FISF130020: Introduction to Computer Science

Lecture 10. PKI and Applications

Hui Xu

xuh@fudan.edu.cn

Recap

- We have learned several different cryptography approaches
  - Asymmetric encryption: RSA
  - Symmetric encryption
  - Hash
- Question: How to use them in practice?
  - Key distribution and management
  - Usage of cryptography algorithms
- In this lecture, we focus on the centralized trust model
  - *i.e.,* there is an authority that all participants can trust

Practice 1

- Given p = 3, q = 11, e = 7,
  1) calculate the private key
  2) use the private key to encrypt M=5
  3) decrypt the previous encrypted data with the public key

Practice 2

- Analyze the security of the following hash function where M = a_1,…, a_n

$$Hash(M) = \left(\sum_{i=1}^{t} a_i^2\right) \bmod n$$

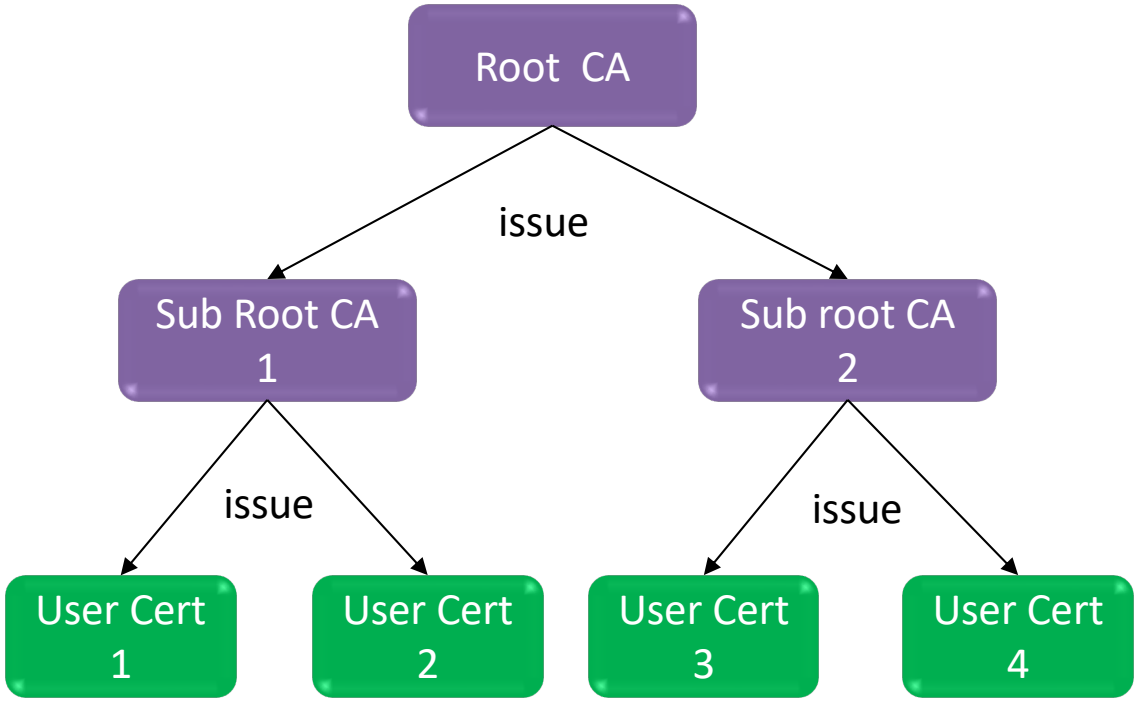
Outline

- 1. Public Key Infrastructure
- 2. Digital Signature and Envelope
- 3. Transport Layer Security
- 4. In-class Practice

1. Public Key Infrastructure

Public Key Infrastructure

- A centralized trust framework for digital certificate management
- Certificate Authority (CA): Issues and verifies digital certificates.
- Digital Certificates: Certifies ownership of a public key, linking it to an entity.

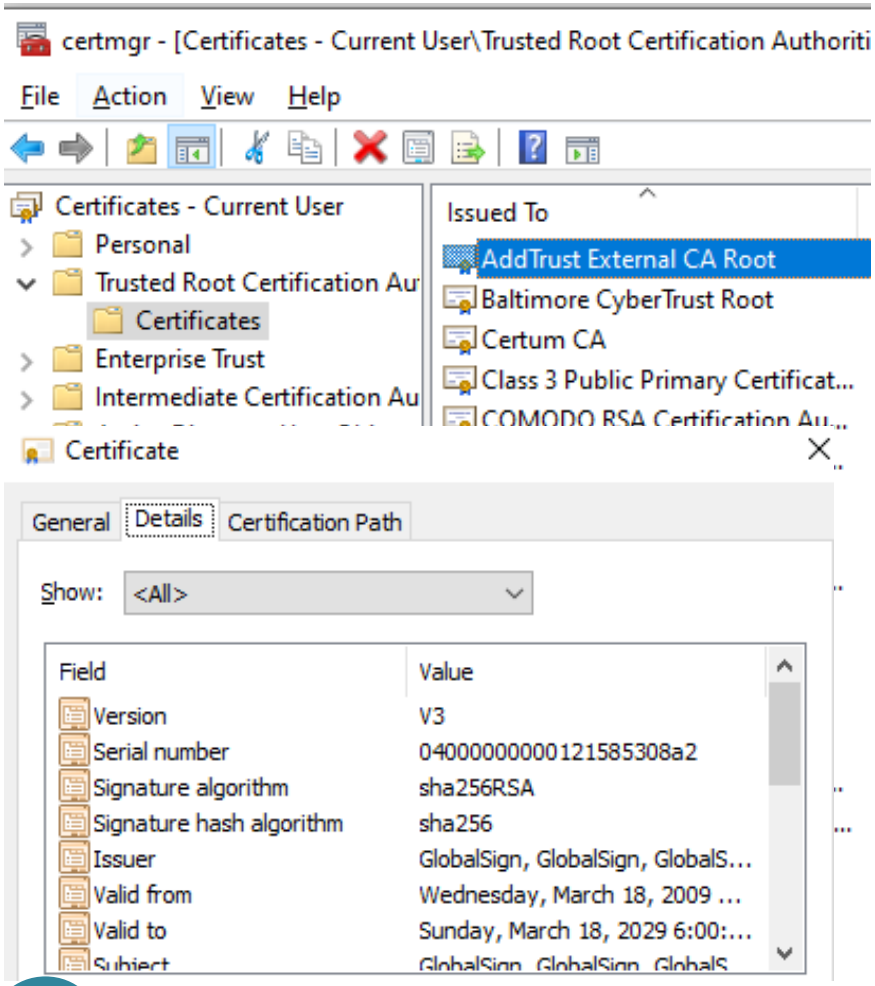


Essential Fields of a Digital Cert

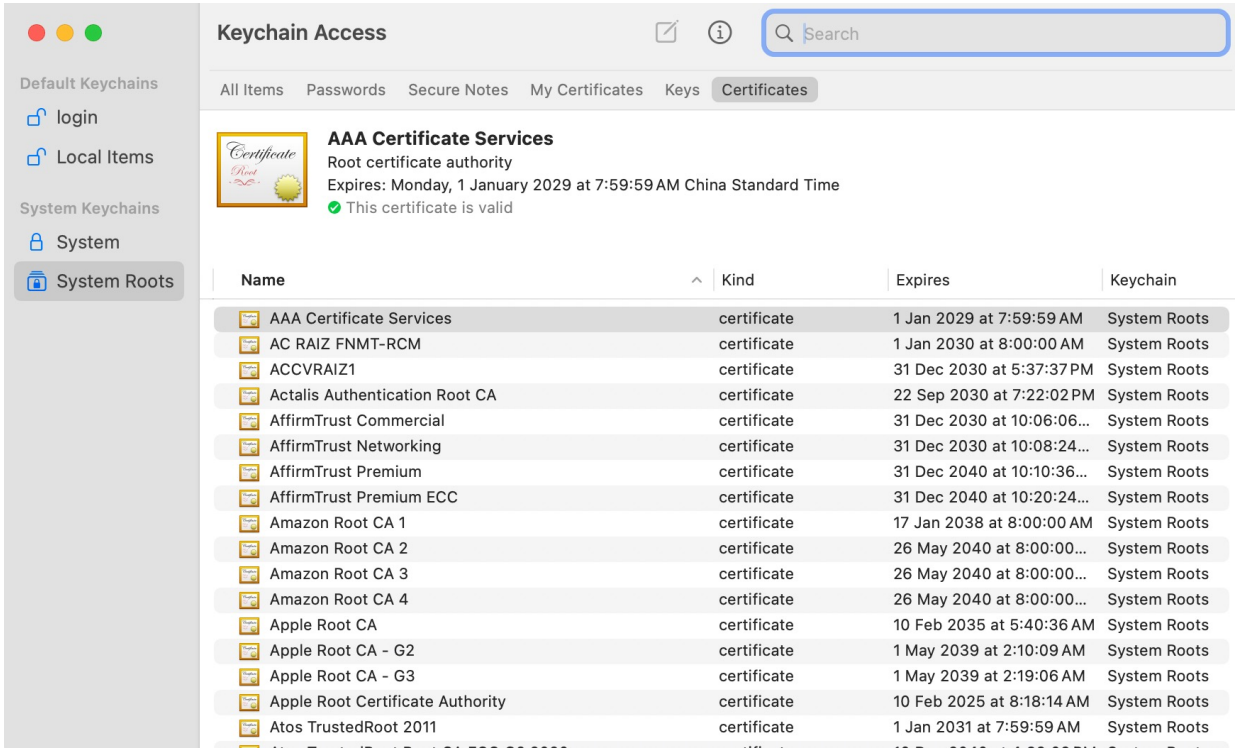
- **Version**: X.509 standard being used (commonly v3).
- **Serial Number**: a unique identifier assigned by the CA to distinguish each certificate it issues.
- **Subject**: The entity the certificate is issued to.
- **Issuer**: Identifies the CA that issued the certificate.
- **Validity Period**: The time frame for which the certificate is valid.
- **Signature**: Generated by the CA for authenticity check.

View the Certs on your PC

Win + R: certmgr.msc



MacOS: Keychain Access



Demonstration of Cert Issuing: User Cert

User key generation (assume using RSA as the public key algorithm):

p=5, q=11, e=3

=> public key: (3, 55), private key: 27

User cert issuing:

Version: x509.v3

Serial Number: 100202

Subject: CN = Alice, O = Fudan University, C = CN

Issuer: CN = RootCA, O = Fudan University, C = CN

Validity: 2024 - 0901 to 2027-07-31

Public Key: (3, 55)

Sign(Hash(Ver || SN || Subject || Issuer || Validity || PubKey), $PriKey_{rootCA}$)

Demonstration of Cert Issuing: Root Cert

Root key generation (assume using RSA as the public key algorithm):

p=3, q=11, e=7

=> public key: (7, 33), private key: 3

Root cert issuing (who signs the cert of the root CA):

Version: x509.v3

Serial Number: 100001

Subject: CN = RootCA, O = Fudan University, C = CN

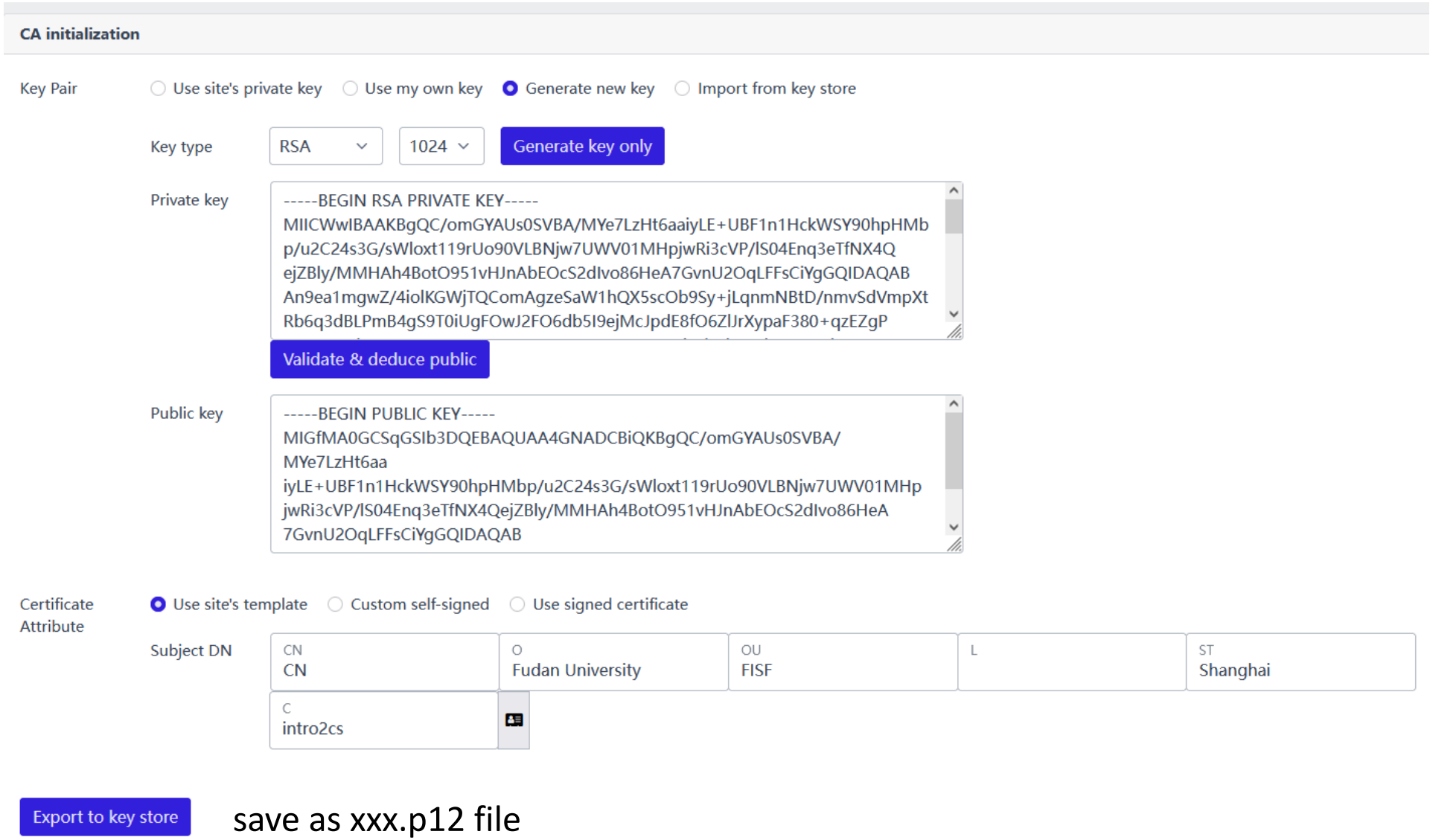
Issuer: CN = RootCA, O = Fudan University, C = CN

Validity: 2000 - 0101 to 2099-12-31

Public Key: (7, 33)

Sign(Hash(Ver || SN || Subject || Issuer || Validity || PubKey), $PriKey_{rootCA}$)

