14 bytes	72 bytes	Optional	Variable length	Variable length
Ethernet Header	NLP basic header	NLP extension header	Transport layer header	Data

FIG. 1

2 bytes	2 bytes	2 bytes	2 bytes	1 byte	1 byte
Version	Service	Flow label	Packet length	Next header	Hop count
NLP source address (32 bytes)					
NLP destination address (32 bytes)					

FIG. 2

Next header (1 byte)	Type (1 byte)	Reserved (2 bytes)
Time stamp (4 bytes)		
Temporary public key (32 bytes)		
Digital signature (64 bytes)		

FIG. 3

Next header (1 byte)	Type (1 byte)	Encrypted data length (2 bytes)
Serial number (Time stamp 4 bytes)		

FIG. 4

Hardware type (2 bytes)	Protocol (2 bytes)	Hardware address size (2 bytes)	Address size (2 bytes)	Request type (2 bytes)
Source MAC address (6 bytes)				
NLP source address (32 bytes)				
Destination MAC address (6 bytes)				
NLP destination address (32 bytes)				
Time stamp+Signature (4+64 bytes)				

FIG. 5

14 bytes	72 bytes	104 bytes
Ethernet Header	NLP basic header	NLPKey extension header

FIG. 6

14 bytes	72 bytes	8 bytes	Variable length
Ethernet Header	NLP basic header	NLPSec extension header	Encrypted data

FIG. 7

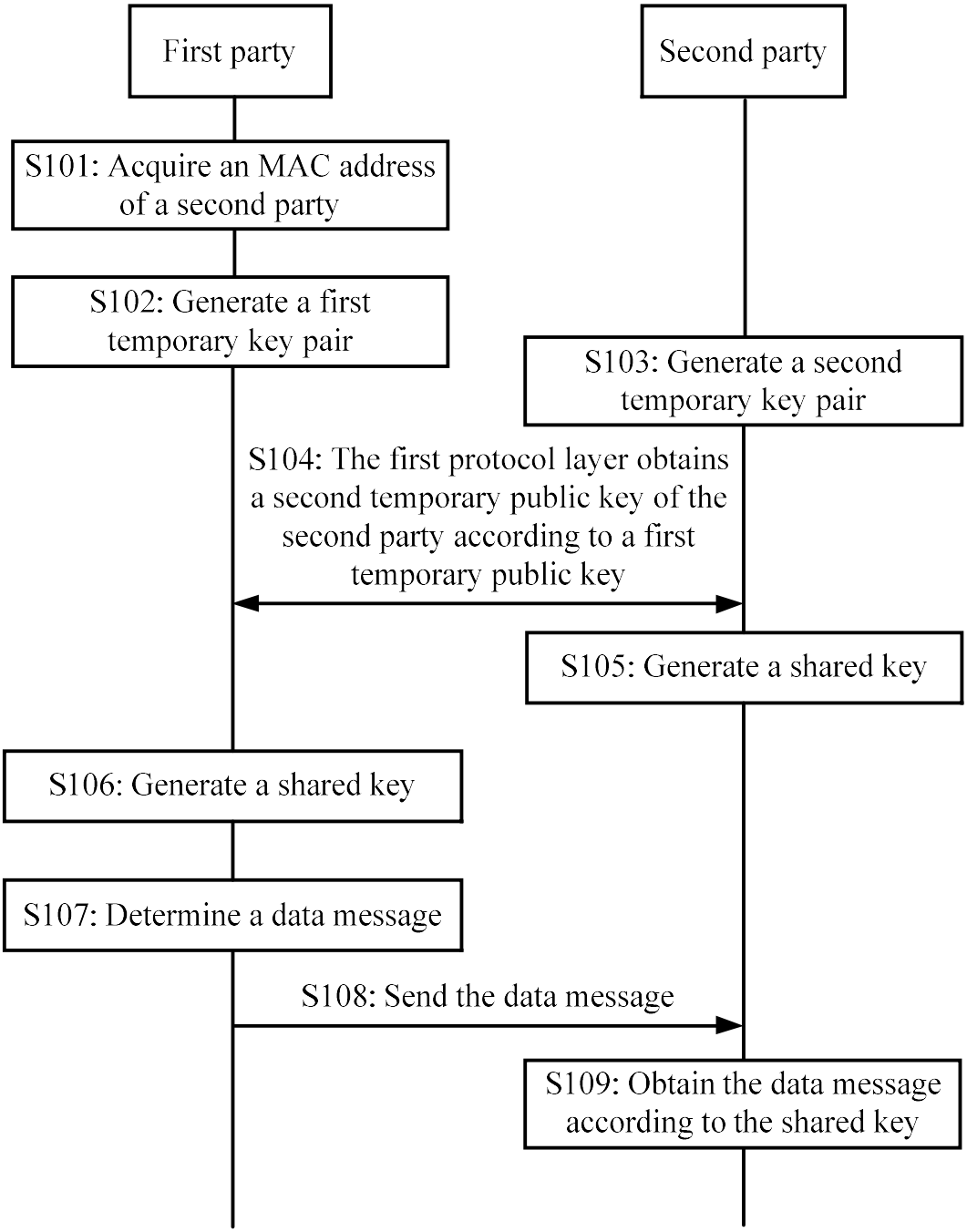

First party
Second party
S101: Acquire an MAC address of a second party
S102: Generate a first temporary key pair
S103: Generate a second temporary key pair
S104: The first protocol layer obtains a second temporary public key of the second party according to a first temporary public key
S105: Generate a shared key
S106: Generate a shared key
S107: Determine a data message
S108: Send the data message
S109: Obtain the data message according to the shared key

