSL 2.0 (0x0002)]
Length: 76
Handshake Message Type: Client Hello (1)
Version: SSL 3.0 (0x0300)
Cipher Spec Length: 51
Session ID Length: 0
Challenge Length: 16
> Cipher Specs (17 specs)

```
0000 00 00 0c 07 ac 00 00 09 6b 10 60 99 08 00 45 00 .....k-...E-
0010 00 76 48 28 40 00 80 06 6f a1 80 ee 26 a2 d8 4b ...vH(@...o...&K
0020 c2 dc 08 df 01 bb 56 d2 08 c5 4c 9e 64 9f 50 18 ....V...L d P
0030 ff ff e7 55 00 00 80 4c 01 03 00 00 33 00 00 00 ...U...L...3...
0040 10 00 00 04 00 00 05 00 00 0a 01 00 80 07 00 c0 .....@...d...b
0050 03 00 80 00 09 06 00 40 00 00 64 00 00 62 00 .....@...d...b
0060 00 03 00 00 06 02 00 00 04 00 00 00 13 00 00 .....@...d...b
0070 12 00 00 63 66 df 78 4c 04 8c d6 04 35 dc 44 89 ...cf.xL....S.D
0080 89 46 99 09 .....f..
```



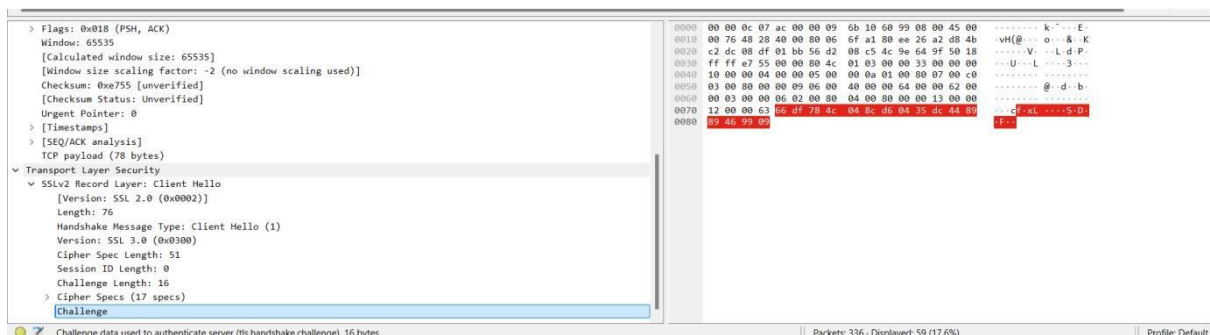
3. Expand the Client Hello record. (If your trace contains multiple ClientHello records, expand the frame that contains the first one.) What is the value of the content type?

ANS:



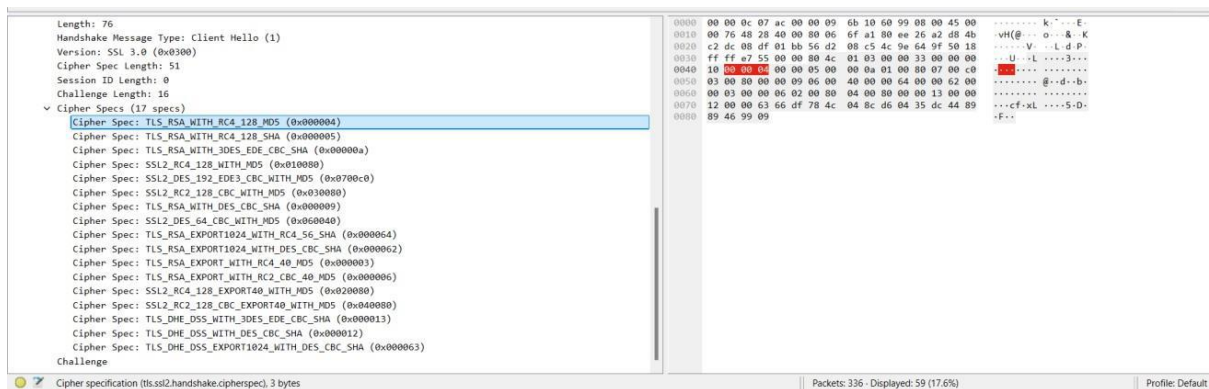
4. Does the Client Hello record contain a nonce (also known as a “challenge”)? If so, what is the value of the challenge in hexadecimal notation?