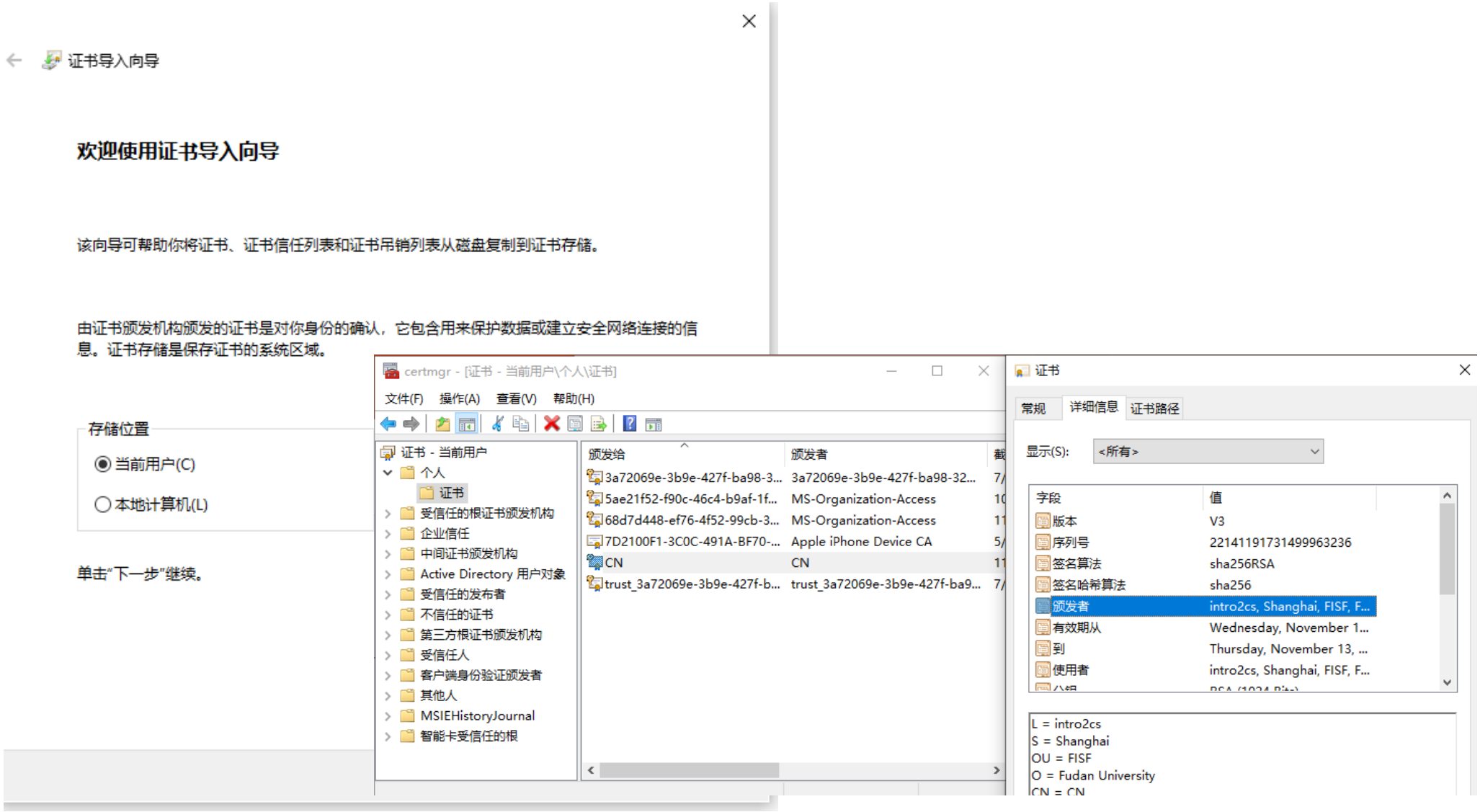
Experiment: Generate Your Own Cert



save as xxx.p12 file

https://pkitools.net/pages/ca/ca.html

Experiment: Install The Cert

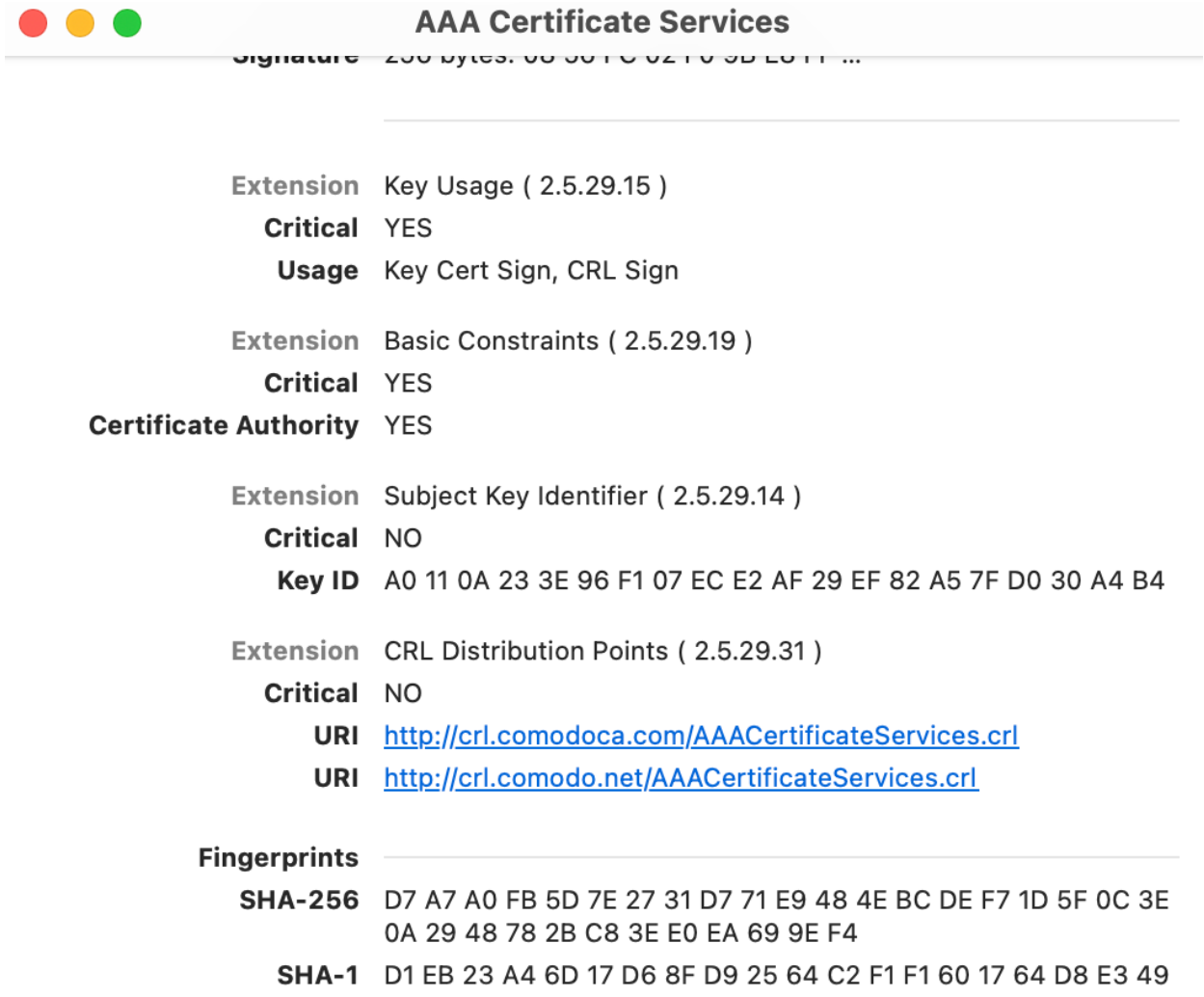


Private Key Protection

- Software approach: Encrypt the private key with a password.
- Security chips:
  - Generate key pairs or import key pairs.
  - Perform encryption and decryption operations.
  - Protect the key from being exported.

Cert Revocation

- Publish to a cert revocation list signed by the Root CA



2. Digital Signature and Envelope

Security Goal: CIAN

- **Confidentiality (C)**: Ensuring that sensitive information is only accessible to authorized individuals or systems.
- **Integrity (I)**: Guaranteeing that data remains accurate, unaltered, and trustworthy throughout its lifecycle
- **Availability (A)**: Ensuring that information and resources are available to authorized users when needed.
- **Non-repudiation (N)**: Providing proof of the origin and integrity of data, ensuring that actions or communications cannot later be denied by the entity that performed them.

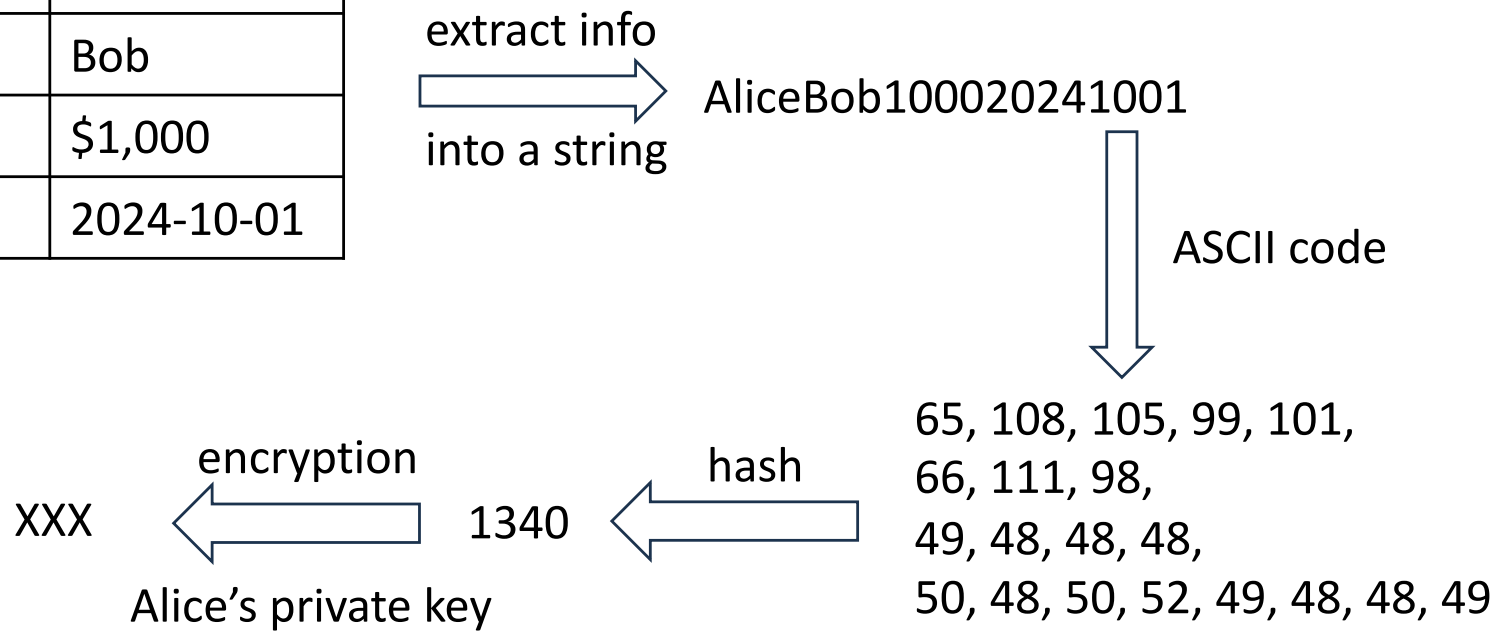
Digital Signature

- A digital signature is a cryptographic method used to validate the authenticity and integrity of a digital message, software, or document.
- **Key Characteristics:**
  - Authentication: Confirms the identity of the sender.
  - Integrity: Ensures the message hasn’t been tampered with.
  - Non-Repudiation: Prevents the sender from denying they sent the message.
- **Applications**: Used widely in e-signatures, financial transactions, contract management, and secure email communication.

How It Works? Sign

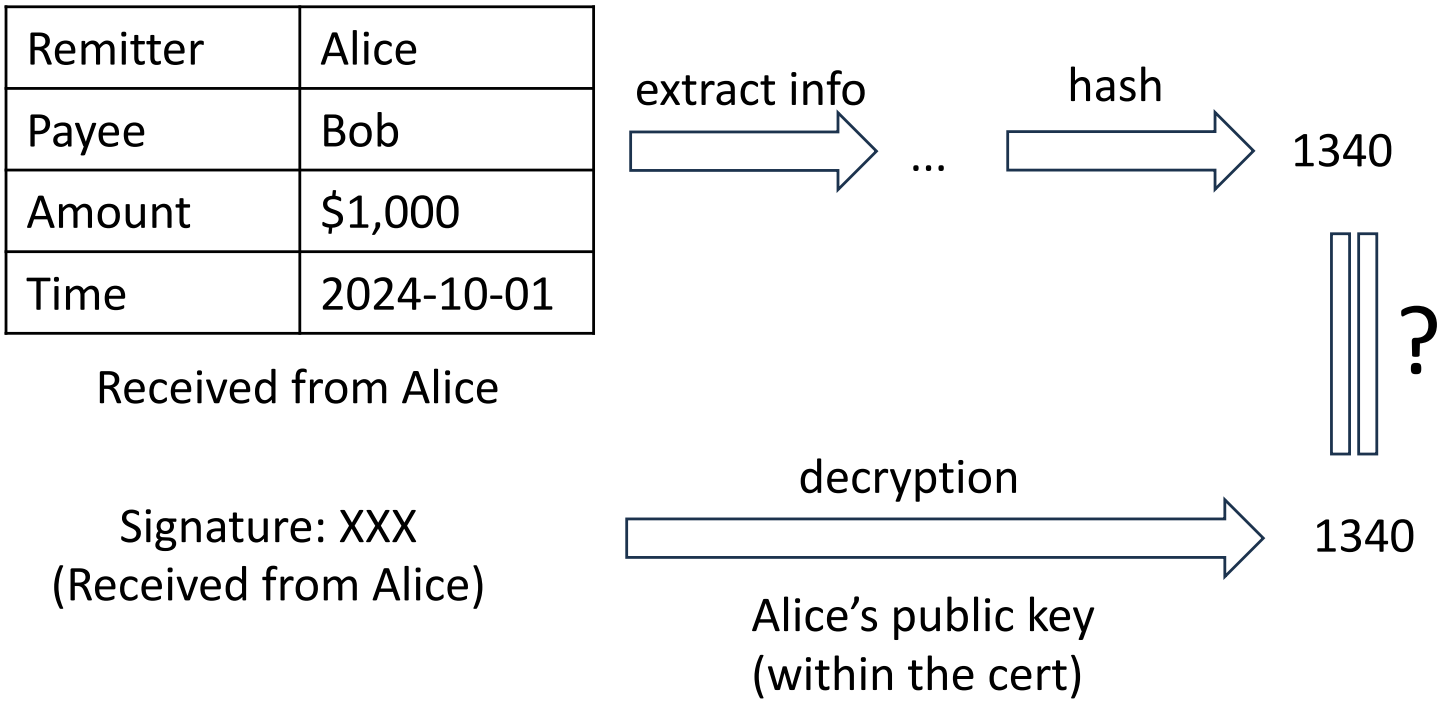
- **Hashing:** The document is hashed into a fixed-length string.
- **Encryption:** The hash is encrypted with the sender's private key.