FIG. 8A

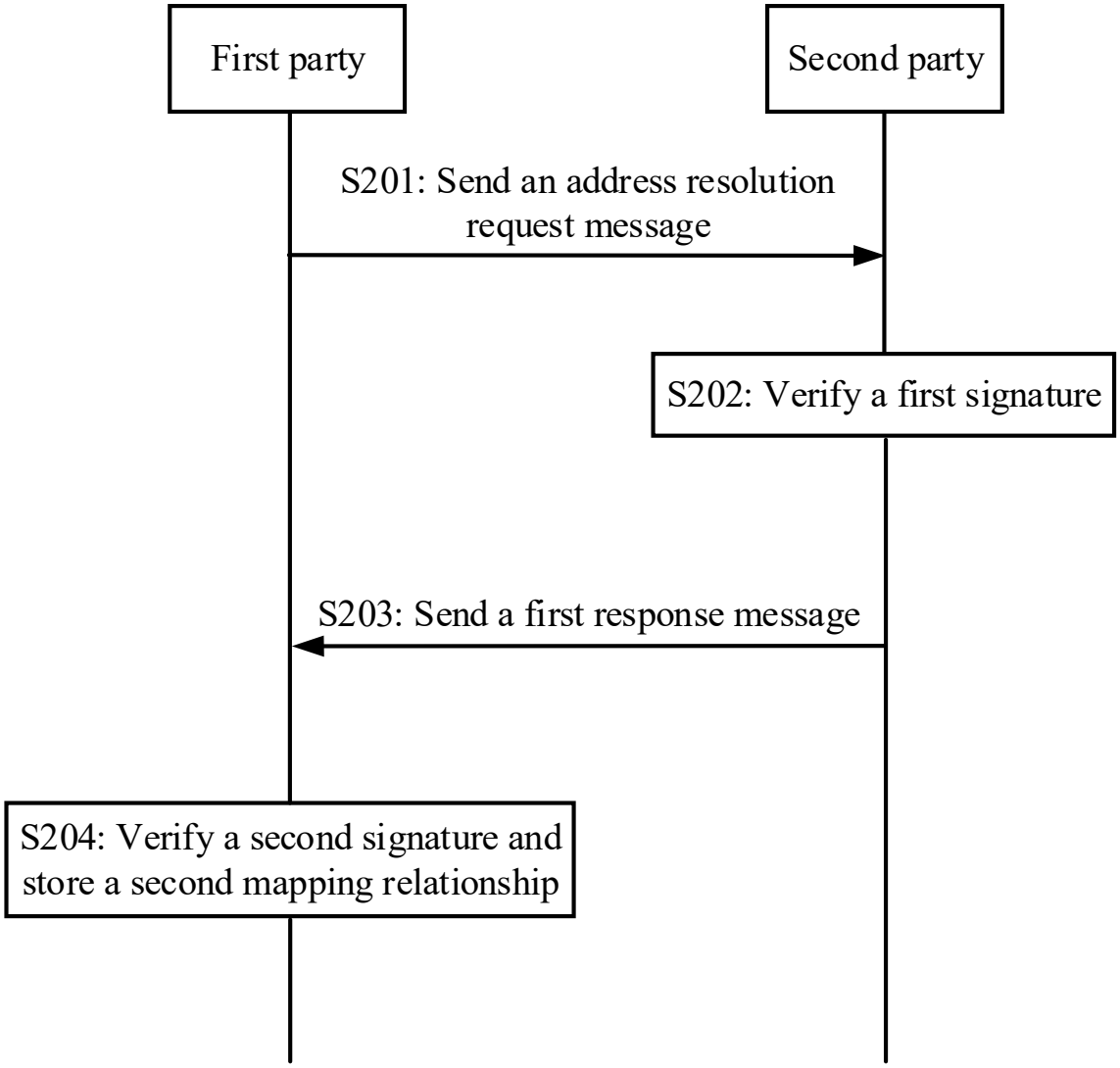

First party
Second party
S201: Send an address resolution request message
S202: Verify a first signature
S203: Send a first response message
S204: Verify a second signature and store a second mapping relationship