ANS:



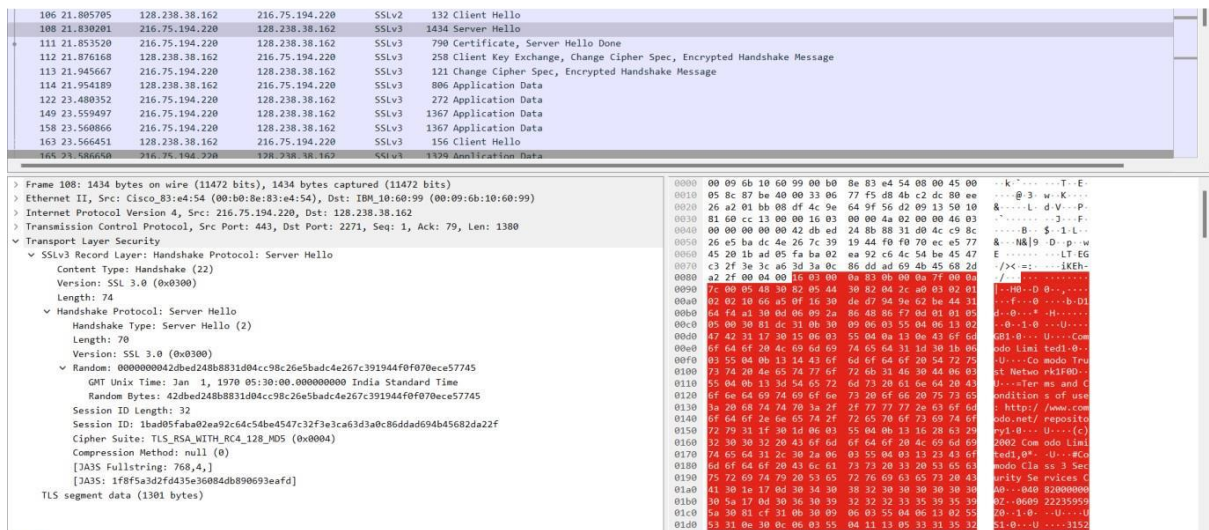
5. Does the Client Hello record advertise the cyber suites it supports? If so, in the first listed suite, what are the public-key algorithm, the symmetric-key algorithm, and the hash algorithm?

ANS:



6. Locate the Server Hello SSL record. Does this record specify a chosen cipher suite? What are the algorithms in the chosen cipher suite?

ANS:



7. Does this record include a nonce? If so, how long is it? What is the purpose of the client and server nonces in SSL?

Yes, this record includes a nonce, as known as Random.

bytes, and it is 28 bytes long (as highlighted above). The purpose of the client and server nonces in SSL is to prevent attacker from replaying or reordering records.

8. Does this record include a session ID? What is the purpose of the session ID?

ANS: Yes, this record includes a Session ID which is 32-bytes long. Its purpose is to allow session resumption, which can significantly reduce the number of time-consuming server handshake to create a new session ID. In the Client Hello record, a nonzero session ID means that the client to resume its previously established session; and a zero session ID means that the client wishes to establish a new session with the server.

9. Does this record contain a certificate, or is the certificate included in a separate record. Does the certificate fit into a single Ethernet frame?

Yes, this record contains a certificate. The certificate is 2684 bytes long, thus it can fit into a single Ethernet frame.

The image shows a Wireshark packet capture of an SSLv3 Record Layer: Handshake Protocol: Certificate. The packet is 736 bytes long and contains a certificate that is 2684 bytes long. The certificate is highlighted in blue. The packet is part of a reassembled TCP segment (2696 bytes) and fits within a single Ethernet frame (790 bytes).

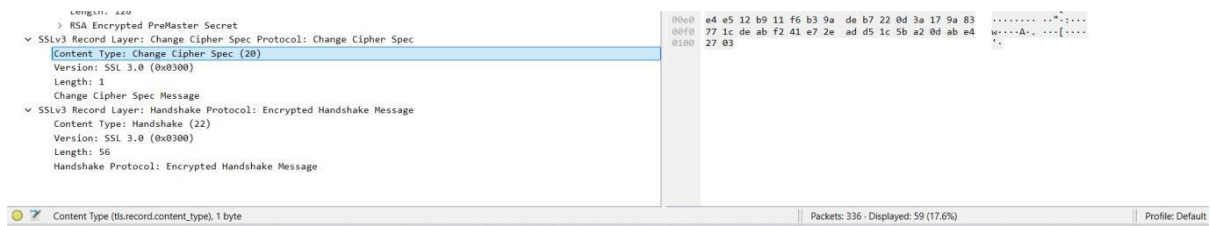
10. Locate the client key exchange record. Does this record contain a pre-master secret? What is this secret used for? Is the secret encrypted? If so, how? How long is the encrypted secret? Change Cipher Spec Record (sent by client) and Encrypted

ANS: Yes, this record contains a pre-master secret (highlighted above). This encrypted pre-master secret is decrypted at the server side and is used to produce a master secret. Then this master secret is used to produce “key block”, which is then sliced and diced into client MAC key, server MAC key, client encryption key, server encryption key, client IV and server IV. The secret is encrypted using server’s public key. The encrypted secret is 128 byte long.