Remitter	Alice
Payee	Bob
Amount	$1,000
Time	2024-10-01



How It Works? Verification

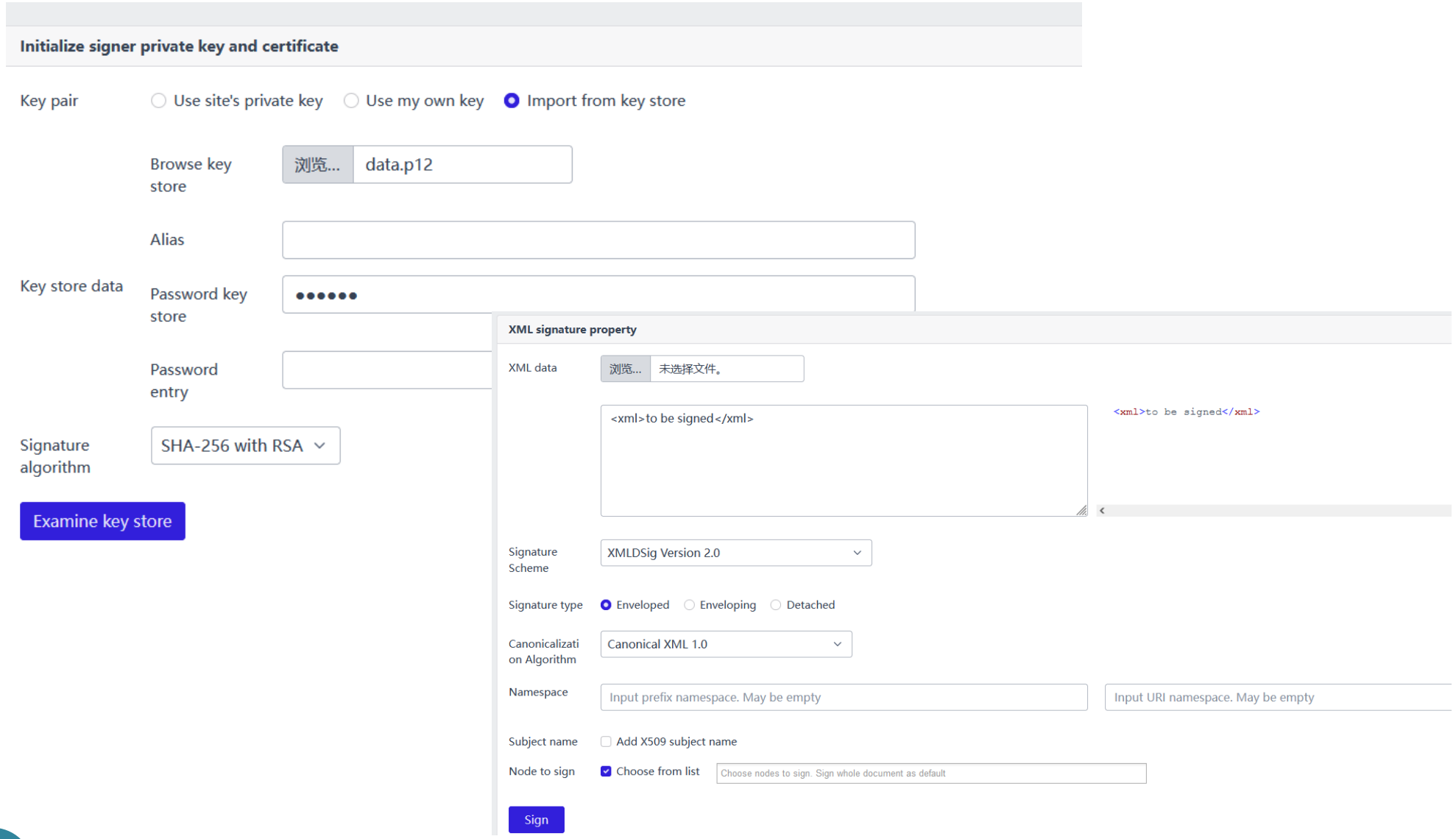
- The recipient uses the sender's public key to verify the signature.
- The recipient uses the root certificate to verify the sender's certificate.



Question

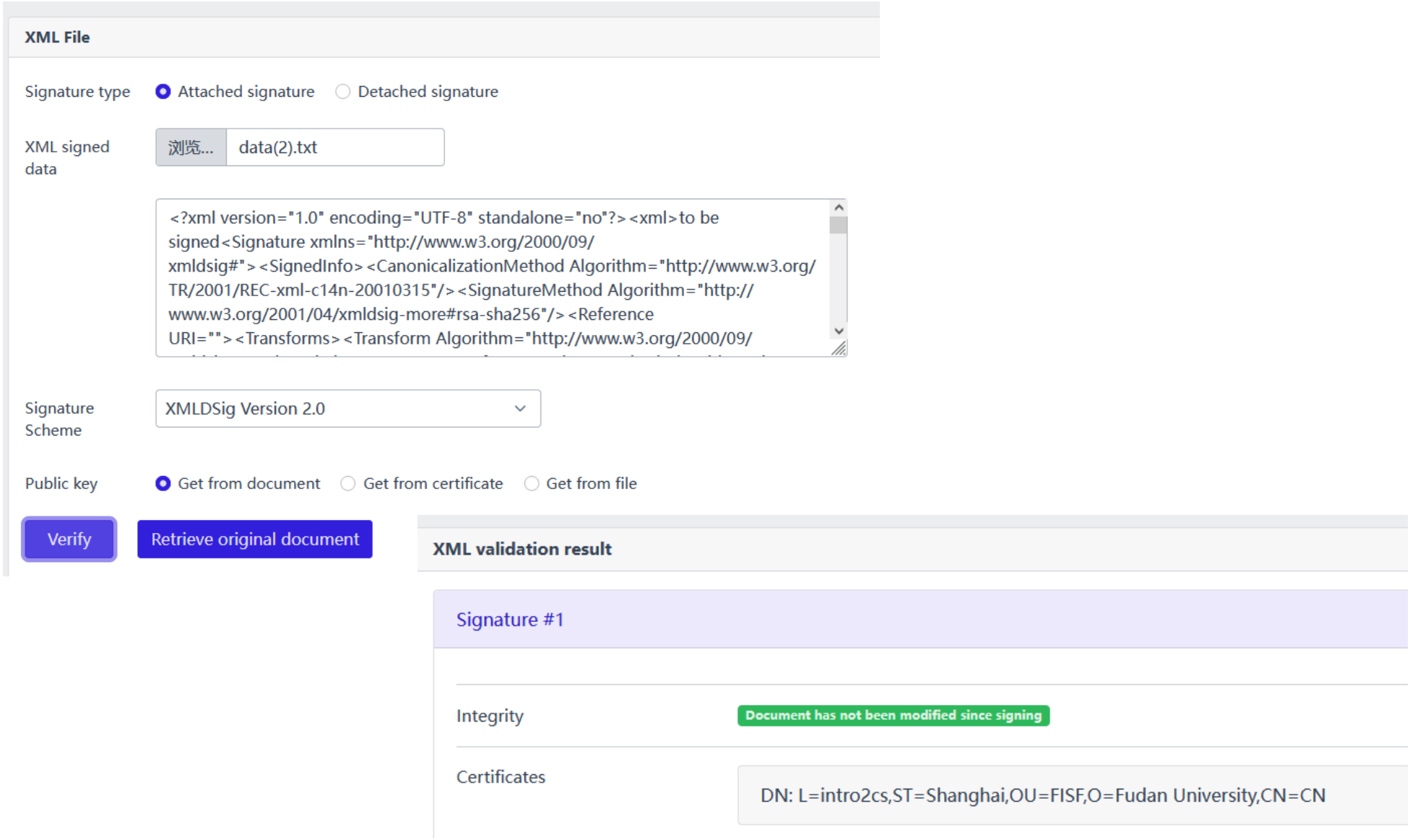
- What is the benefit of hashing before RSA for digital signature? Is it necessary?
- Can you design a login mechanism based on digital signature?

Experiment: Sign an XML file



https://pkitools.net/pages/sign/xml.html

Experiment: Verify the Signed XML file



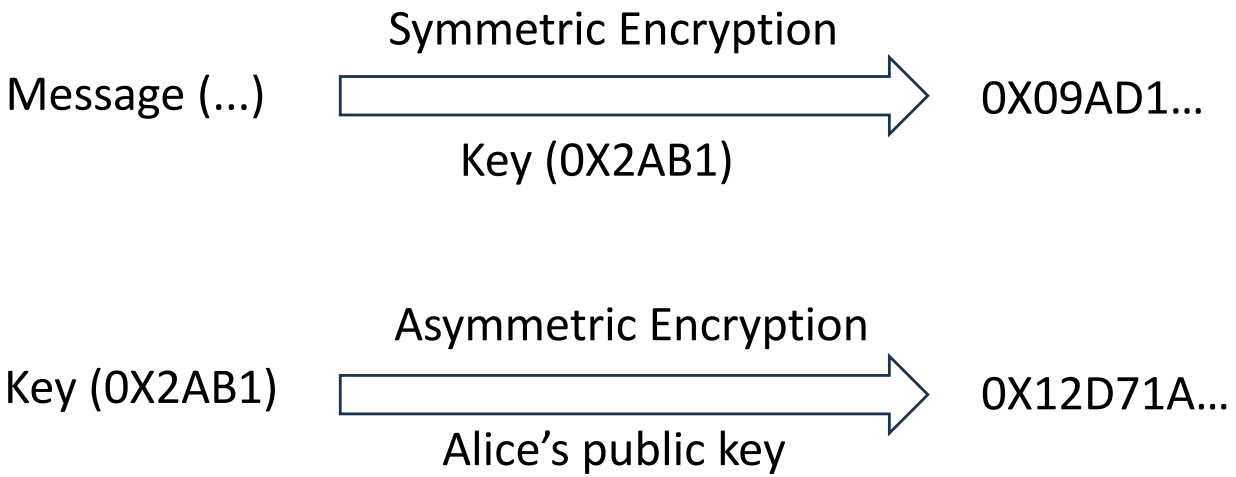
https://pkitools.net/pages/sign/xml.html

Digital Envelope

- A digital envelope is a secure method for transmitting sensitive data by combining encryption techniques to ensure both confidentiality and integrity of the message.
- **Benefits:**
  - Confidentiality: Ensures only the intended recipient can read the message.
  - Efficiency: Combines fast symmetric encryption with secure asymmetric encryption.
  - Scalability: Effective for transmitting large files securely.
- **Applications:** Used in email encryption, secure data exchange, and digital communication protocols.

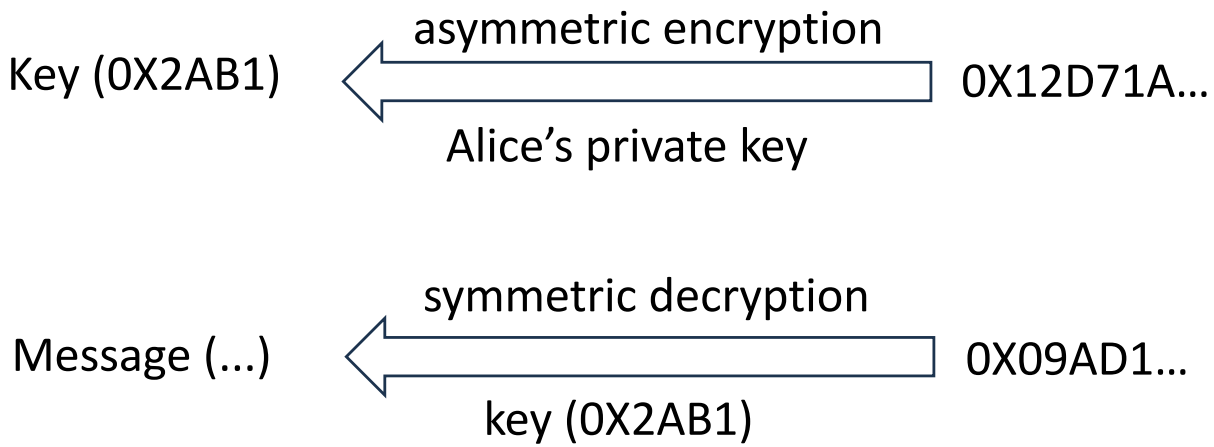
How It Works? Encryption

- **Symmetric Key Encryption:** The message is encrypted with a randomly generated symmetric key (e.g., AES).
- **Asymmetric Key Encryption:** The symmetric key is then encrypted using the recipient's public key.
- **Transmission:** The encrypted message and the encrypted symmetric key (the "digital envelope") are sent to the recipient.



How It Works? Decryption

- **Decryption by Recipient:** The recipient decrypts the symmetric key using their private key.
- The decrypted symmetric key is used to decrypt the original message.

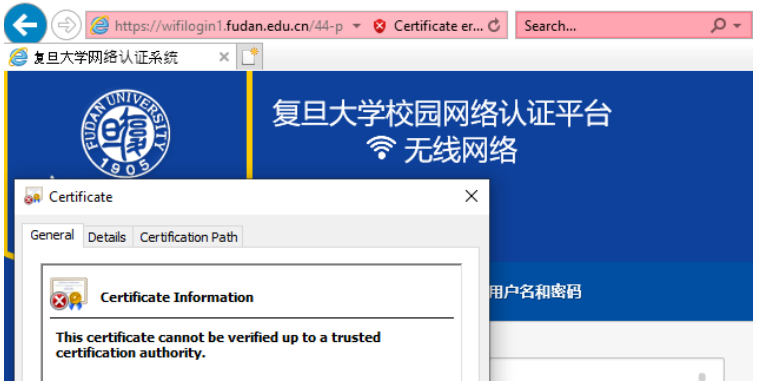


3. Transport Layer Security

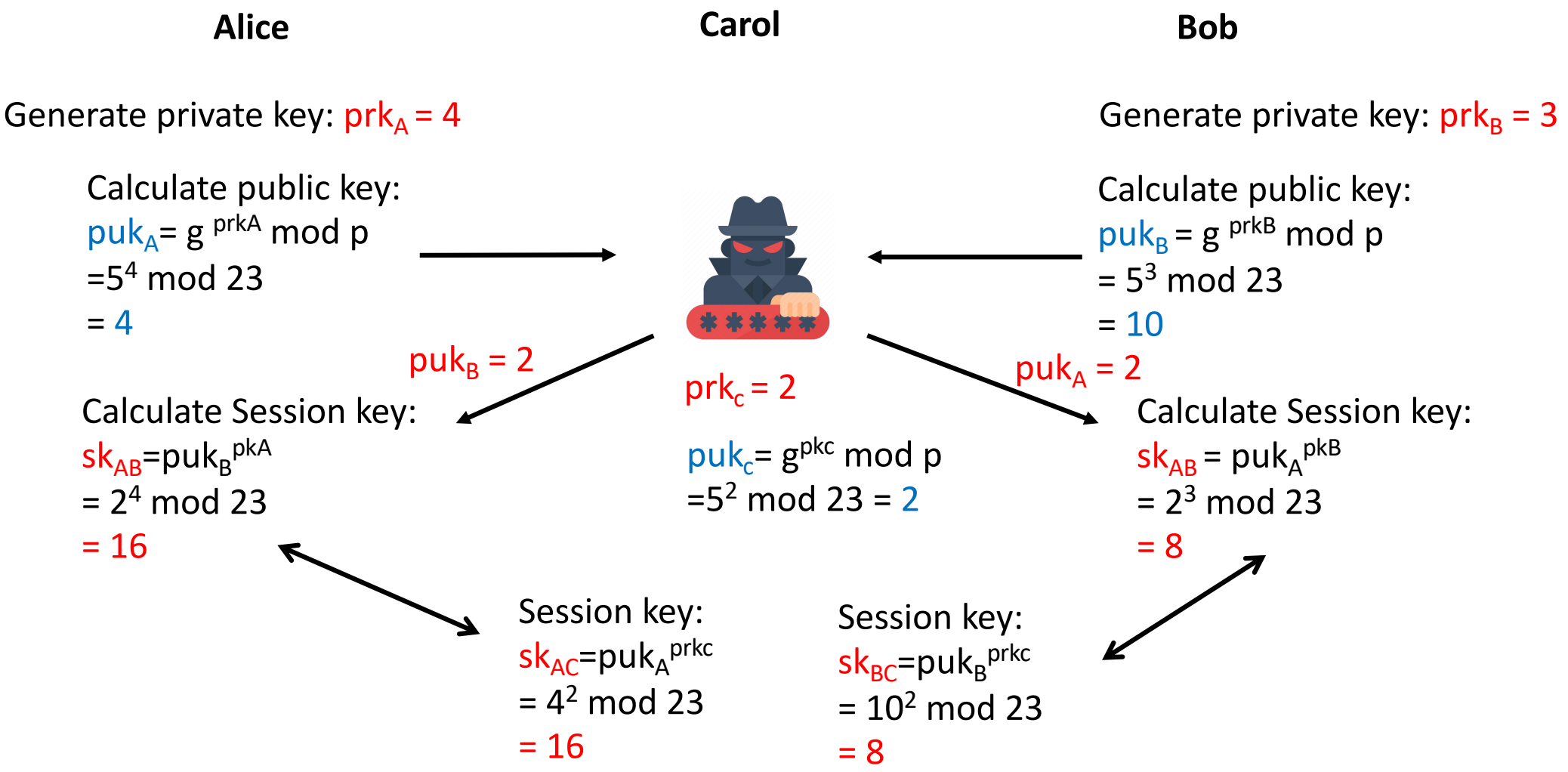
Previously known as SSL (secure socket layer)