FIG. 8B

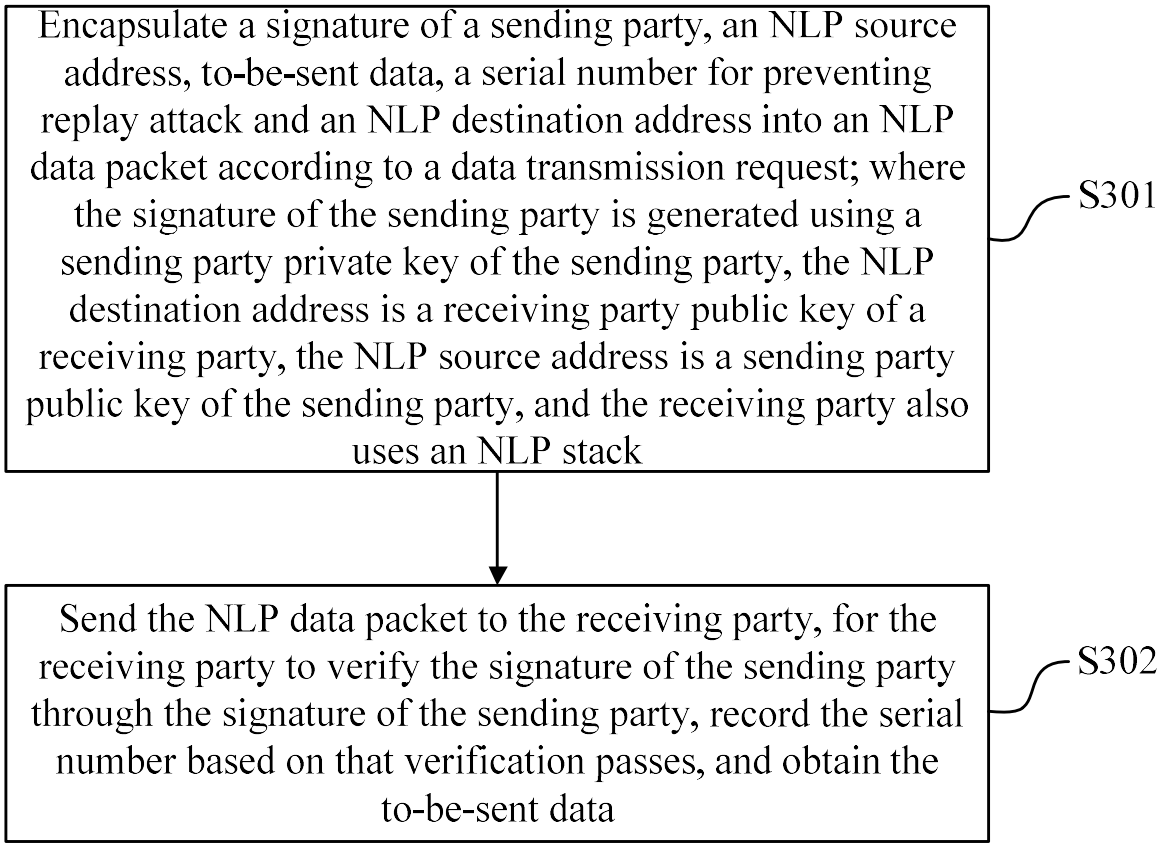

Encapsulate a signature of a sending party, an NLP source address, to-be-sent data, a serial number for preventing replay attack and an NLP destination address into an NLP data packet according to a data transmission request; where the signature of the sending party is generated using a sending party private key of the sending party, the NLP destination address is a receiving party public key of a receiving party, the NLP source address is a sending party public key of the sending party, and the receiving party also uses an NLP stack
S301
Send the NLP data packet to the receiving party, for the receiving party to verify the signature of the sending party through the signature of the sending party, record the serial number based on that verification passes, and obtain the to-be-sent data
S302

