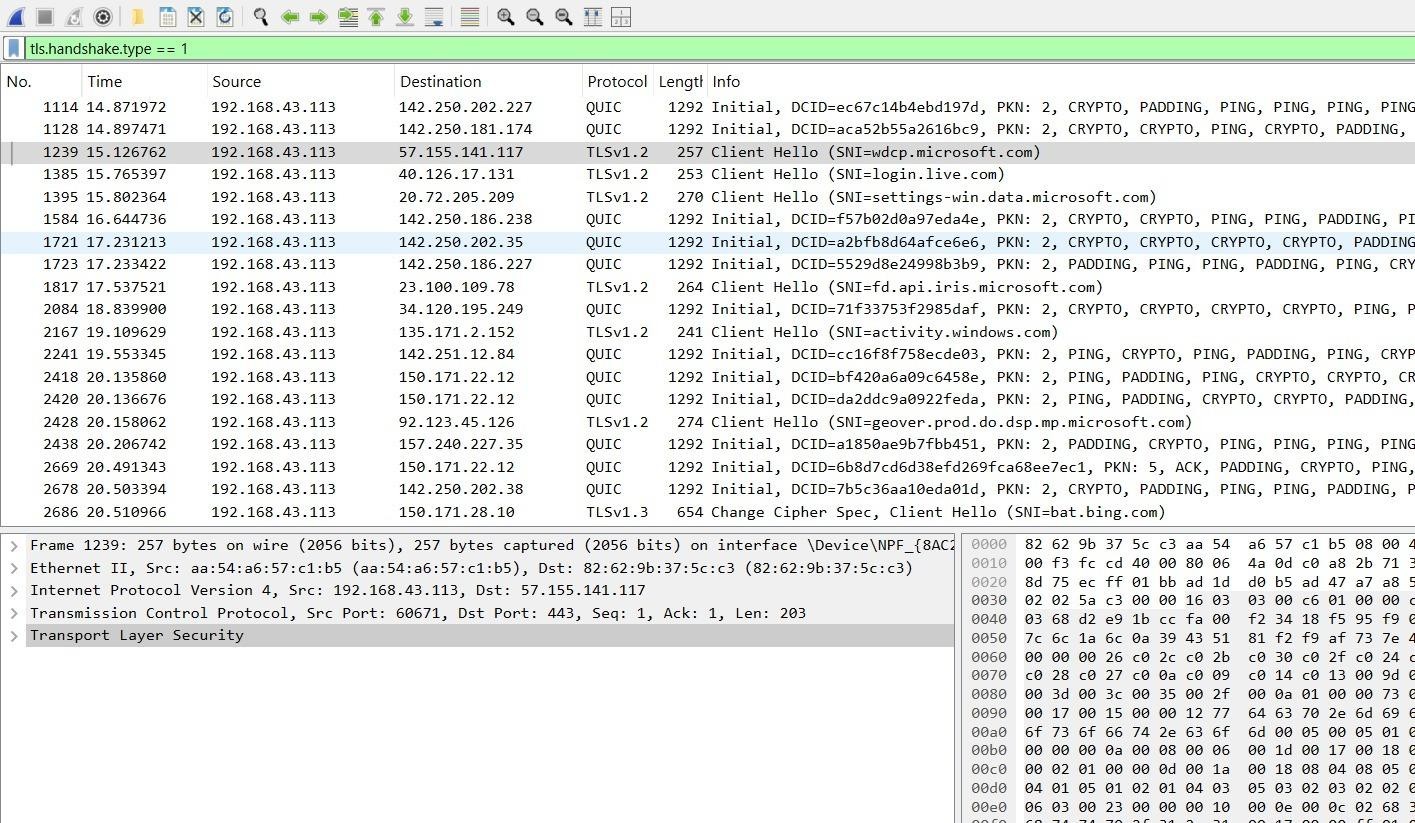
# What is the name of website?

* + You’ll see it in the **SNI (Server Name Indication)** extension in the **ClientHello**

message.

→ It should show [**www.microsoft.com**](http://www.microsoft.com/).

# Find the packet that contains the ClientHello message for the website you are accessing.

* + Look for a packet with **TLSv1.2 or TLSv1.3 → Handshake → Client Hello**.
  + Usually one of the first packets after the TCP handshake.
  + Note the **packet number** in Wireshark.