TLS/HTTPS

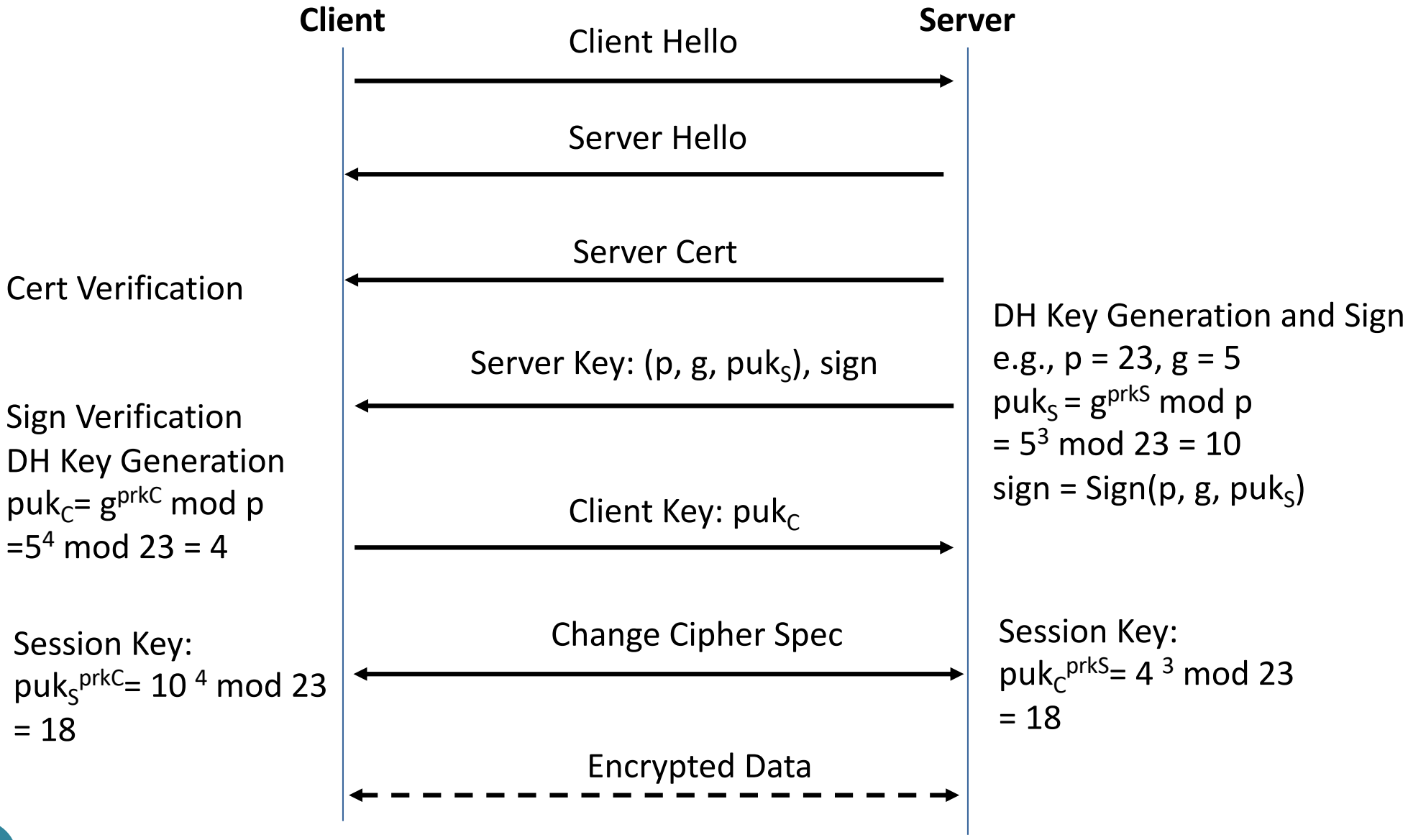
- TLS is a protocol designed to secure data transmitted over networks.
  - Useful for network traffic encryption.
  - Generally based on digital certificates.
- HTTPS: HTTP over TLS
  - The website is configured with digital certificates.
  - Useful to prevent phishing attacks (forgery websites).
    - For example, a fake website that looks exactly the same as the target web page.
    - Attackers can steal your account and password if you enter your credentials.



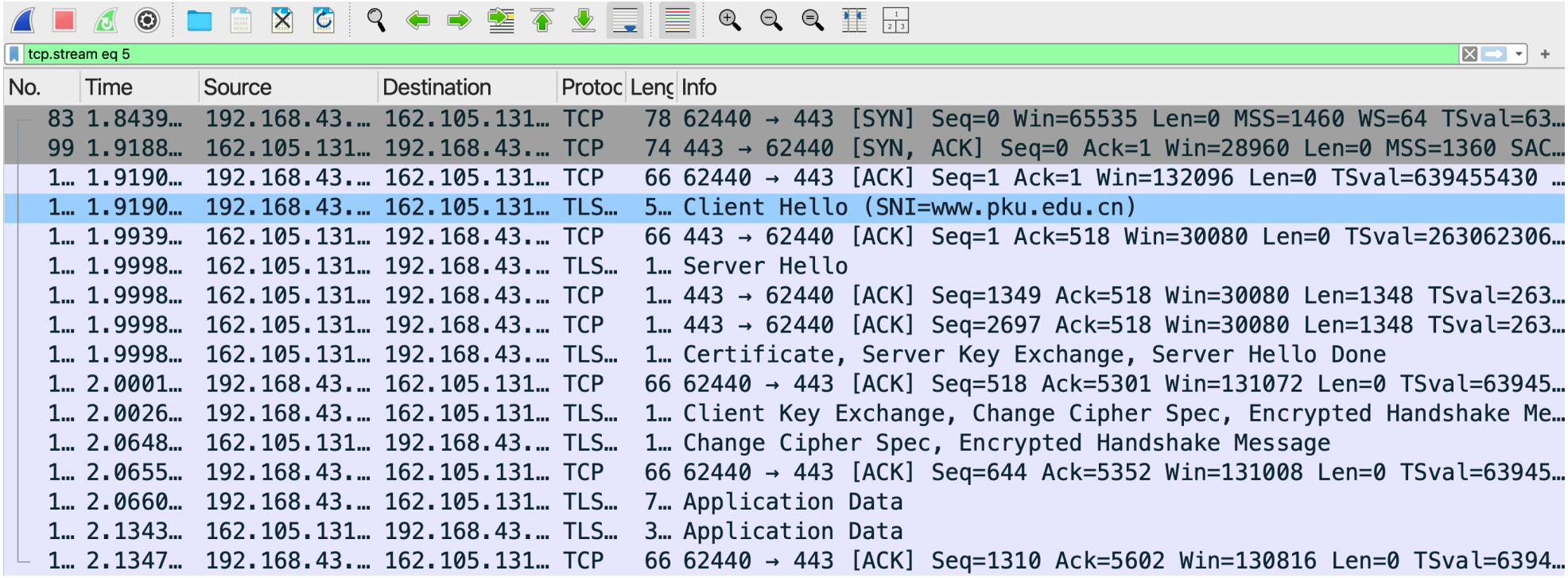
Recap: MITM Attack for Diffie-Hellman Key Exchange



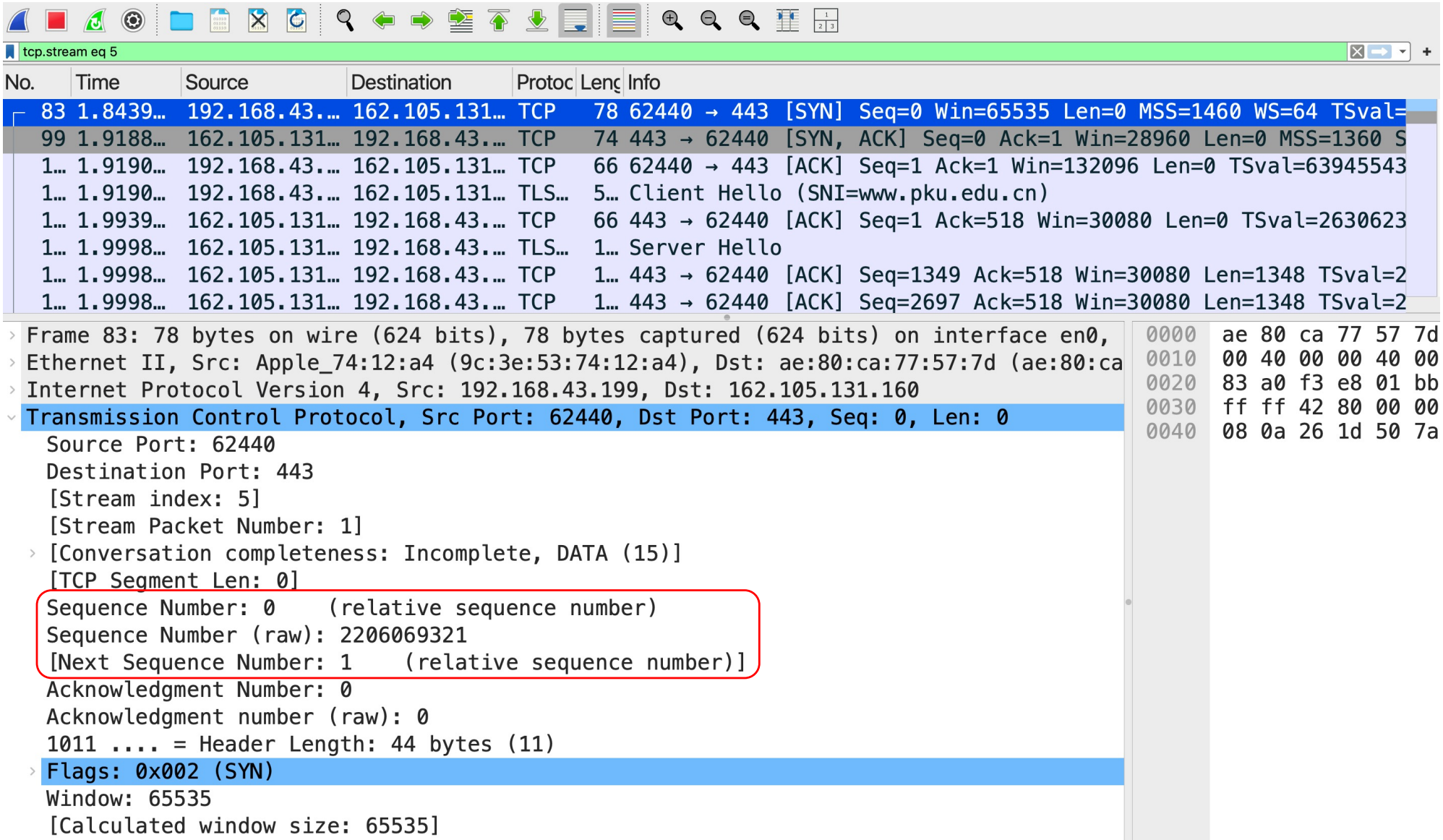
Mechanism (Simplified 1-Way TLS)