FIG. 8C

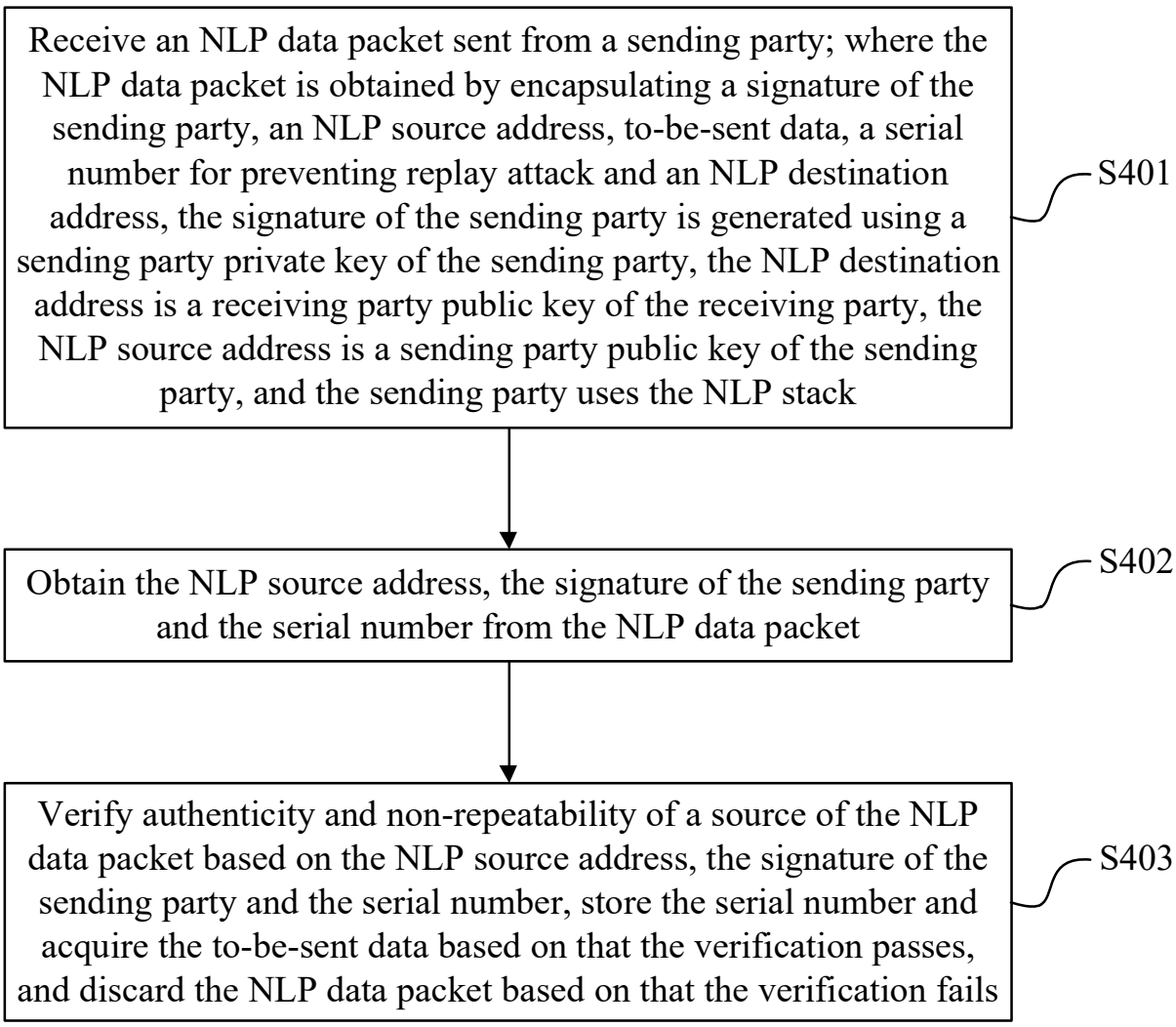

Receive an NLP data packet sent from a sending party; where the NLP data packet is obtained by encapsulating a signature of the sending party, an NLP source address, to-be-sent data, a serial number for preventing replay attack and an NLP destination address, the signature of the sending party is generated using a sending party private key of the sending party, the NLP destination address is a receiving party public key of the receiving party, the NLP source address is a sending party public key of the sending party, and the sending party uses the NLP stack
S401
Obtain the NLP source address, the signature of the sending party and the serial number from the NLP data packet
S402
Verify authenticity and non-repeatability of a source of the NLP data packet based on the NLP source address, the signature of the sending party and the serial number, store the serial number and acquire the to-be-sent data based on that the verification passes, and discard the NLP data packet based on that the verification fails
S403