# G. List all the TLS extensions included in the ClientHello.

* Expand:

Frame → Transport Layer Security → Handshake Protocol: Client Hello

* Scroll down to **Extensions**. Common ones include:
  + Server Name Indication (SNI)
  + Supported Groups
  + EC Point Formats
  + Signature Algorithms
  + Supported Versions (TLS 1.3, TLS 1.2)
  + Session Ticket
  + Key Share (TLS 1.3)
  + ALPN (Application-Layer Protocol Negotiation, often "h2" and "http/1.1")



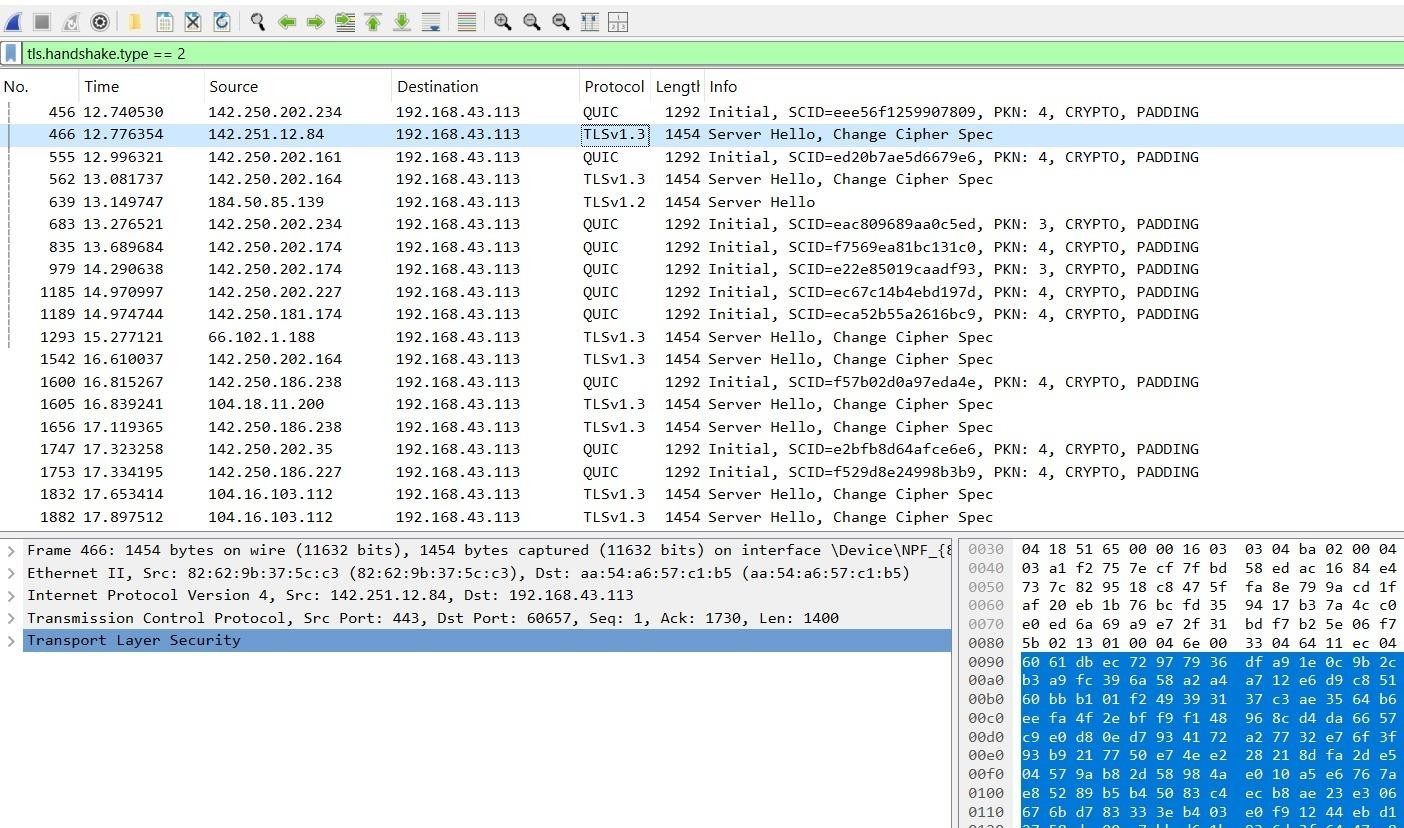
# Identify the ServerHello message. What cipher suite is chosen by the server?