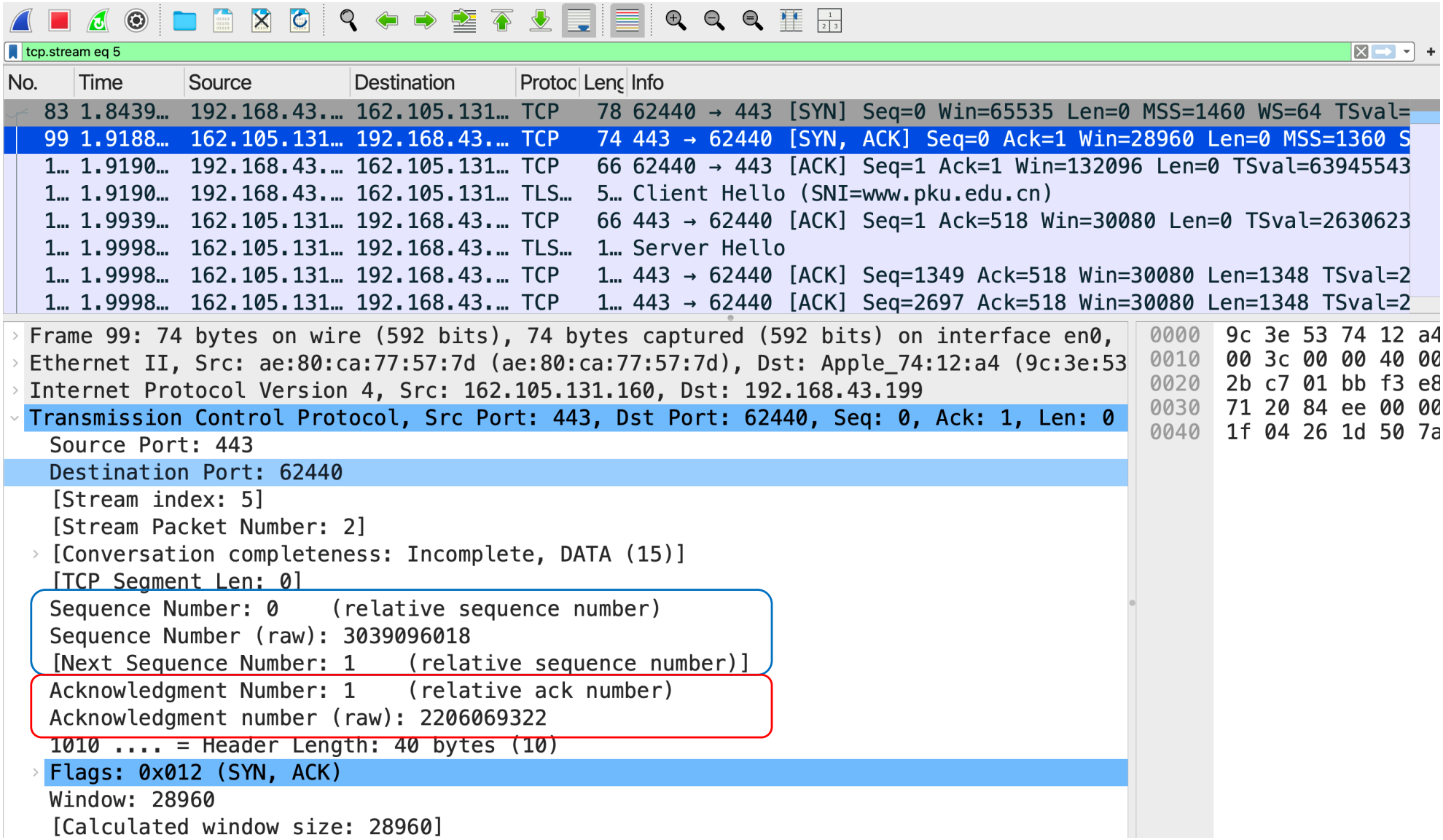
Experiment: TLS



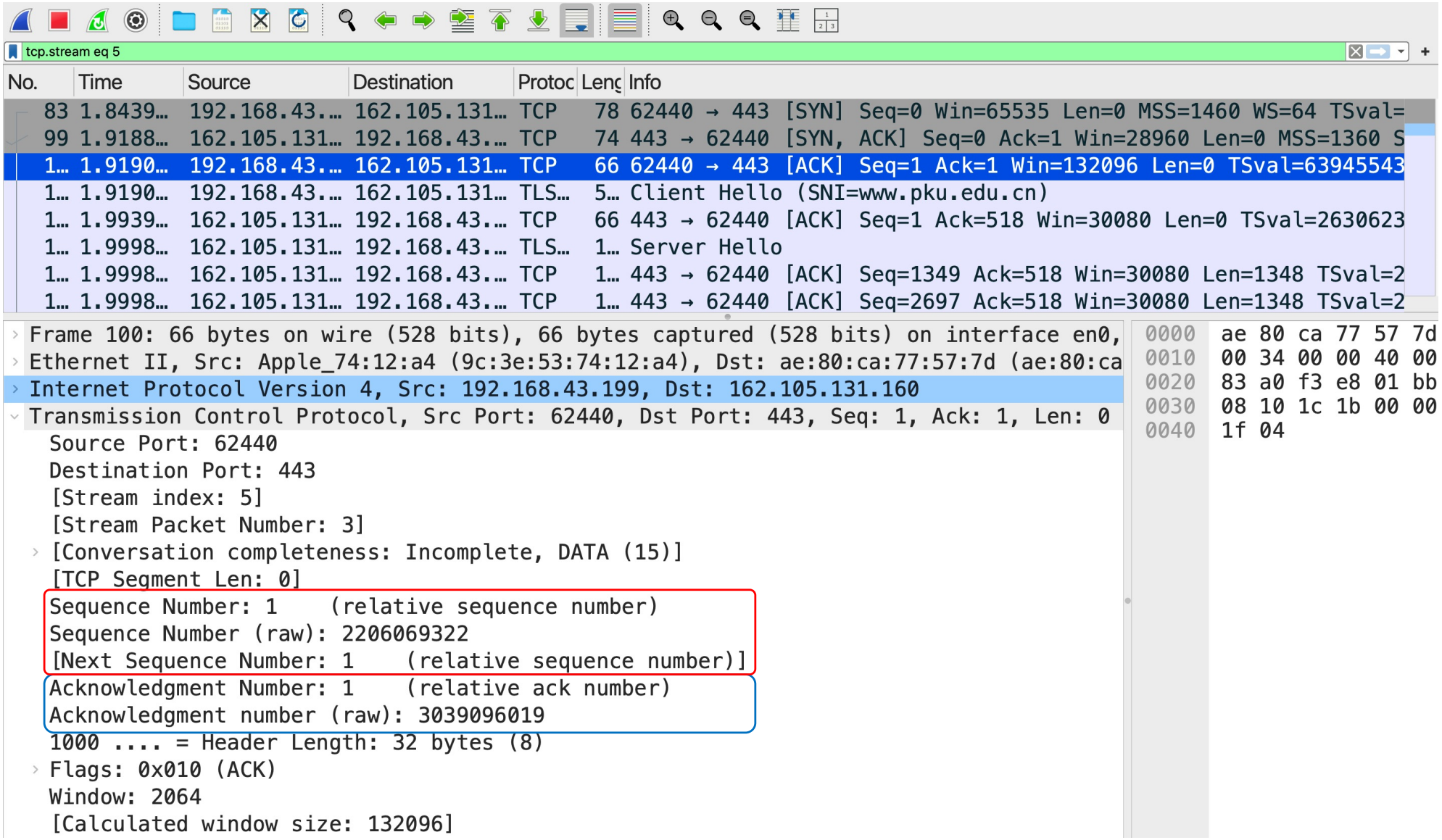
TCP Handshake: SYN



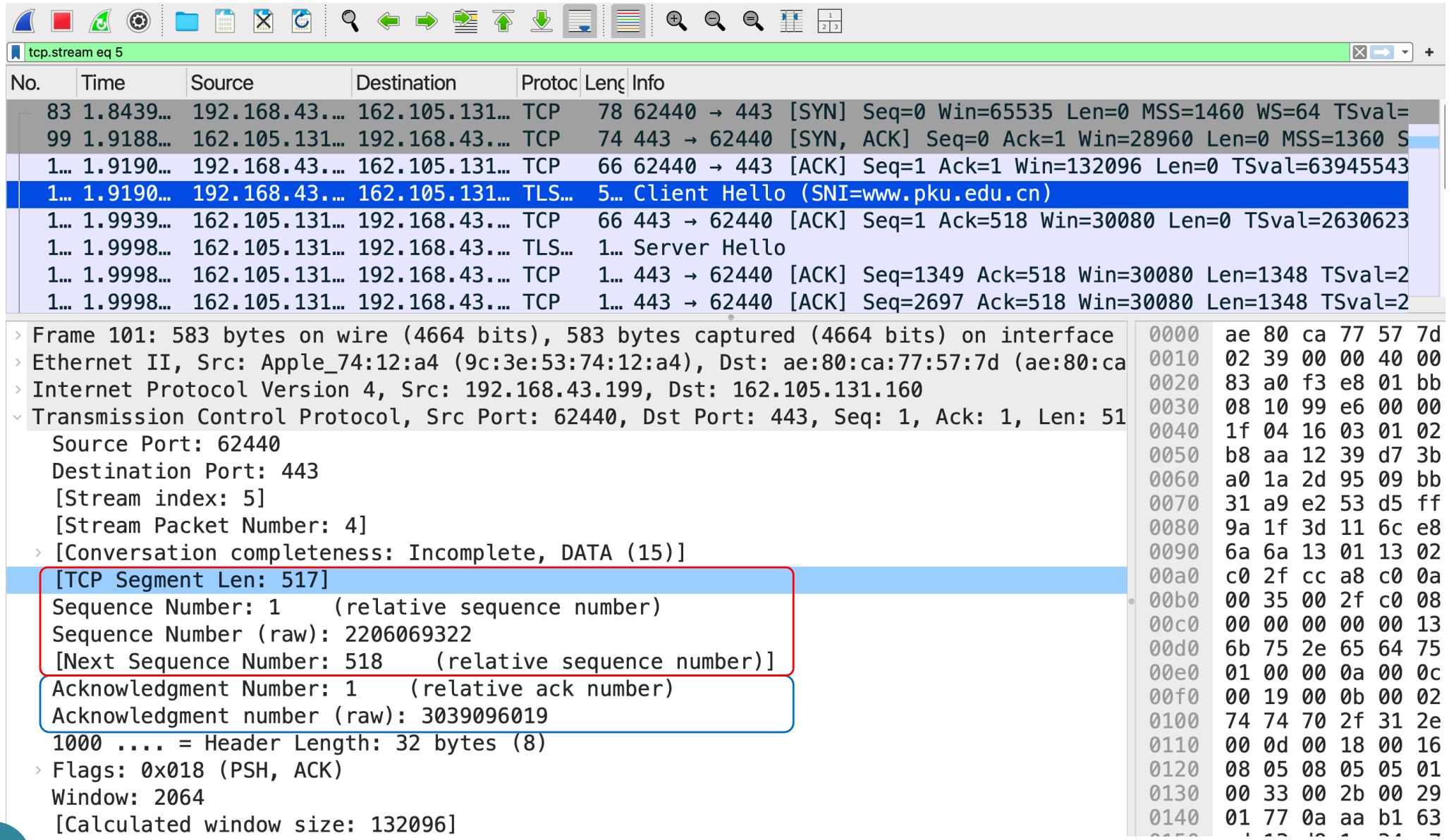
TCP Handshake: SYN-ACK



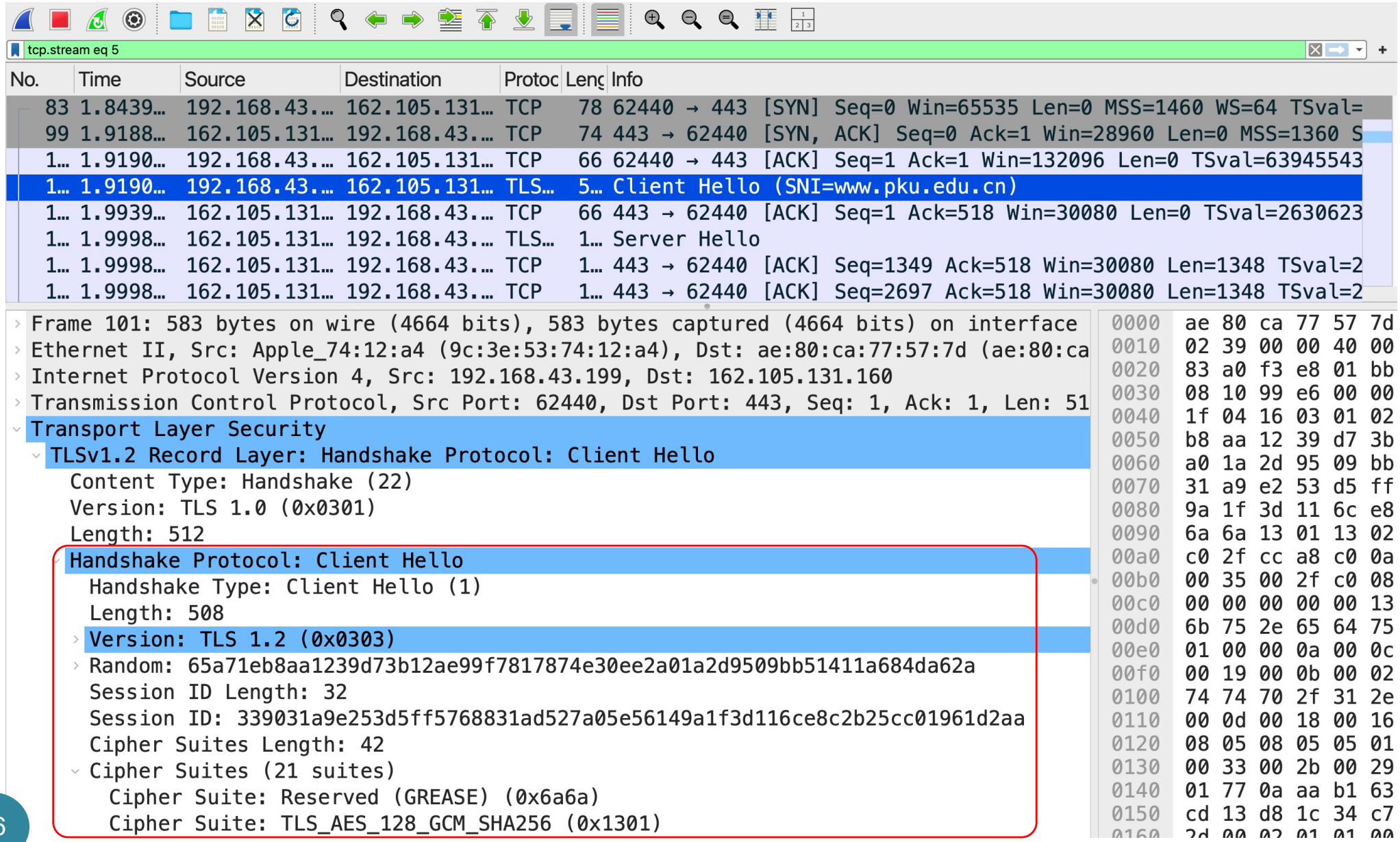
TCP Handshake: ACK



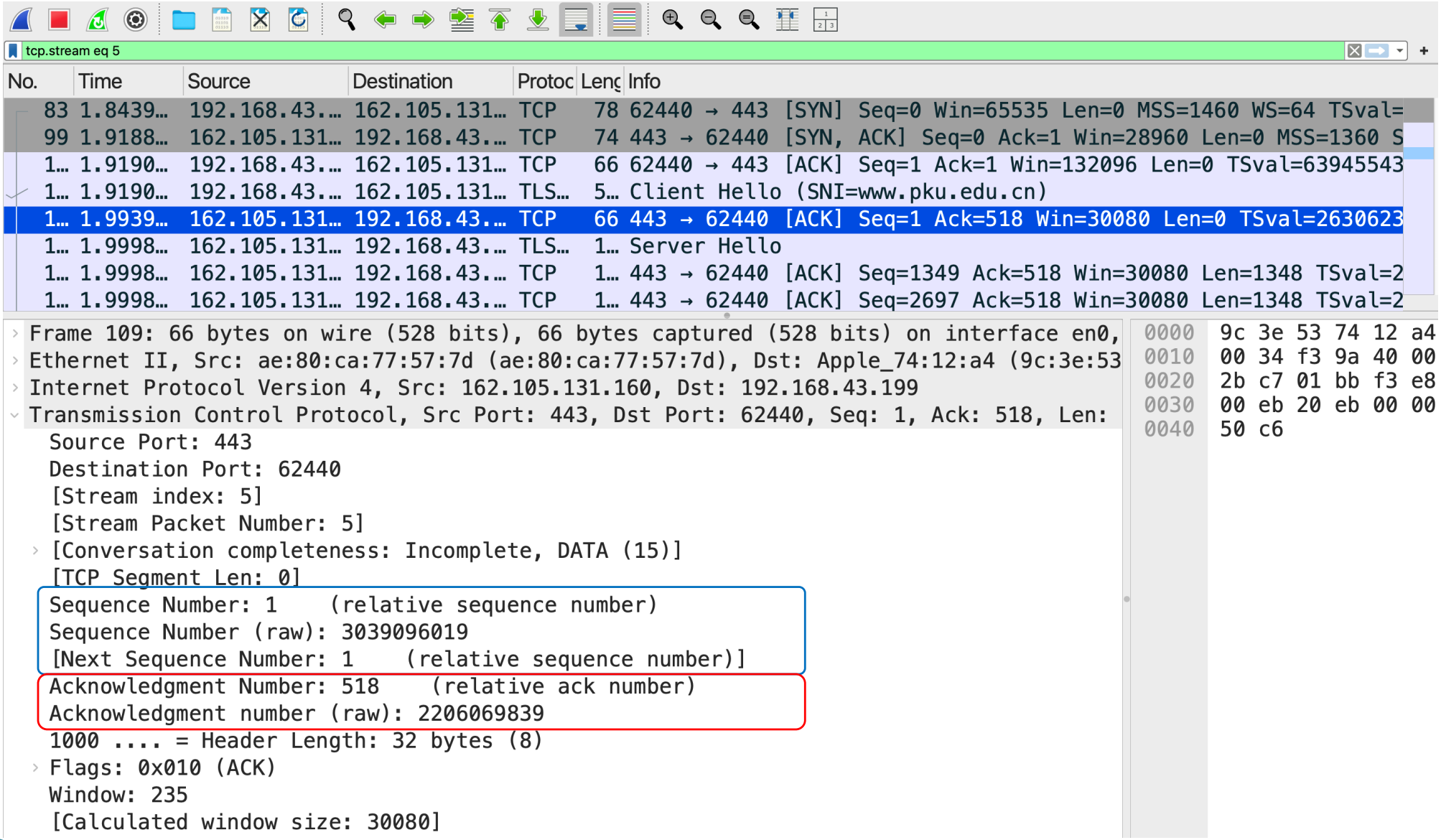
TCP: Client Hello



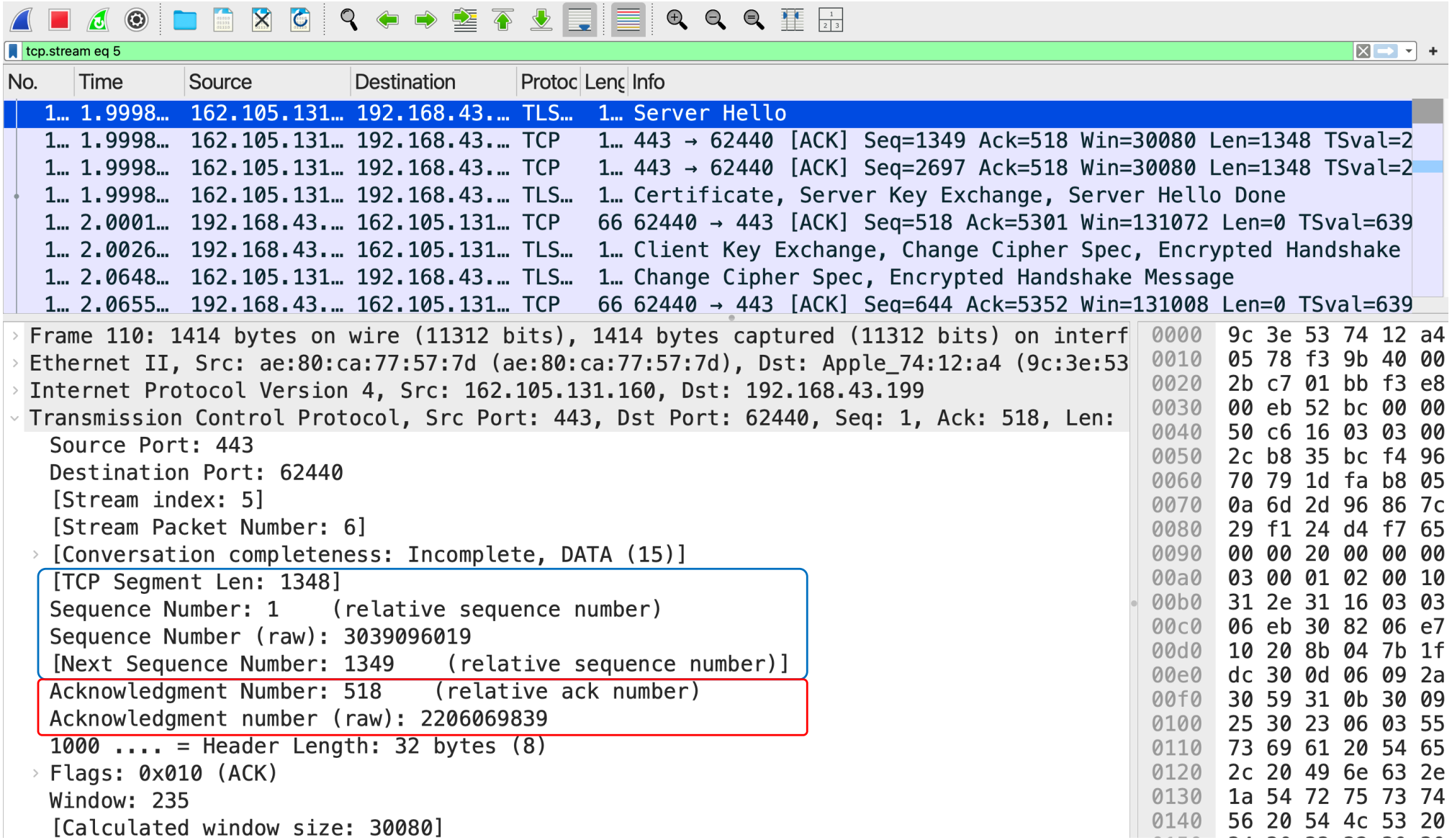
TLS: Client Hello



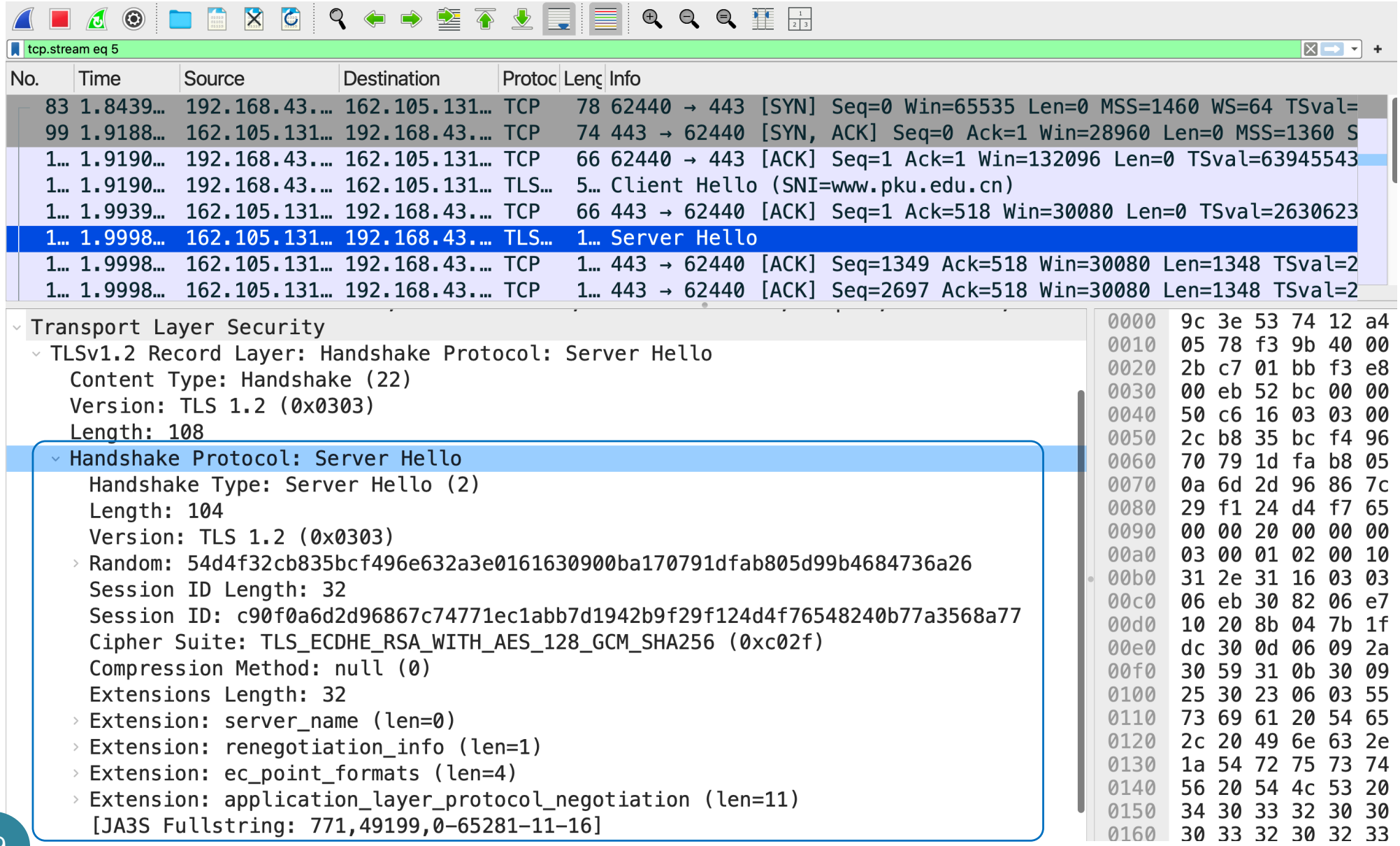
TCP: ACK Client Hello



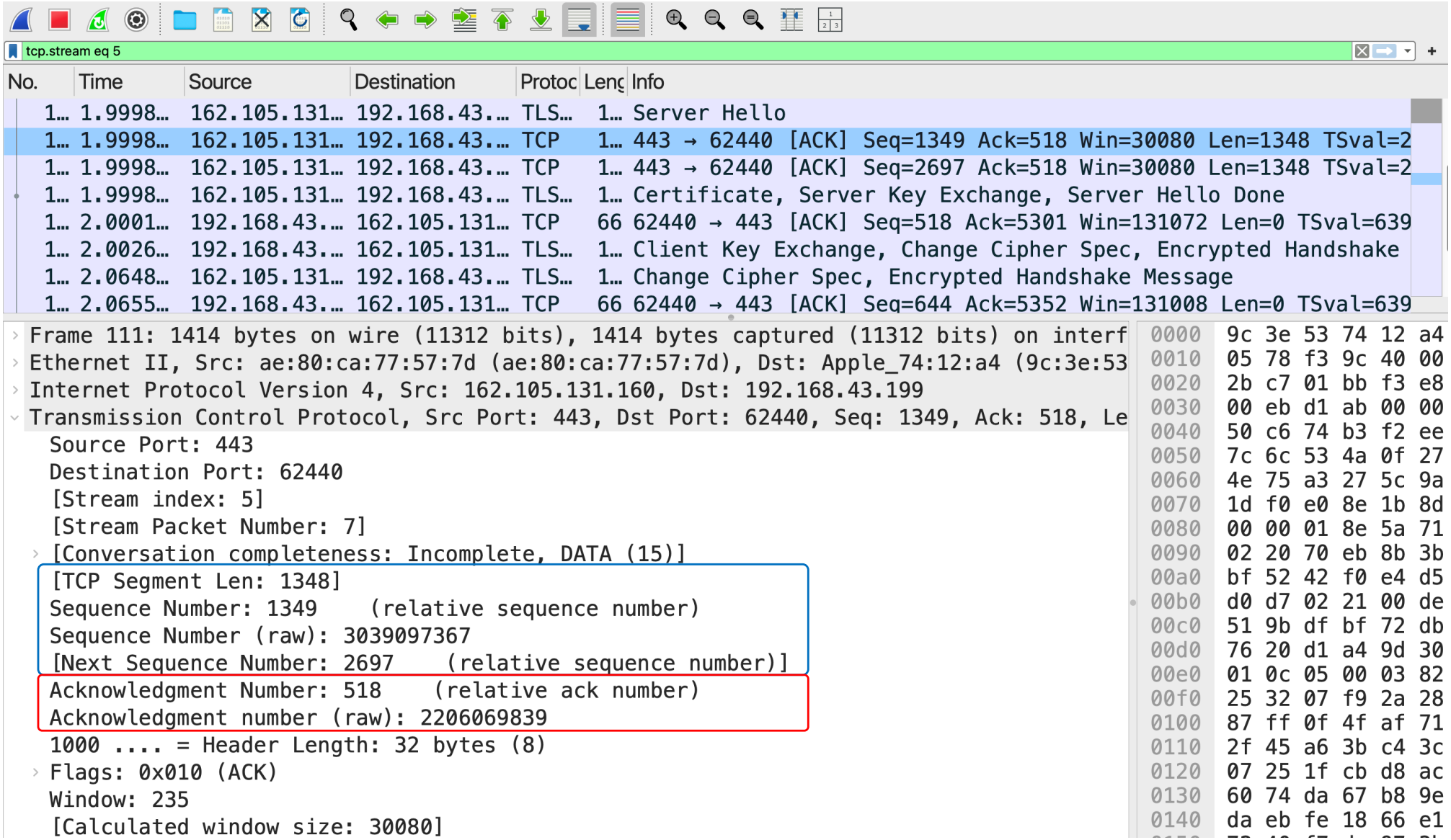
TCP: Server Hello



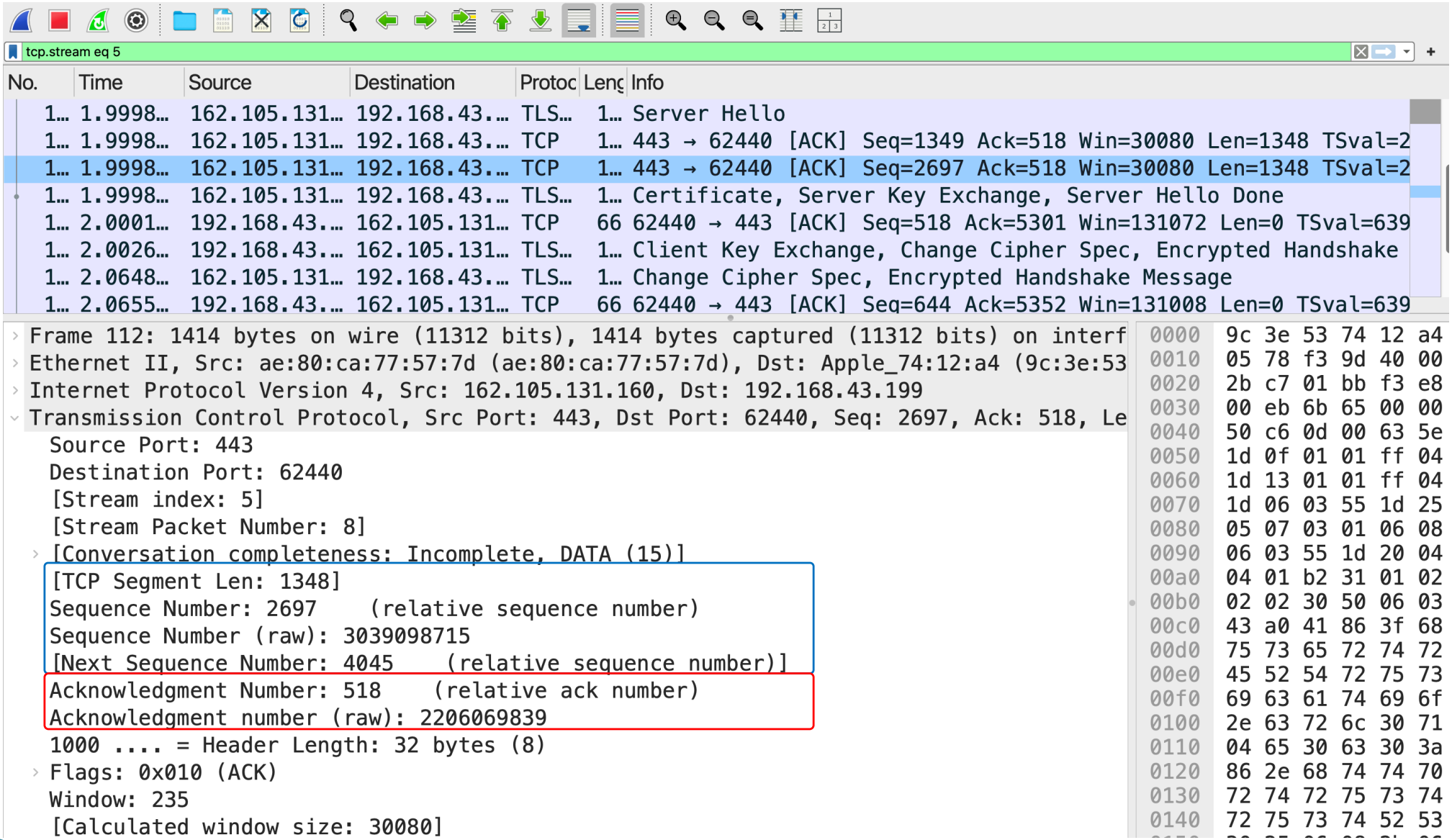
TLS: Server Hello



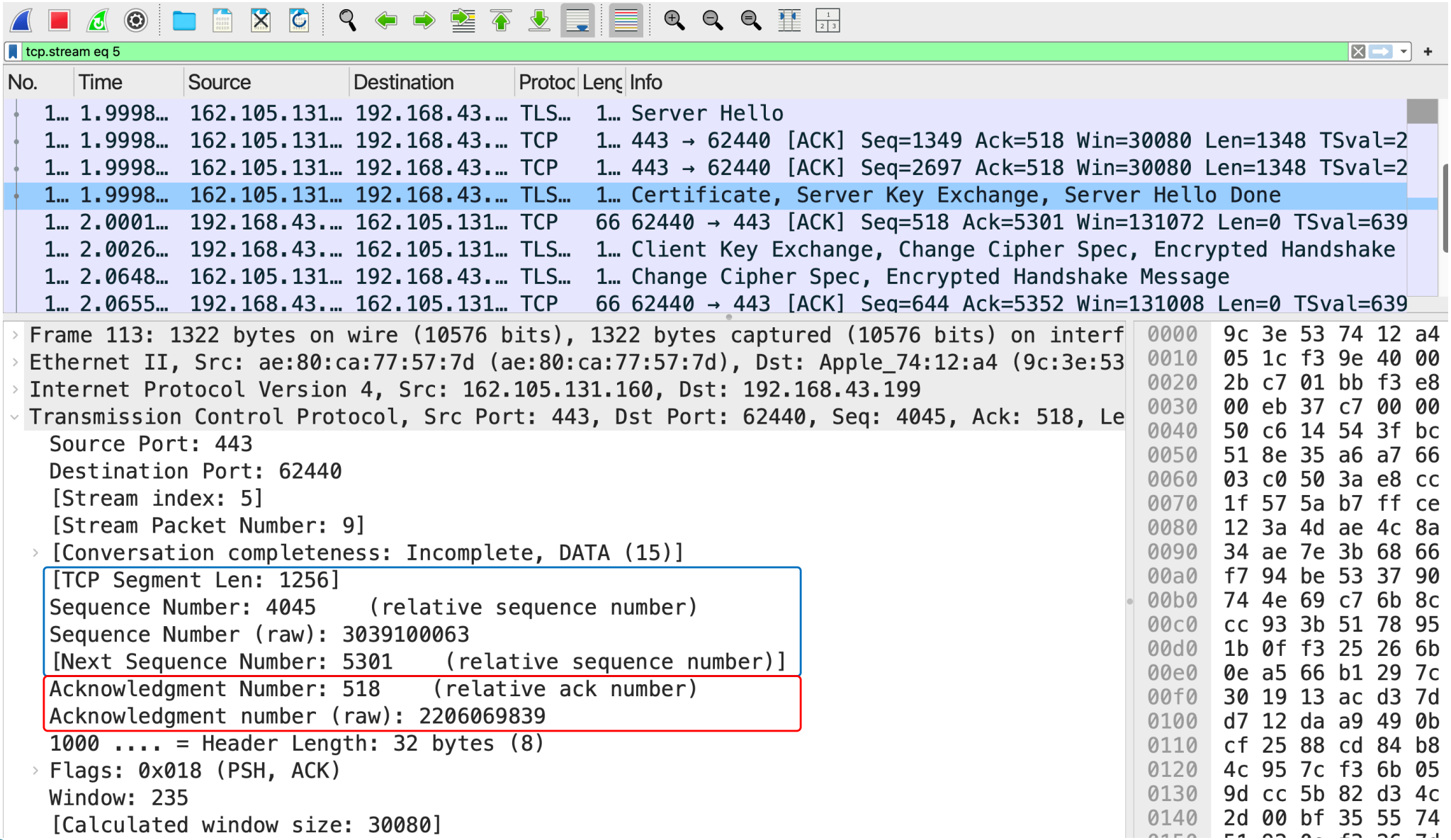
TCP: Certificate, Server Key Exchange, Done