FIG. 8D

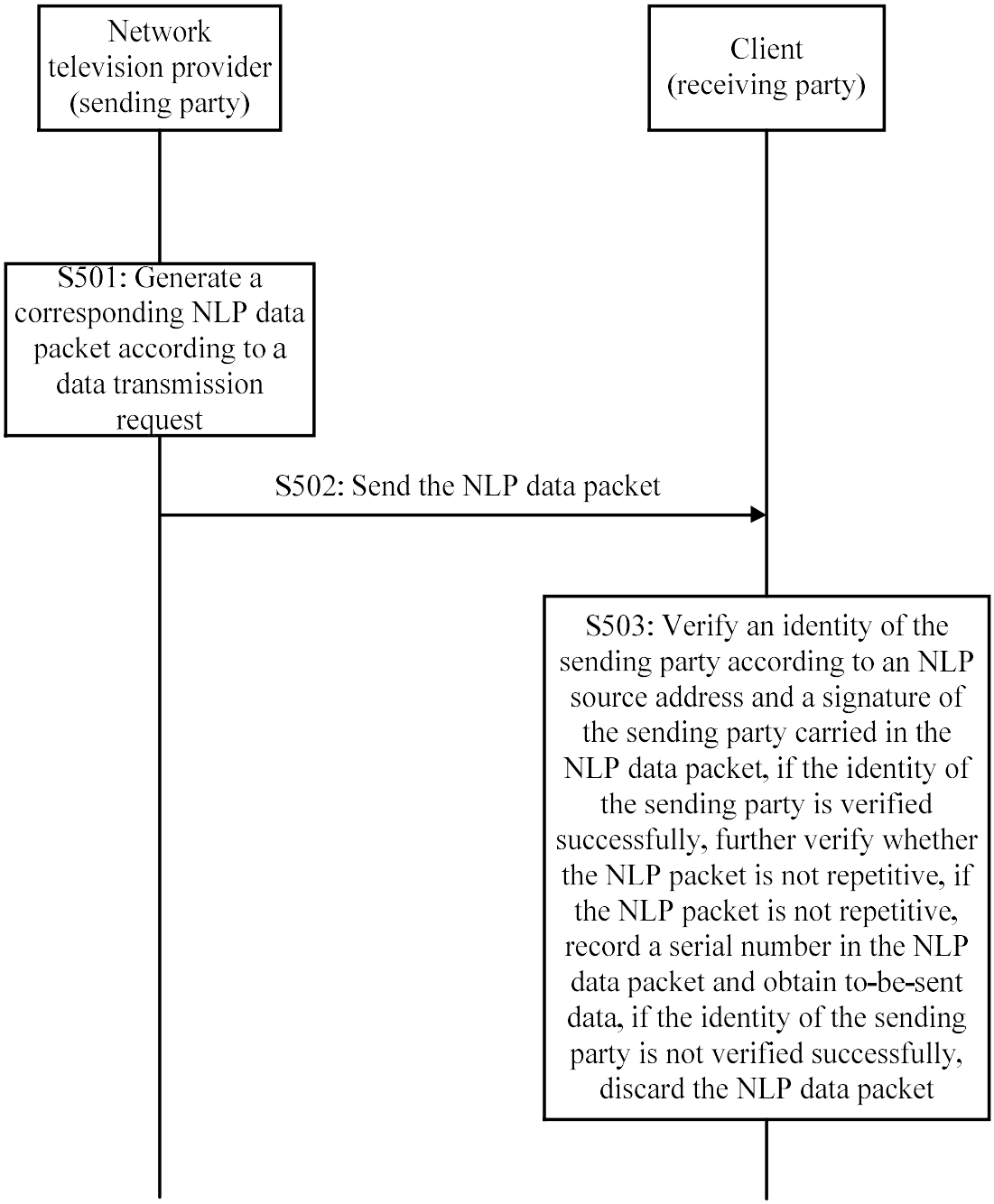

Network television provider (sending party)
Client (receiving party)
S501: Generate a corresponding NLP data packet according to a data transmission request
S502: Send the NLP data packet
S503: Verify an identity of the sending party according to an NLP source address and a signature of the sending party carried in the NLP data packet, if the identity of the sending party is verified successfully, further verify whether the NLP packet is not repetitive, if the NLP packet is not repetitive, record a serial number in the NLP data packet and obtain to-be-sent data, if the identity of the sending party is not verified successfully, discard the NLP data packet