The image shows a Wireshark packet capture of an SSLv3 Record Layer: Handshake Protocol: Client Key Exchange. The record is 132 bytes long and contains a pre-master secret that is 128 bytes long. The pre-master secret is highlighted in blue. The record is part of a reassembled TCP segment (2696 bytes) and fits within a single Ethernet frame (790 bytes).

11. What is the purpose of the Change Cipher Spec record? How many bytes is the record in your trace?

ANS: The purpose of Change Cipher Spec is to indicate change in encryption and authentication algorithms and to update the cipher suite to be used on this connection. This record is only 1 byte long in my trace.



12. In the encrypted handshake record, what is being encrypted? How?

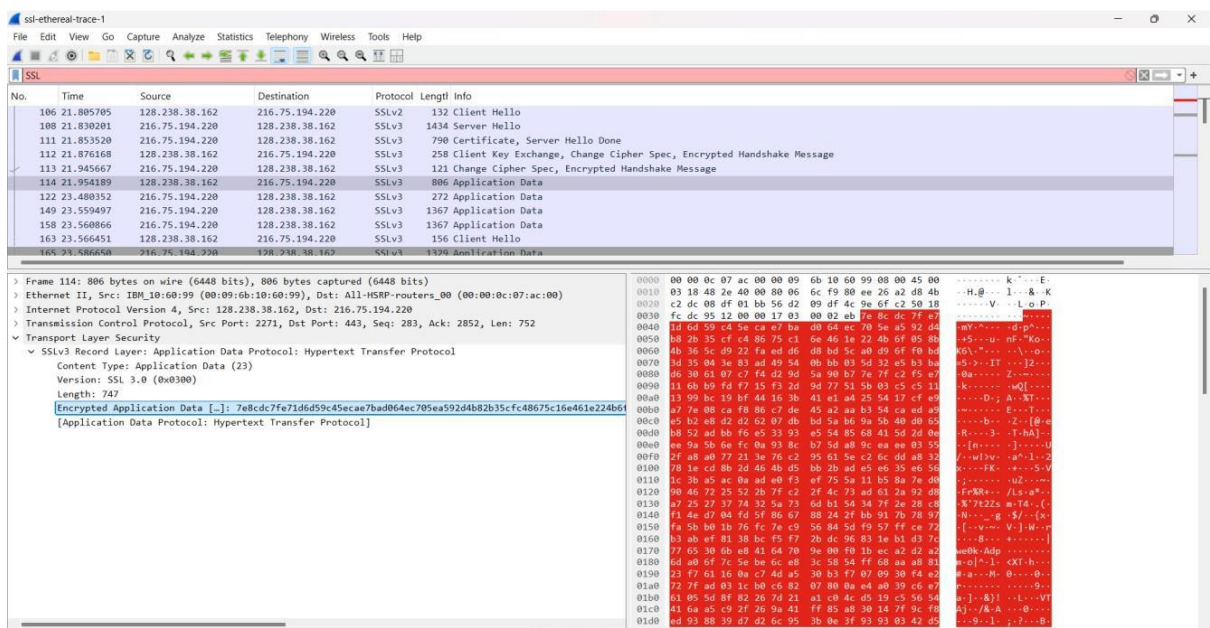
ANS: The sender of this Encrypted Handshake Records and all handshake messages up to but not including this message are encrypted in record. This information is concatenated and hashed using two hash algorithms, MD5 and SHA. The content of this record is the concatenation of these two hash values. The Encrypted Handshake Record is used to verify that key exchange and authentication processes were successful.

13. Does the server also send a change cipher record and an encrypted handshake record to the client? How are those records different from those sent by the client? Application Data

Yes, the server also sends its own Change Cipher Spec and Encrypted Handshake records. The only difference is the sender of this record; the sender is now the server while the sender was the client in previous message.

14. How is the application data being encrypted? Do the records containing application data include a MAC? Does Wireshark distinguish between the encrypted application data and the MAC?

ANS:



The application data is encrypted using the specified algorithms in the chosen cipher suite; in my case, RSA (public-key), 256-bit CBC AES (symmetric)