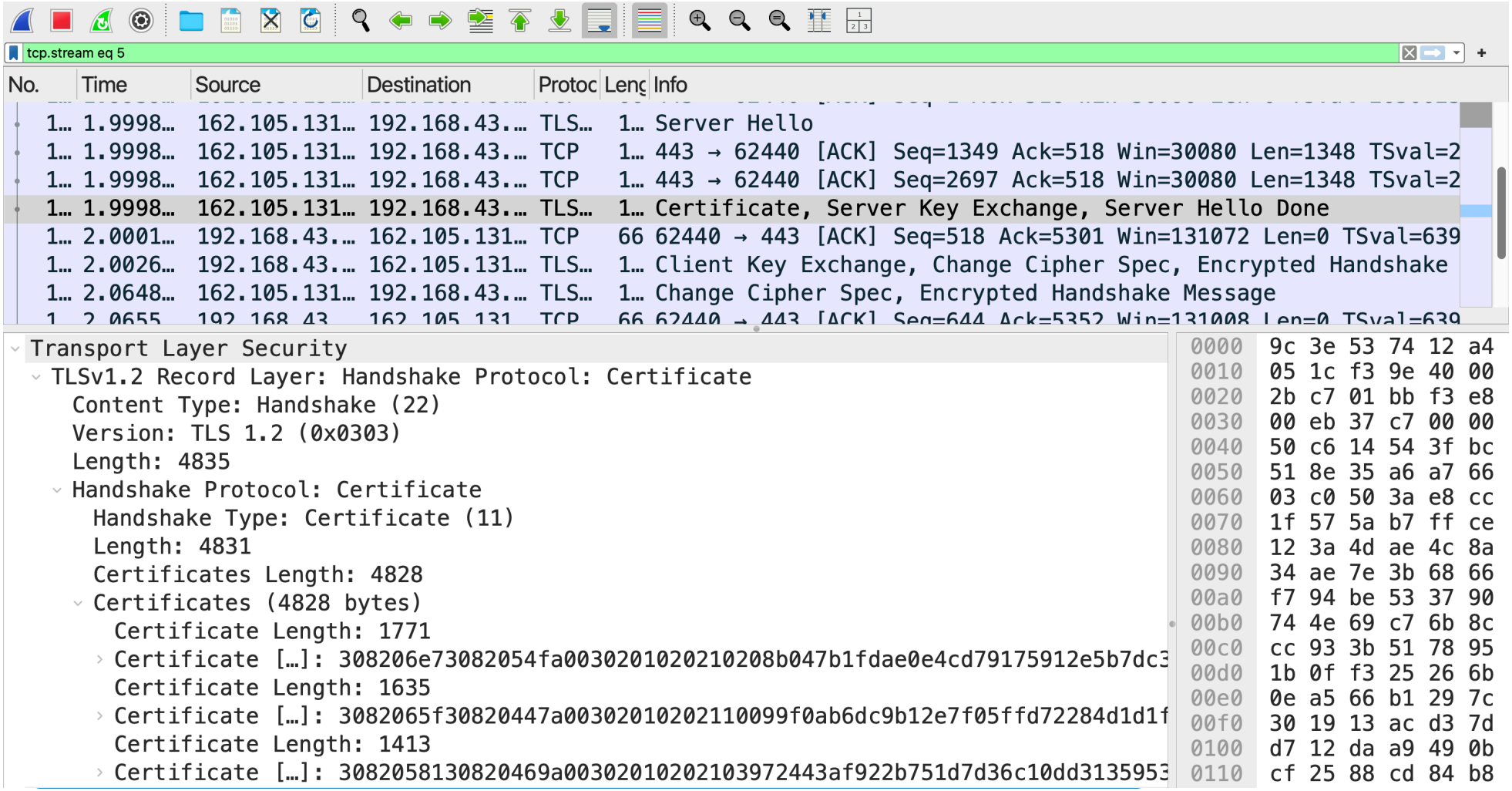
TCP: Certificate, Server Key Exchange, Done



TCP: Certificate, Server Key Exchange, Done



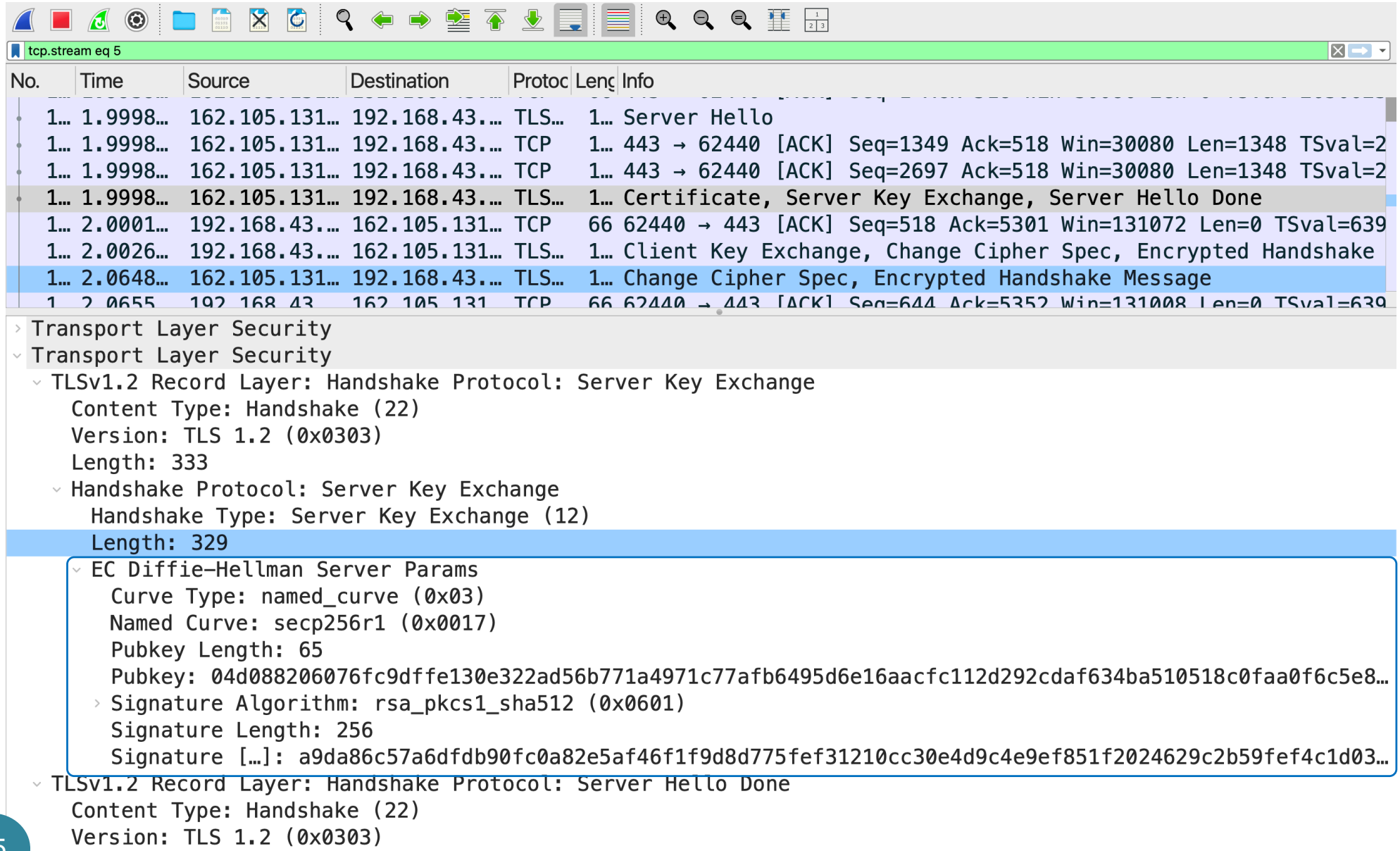
TLS: Certificate



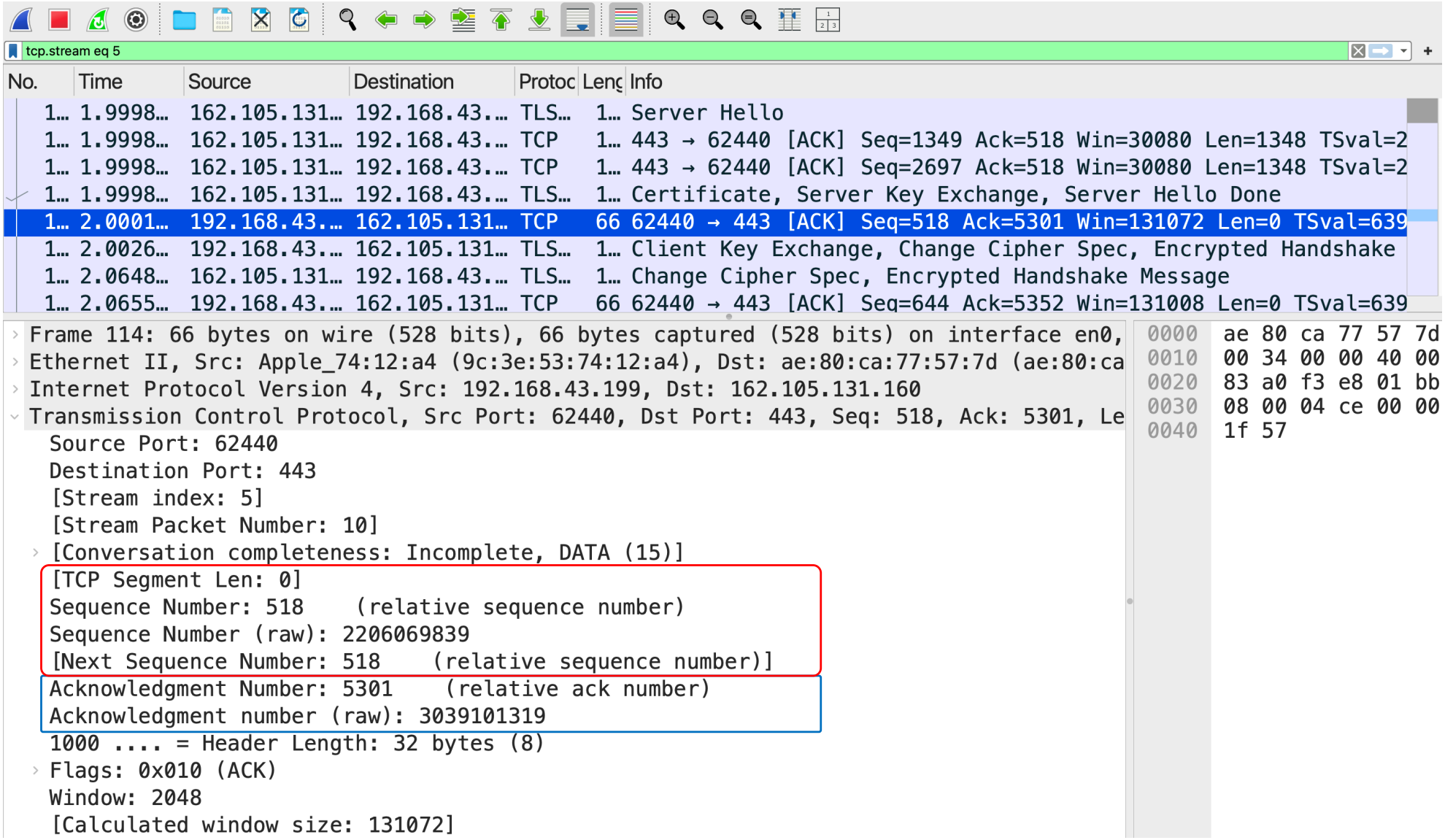
TLS: Certificate

```
▾ Certificates (4828 bytes)
    Certificate Length: 1771
  ▾ Certificate […]: 308206e73082054fa0030201020210208b047b1fdae0e4cd79175912e5b7dc300d06092a864886f70d01…
    ▾ signedCertificate
        version: v3 (2)
        serialNumber: 0x208b047b1fdae0e4cd79175912e5b7dc
      › signature (sha384WithRSAEncryption)
      ▾ issuer: rdnSequence (0)
        › rdnSequence: 3 items (id-at-commonName=TrustAsia RSA OV TLS CA G3,id-at-organizationName=TrustAsi…
      › validity
      ▾ subject: rdnSequence (0)
        › rdnSequence: 4 items (id-at-commonName=www.pku.edu.cn,id-at-organizationName=北京大学,id-at-stateO…
      › subjectPublicKeyInfo
      › extensions: 10 items
    › algorithmIdentifier (sha384WithRSAEncryption)
      Padding: 0
      encrypted […]: 46cd8ac5a0e18a253207f92a280120e0d97df9dd5f6aa587ff0f4faf710b74131e678536d6e1b52f45a63…
    Certificate Length: 1635
  › Certificate […]: 3082065f30820447a00302010202110099f0ab6dc9b12e7f05ffd72284d1d1ff300d06092a864886f70d…
```