FIG. 8E

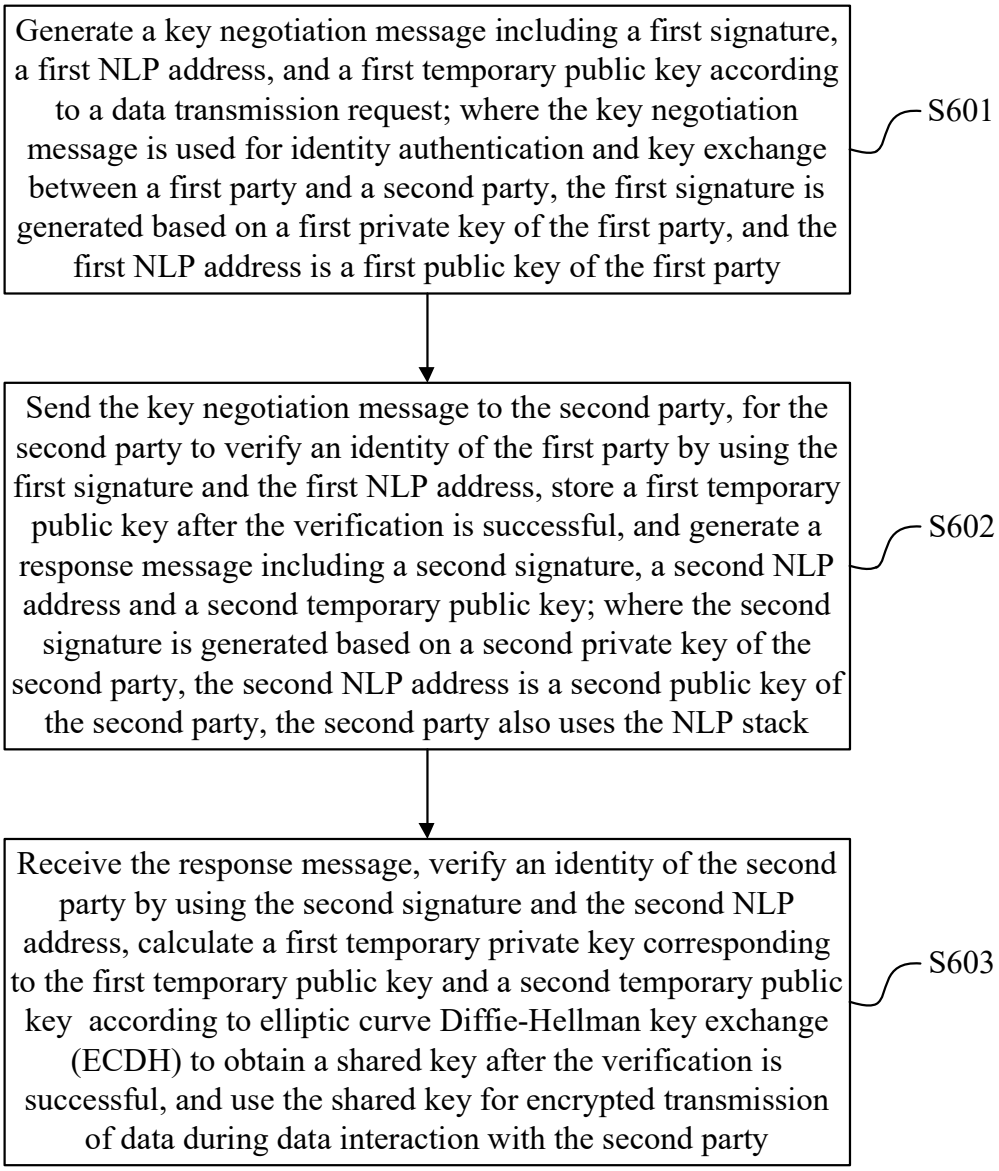

Generate a key negotiation message including a first signature, a first NLP address, and a first temporary public key according to a data transmission request; where the key negotiation message is used for identity authentication and key exchange between a first party and a second party, the first signature is generated based on a first private key of the first party, and the first NLP address is a first public key of the first party
S601
Send the key negotiation message to the second party, for the second party to verify an identity of the first party by using the first signature and the first NLP address, store a first temporary public key after the verification is successful, and generate a response message including a second signature, a second NLP address and a second temporary public key; where the second signature is generated based on a second private key of the second party, the second NLP address is a second public key of the second party, the second party also uses the NLP stack
S602
Receive the response message, verify an identity of the second party by using the second signature and the second NLP address, calculate a first temporary private key corresponding to the first temporary public key and a second temporary public key according to elliptic curve Diffie-Hellman key exchange (ECDH) to obtain a shared key after the verification is successful, and use the shared key for encrypted transmission of data during data interaction with the second party
S603