* + Find the **Server Hello** packet.
  + Expand the handshake details.
  + Look under:

Handshake Protocol: Server Hello → Cipher Suite

* + Example (for microsoft.com):

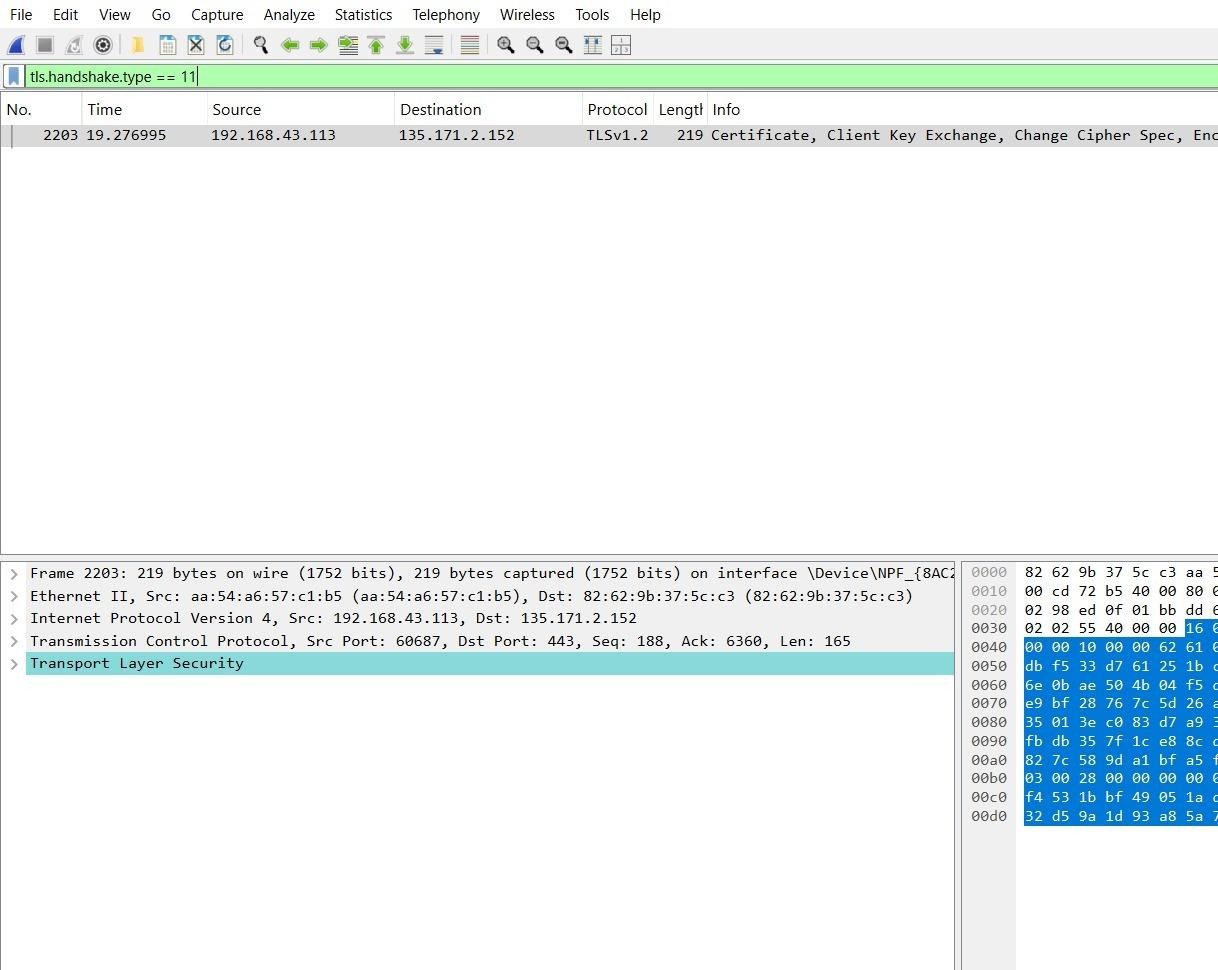
Cipher Suite: TLS\_AES\_256\_GCM\_SHA384 (0x1302)



