Configure Packet Captures on AireOS WLC

Contents

Introduction

This document describes how to run a packet dump on a AireOS Wireless LAN Controller(WLC). This method displays the packets sent and/or received at CPU level of the WLC in hex format, which then be translated to a .pcap file with Wireshark.

It is helpful in cases where communication between a WLC and a Remote Authentication Dial-In User Service (RADIUS) server, an Access Point (AP) or other controllers needs to be verified in a quick way with a packet capture at the WLC level but a port-span is hard to perform.

Requirements

Cisco recommends that you have knowledge of these topics:

- Command line Interface (CLI) access to the WLC, preferrably SSH since the output is faster than console.
- PC with Wireshark installed

Components Used

The information in this document is based on these software and hardware versions:

- WLC v8.3
- Wireshark v2 or later

Note: This feature is available since AireOS version 4.

Limitations

The packet logging will capture only bidirectional Control Plane (CP) to Data Plane (DP) packets in WLC. Those packets which are not sent from WLC Data plane to/from control plane (i.e. foreign to anchor tunneled traffic, DP-CP drops and so on) will not be captured.

Examples of types of traffic to/from the WLC processed at the CP are:

- Telnet
- SSH
- HTTP
- HTTPS
- SNMP
- NTP
- RADIUS
- TACACS+
- Mobility Messages
- CAPWAP control
- NMSP
- TFTP/FTP/SFTP
- Syslog
- IAPP

The traffic to/from the client is processed in the Data Plane (DP) except for: 802.11 management, 802.1X/EAPOL, ARP, DHCP and Web Authentication.

Configure

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Enable packet logging in WLC

Step 1. Log in to WLC's CLI.

Due to the quantity and speed of logs that this feature displays it is recommended to login to the WLC by SSH and not by console.

Step 2. Apply an Access Control List (ACL) to limit which traffic is captured.

In the given example the capture shows the traffic to/from the WLC's management interface (IP address 172.16.0.34) and the RADIUS server (172.16.56.153).

```
> debug packet logging acl ip 1 permit 172.16.0.34 172.16.56.153
> debug packet logging acl ip 2 permit 172.16.56.153 172.16.0.34
```

> **Tip**: To capture all the traffic to/from the WLC it is recommended to apply an ACL that discards the SSH traffic to/from the host that initiated the SSH session. These are the commands that you can use to build the ACL:
>
> >debug packet logging acl ip 1 deny <WLC-IP> <host-IP> tcp 22 any
> >debug packet logging acl ip 2 deny <host-IP> <WLC-IP> tcp any 22
> >debug packet logging acl ip 3 permit any any

Step 3. Configure the format readable by Wireshark.

```
> debug packet logging format text2pcap
```

Step 4. Enable packet logging feature.

This example shows how to capture 100 received/transmitted packets (it supports 1 - 65535 packets):

```
> debug packet logging enable all 100
```

Step 5. Log the output to a text file.

> **Note:** By default, it only logs 25 received packets with the command **debug packet logging enable**.

> **Note**: Instead of **all** you can use **rx** or **tx** to capture only received or transmitted traffic.

For further details about configuring packet logging feature consult this link:

Cisco Wireless Controller Configuration Guide, Release 8.3, Using the Debug Facility

Verify

Use this section in order to confirm that your configuration works properly.

Use the given command to verify the current configuration of packet logging.

```
> show debug packet

Status..................................... rx/tx            !!! This means the capture is
active
Number of packets to display..................... 100
Bytes/packet to display.......................... 0
Packet display format............................ text2pcap

   Driver ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   Ethernet ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   IP ACL:
```

```
      [1]: permit s=172.16.0.34 d=172.16.56.153 any
      [2]: permit s=172.16.56.153 d=172.16.0.34 any
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   EoIP-Ethernet ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   EoIP-IP ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   LWAPP-Dot11 ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   LWAPP-IP ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
```

Reproduce the needed behavior to generate the traffic.