FIG. 8F

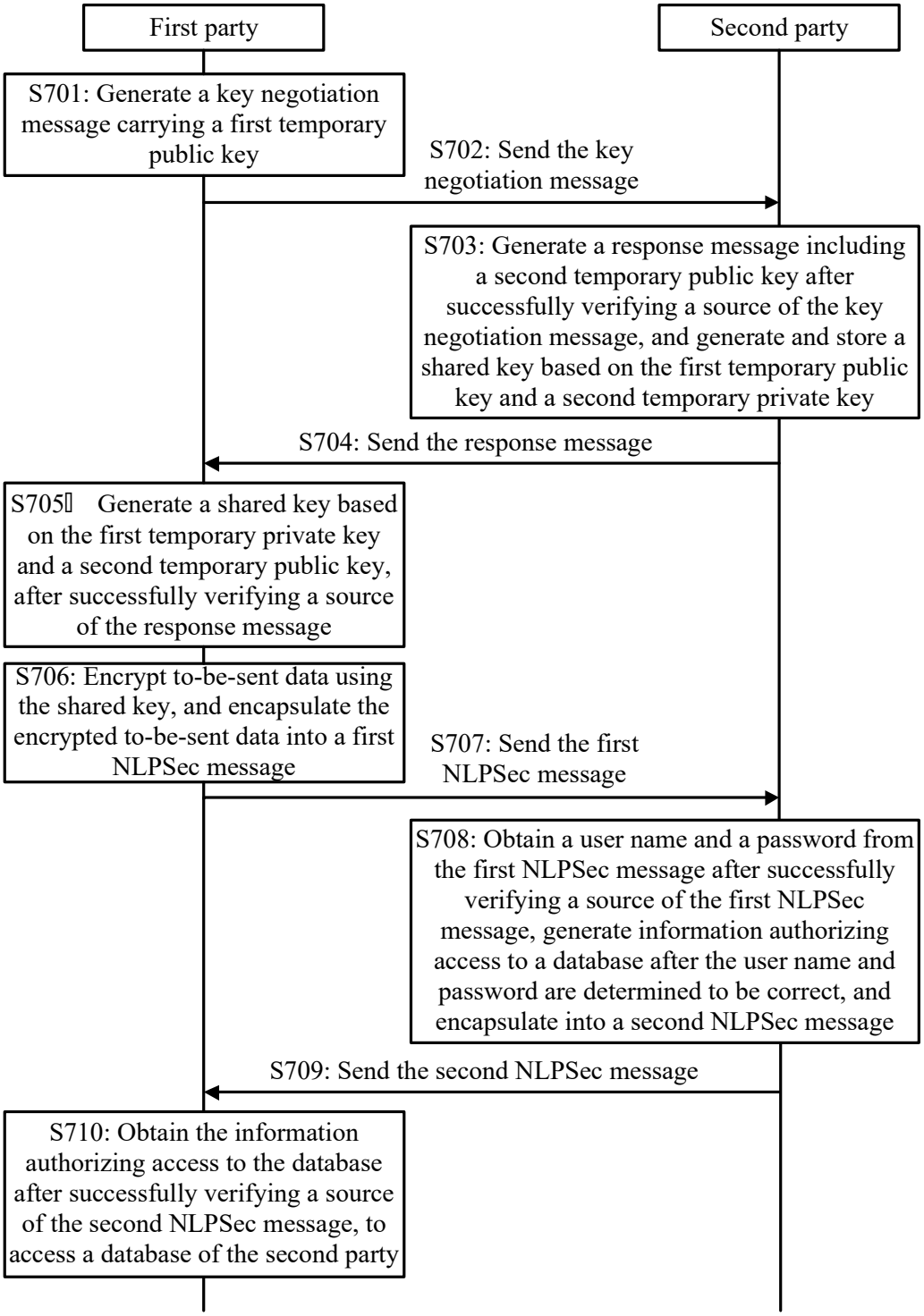

First party
Second party
S701: Generate a key negotiation message carrying a first temporary public key
S702: Send the key negotiation message
S703: Generate a response message including a second temporary public key after successfully verifying a source of the key negotiation message, and generate and store a shared key based on the first temporary public key and a second temporary private key
S704: Send the response message
S705 Generate a shared key based on the first temporary private key and a second temporary public key, after successfully verifying a source of the response message
S706: Encrypt to-be-sent data using the shared key, and encapsulate the encrypted to-be-sent data into a first NLPSec message
S707: Send the first NLPSec message
S708: Obtain a user name and a password from the first NLPSec message after successfully verifying a source of the first NLPSec message, generate information authorizing access to a database after the user name and password are determined to be correct, and encapsulate into a second NLPSec message
S709: Send the second NLPSec message
S710: Obtain the information authorizing access to the database after successfully verifying a source of the second NLPSec message, to access a database of the second party