# Locate the Certificate message. Extract the server’s certificate information

**(issuer, subject, validity dates).**

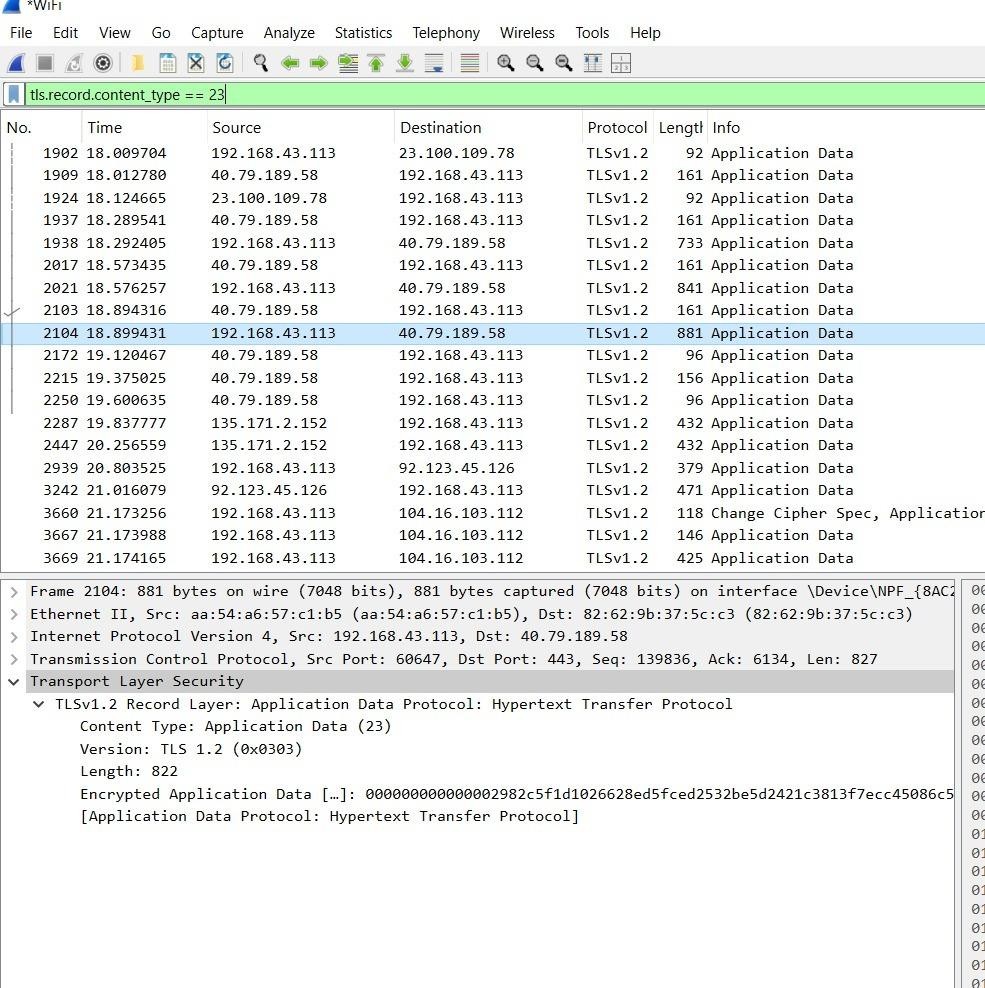
* + In the handshake, look for **Certificate** message.
  + Expand → Certificate → Certificate: <length> → Certificate
  + There you’ll see:
    - **Subject:** CN=[www.microsoft.com](http://www.microsoft.com/)
    - **Issuer:** Usually DigiCert or Microsoft CA
    - **Validity:** Not Before / Not After dates



# After the TLS handshake, identify the first encrypted application data packet. Why

**can’t you directly see the HTTP headers in this packet?**

* + Look for the first packet labeled **Application Data** after the handshake.
  + Reason:
    - Because HTTPS encrypts all application data after the handshake.
    - HTTP headers and content are encrypted inside TLS records.
    - Without decryption (and private keys), Wireshark can’t show them.



# In summary, the steps you’ll take in Wireshark:

1. Filter traffic: tls
2. Find Client Hello → note SNI, extensions.
3. Find Server Hello → note cipher suite.
4. Find Certificate → extract subject/issuer/validity.
5. Find Application Data → explain why encrypted.