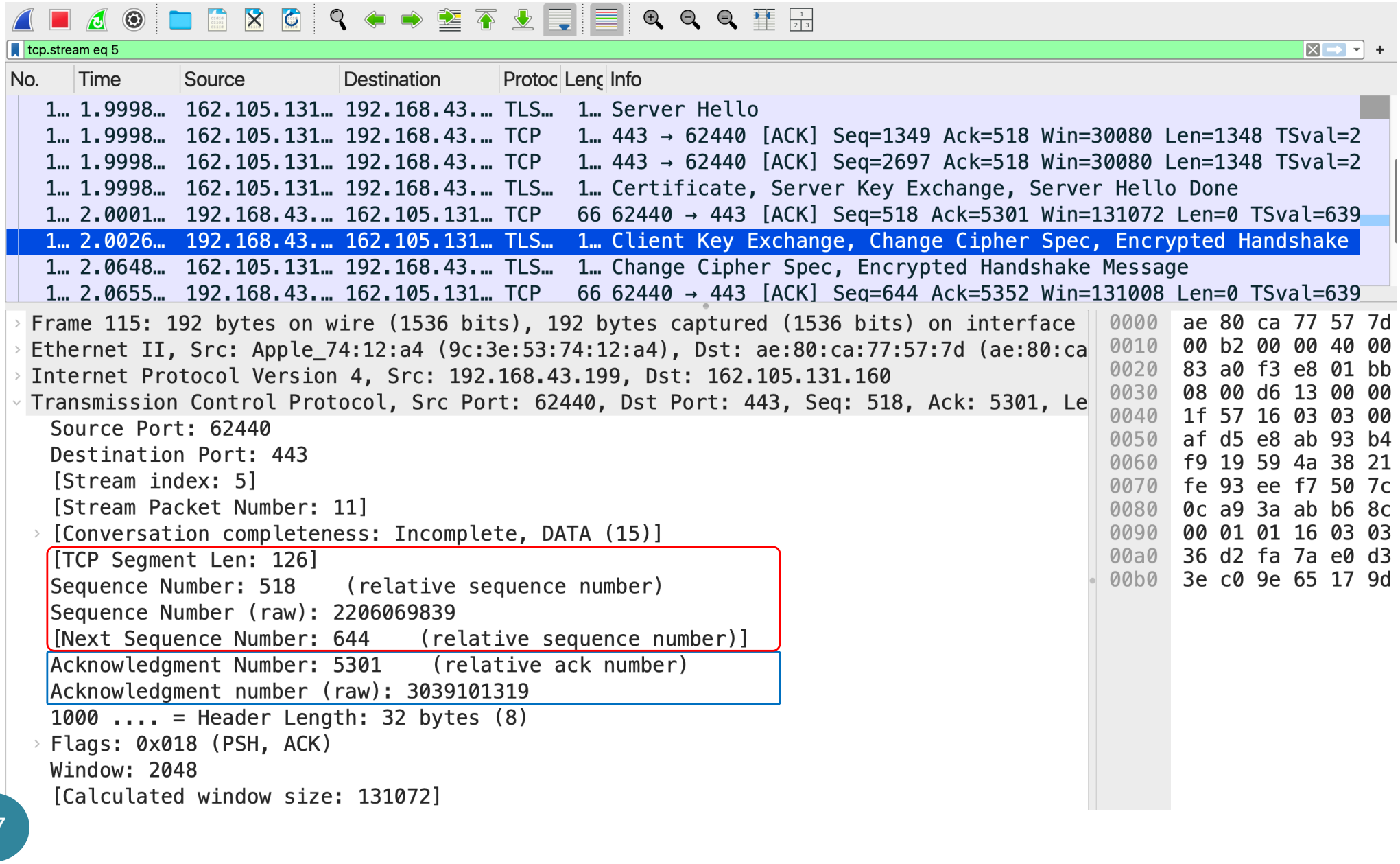
TLS: Server Key Exchange, Server Hello Done



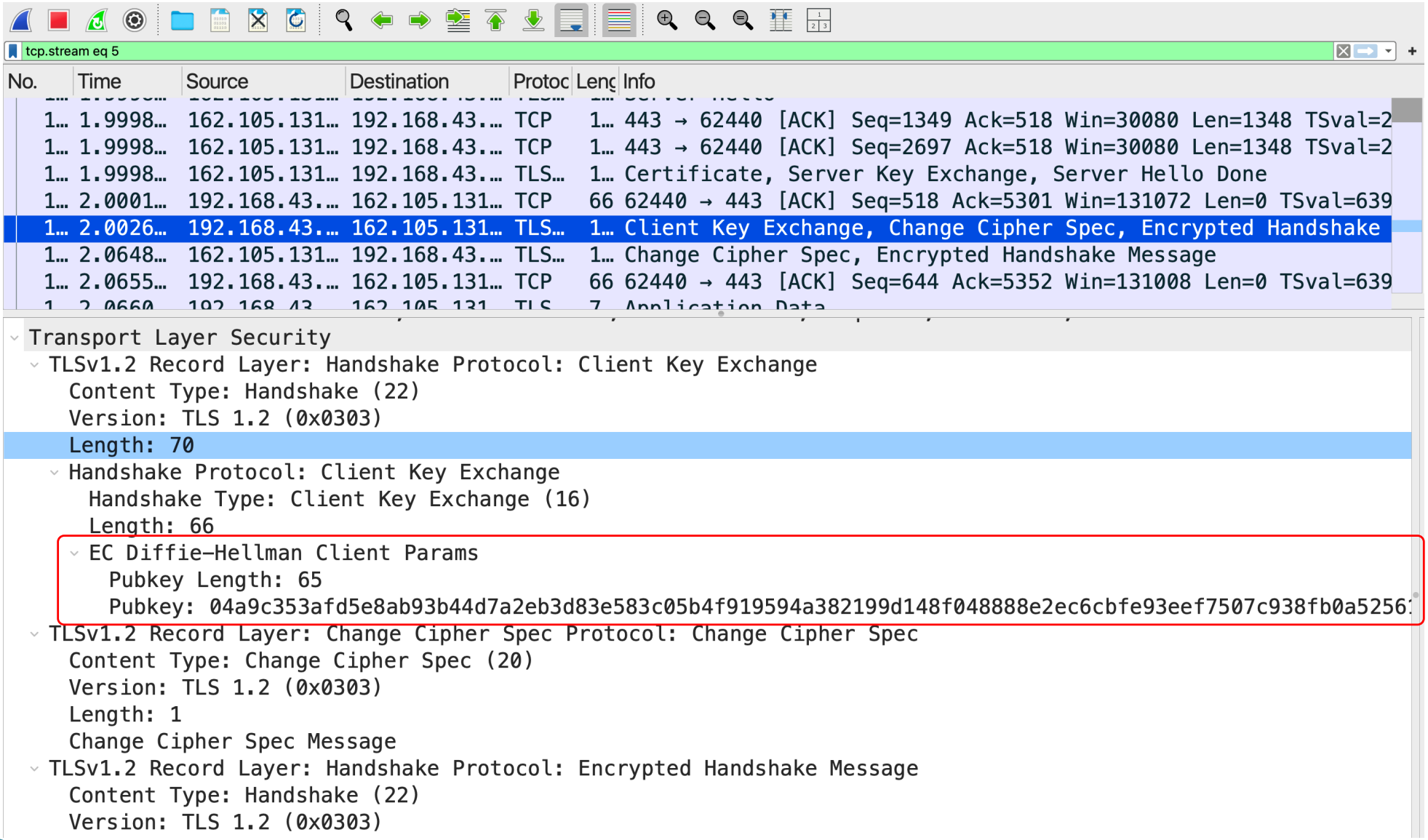
TCP: ACK Server Hello



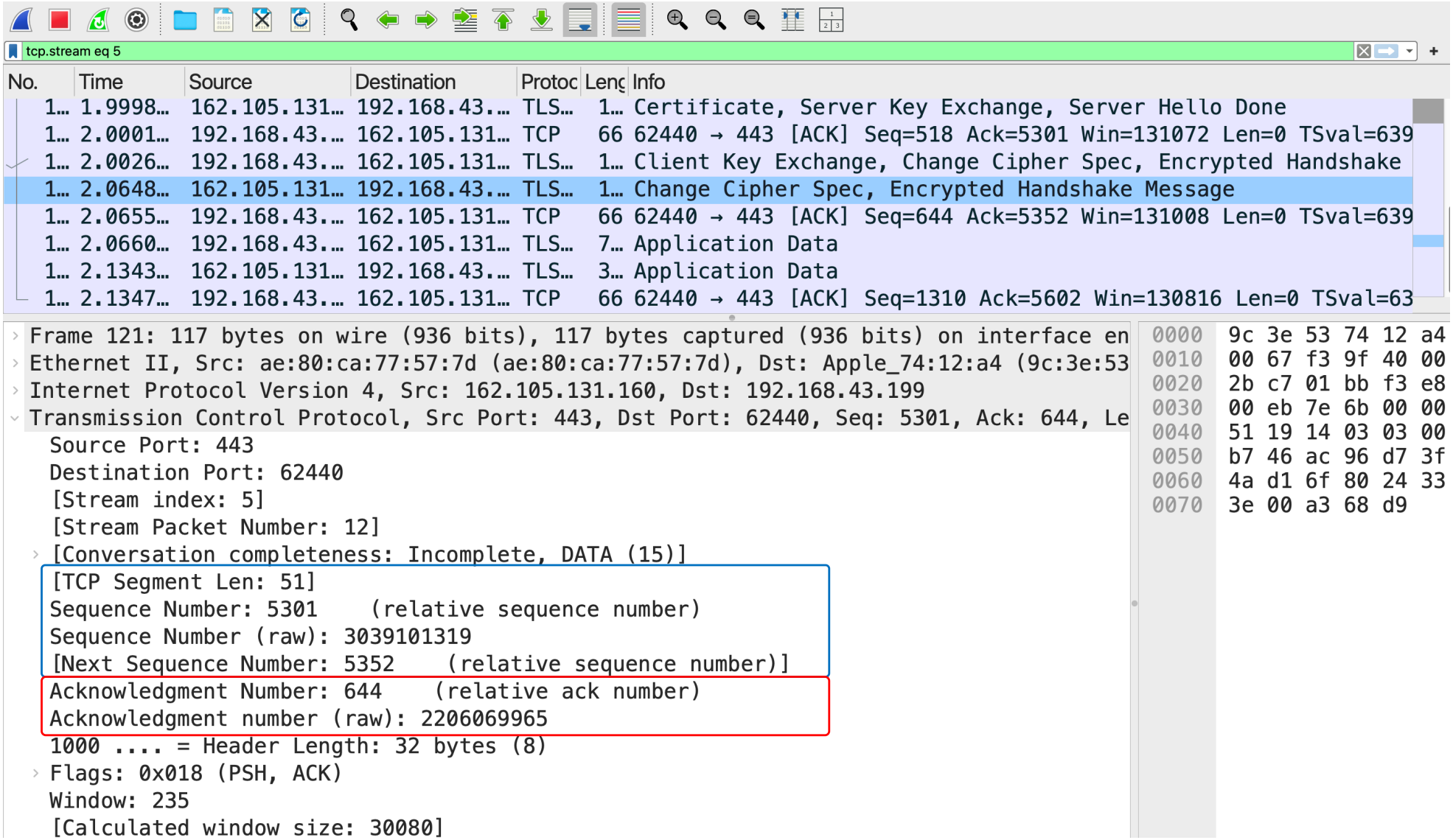
TCP: Client Key Exchange, Change Cipher Spec



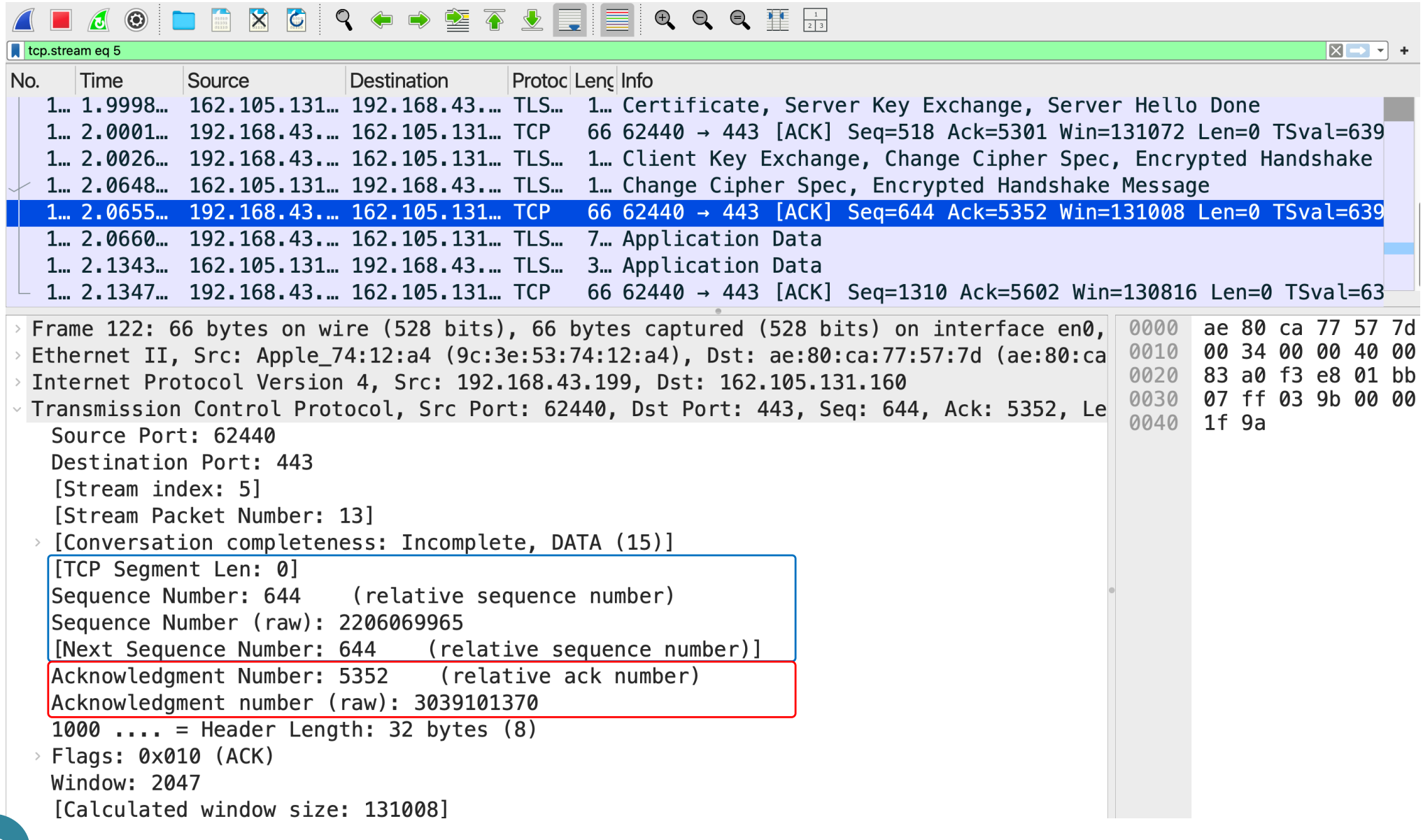
TLS: Client Key Exchange



TCP: Change Cipher Spec



TCP: ACK Change Cipher Spec



4. In-class Practice

Step 1

- Assume the RSA key of the root CA:
  - p=3, q=11, e=7
  - => public key: (7, 33), private key: 3
- Demonstrate the full process to issue a digital cert to yourself.
  - Generate a key pair
  - Sign the cert
  - Assume the hash function: $Hash(M) = \left(\sum_{i=1}^{l} m_i\right) \bmod 256$

    Version: x509.v3

    Serial Number: 100100

    Subject: CN = YOURNAME, O = Fudan University, C = CN

    Issuer: CN = RootCA, O = Fudan University, C = CN

    Validity: 2024 - 0901 to 2027-07-31

Step 2

- Sign the message with FISF's digital certificate
  - "FISF is a great school of Fudan University"
- Demonstrate the full process of signature verification
  - Verify the message signature
  - Verify the certificate signature

Step 3: Experiment

- Repeat the process with a real PKI tool: https://pkitools.net
  - Generate a certificate
  - Sign a message and verify the signed message