FIG. 8G

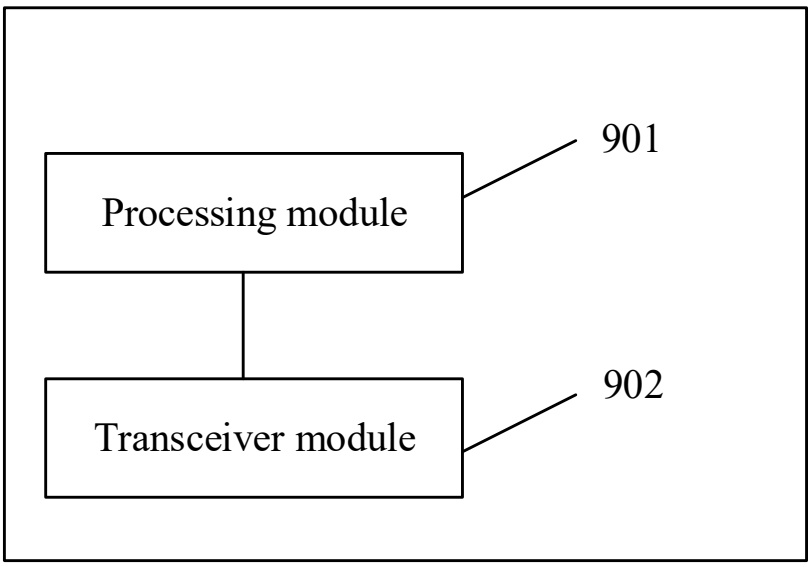

901
Processing module
902
Transceiver module

FIG. 9A

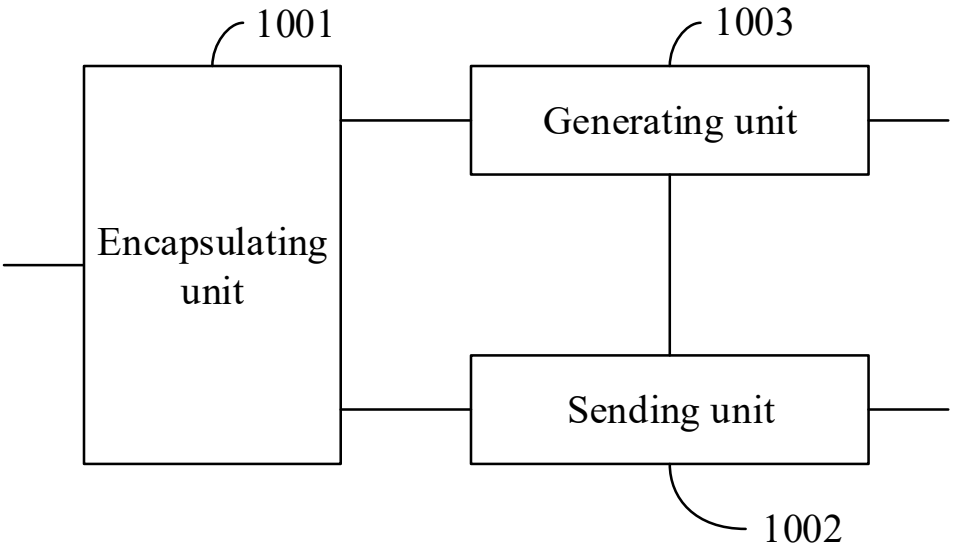

1001
1003
Generating unit
Encapsulating unit
Sending unit
1002

FIG. 9B

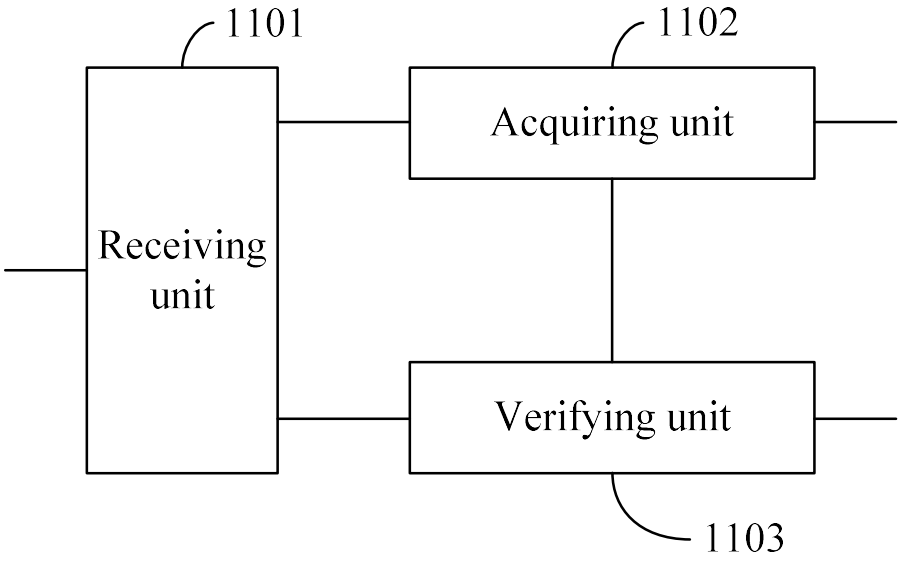

1101
1102
Receiving unit
Acquiring unit
Verifying unit
1103

FIG. 9C

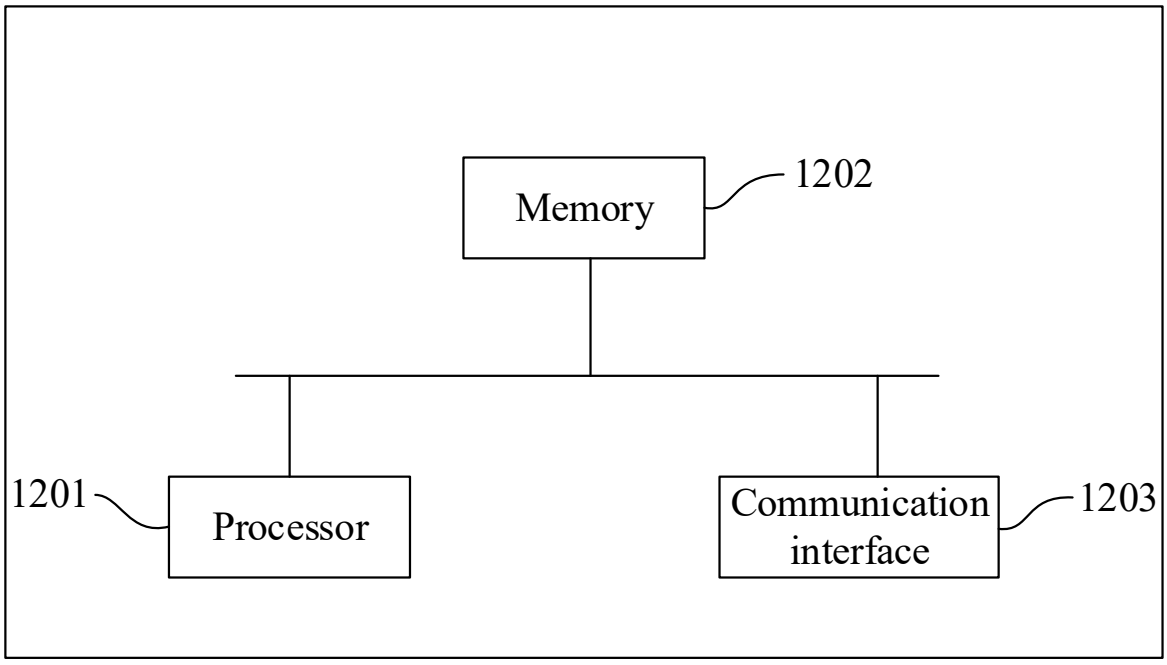

Memory
1202
1201
Processor
Communication interface
1203

FIG. 10

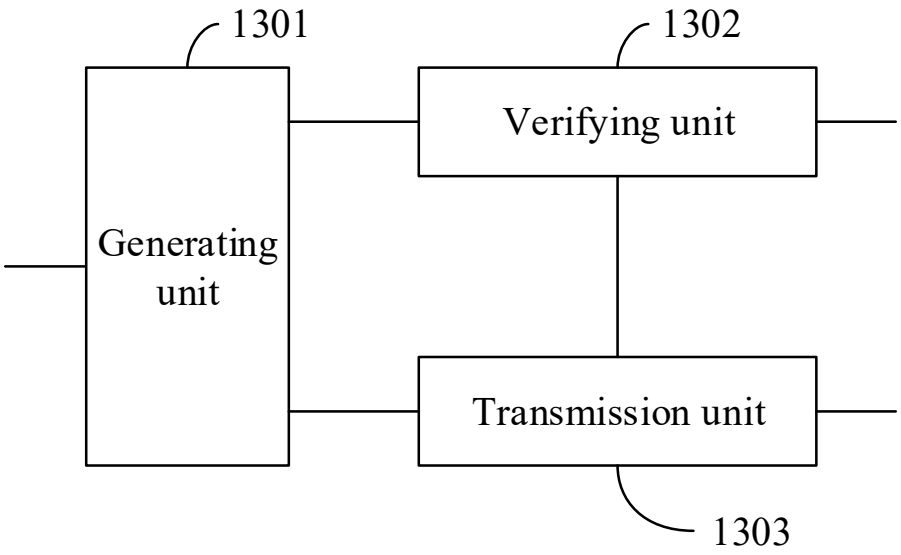

1301
1302
Verifying unit
Generating unit
Transmission unit
1303

FIG. 11