An output similar to this appears:

```
> show debug packet

Status..................................... rx/tx            !!! This means the capture is
active
Number of packets to display..................... 100
Bytes/packet to display.......................... 0
Packet display format............................ text2pcap

   Driver ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   Ethernet ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
```

```
   IP ACL:
      [1]: permit s=172.16.0.34 d=172.16.56.153 any
      [2]: permit s=172.16.56.153 d=172.16.0.34 any
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   EoIP-Ethernet ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   EoIP-IP ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   LWAPP-Dot11 ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   LWAPP-IP ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
```

Remove ACLs from packet logging

In order to disable the filters applied by the ACLs use these commands:

```
> show debug packet

Status..................................... rx/tx            !!! This means the capture is
active
Number of packets to display..................... 100
Bytes/packet to display.......................... 0
Packet display format............................ text2pcap

   Driver ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   Ethernet ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
```

```
      [6]: disabled
   IP ACL:
      [1]: permit s=172.16.0.34 d=172.16.56.153 any
      [2]: permit s=172.16.56.153 d=172.16.0.34 any
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   EoIP-Ethernet ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   EoIP-IP ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   LWAPP-Dot11 ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   LWAPP-IP ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
```

Disable packet logging

In order to disable the packet logging without removing the ACLs simply use this command:

```
> show debug packet

Status..................................... rx/tx            !!! This means the capture is
active
Number of packets to display..................... 100
Bytes/packet to display.......................... 0
Packet display format............................ text2pcap

   Driver ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
      [5]: disabled
      [6]: disabled
   Ethernet ACL:
      [1]: disabled
      [2]: disabled
      [3]: disabled
      [4]: disabled
```

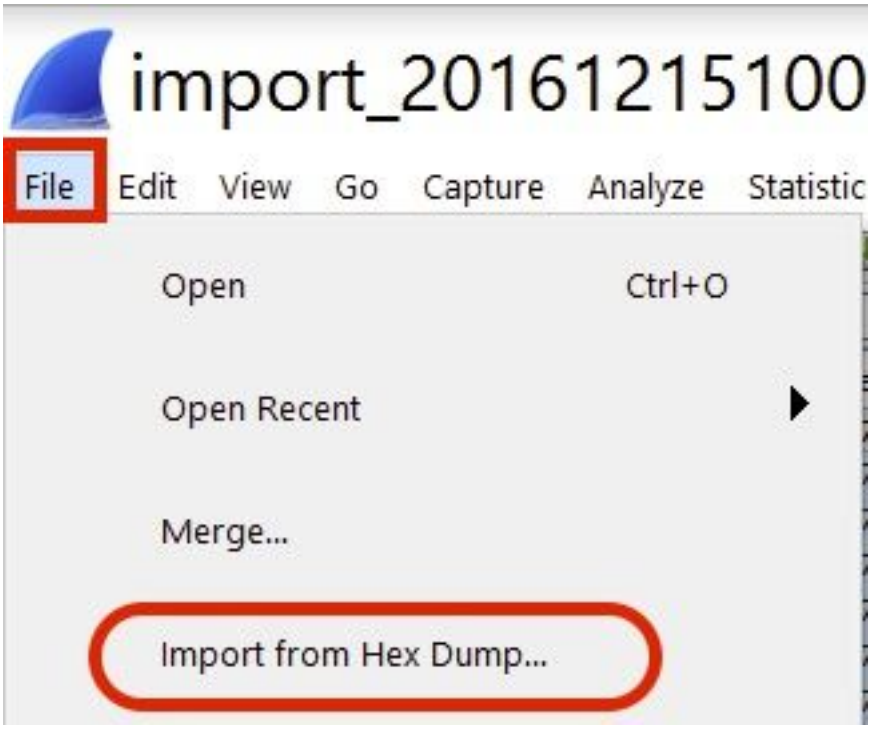
```
   [5]: disabled
   [6]: disabled
IP ACL:
   [1]: permit s=172.16.0.34 d=172.16.56.153 any
   [2]: permit s=172.16.56.153 d=172.16.0.34 any
   [3]: disabled
   [4]: disabled
   [5]: disabled
   [6]: disabled
EoIP-Ethernet ACL:
   [1]: disabled
   [2]: disabled
   [3]: disabled
   [4]: disabled
   [5]: disabled
   [6]: disabled
EoIP-IP ACL:
   [1]: disabled
   [2]: disabled
   [3]: disabled
   [4]: disabled
   [5]: disabled
   [6]: disabled
LWAPP-Dot11 ACL:
   [1]: disabled
   [2]: disabled
   [3]: disabled
   [4]: disabled
   [5]: disabled
   [6]: disabled
LWAPP-IP ACL:
   [1]: disabled
   [2]: disabled
   [3]: disabled
   [4]: disabled
   [5]: disabled
   [6]: disabled
```

Convert packet logging output to a .pcap file

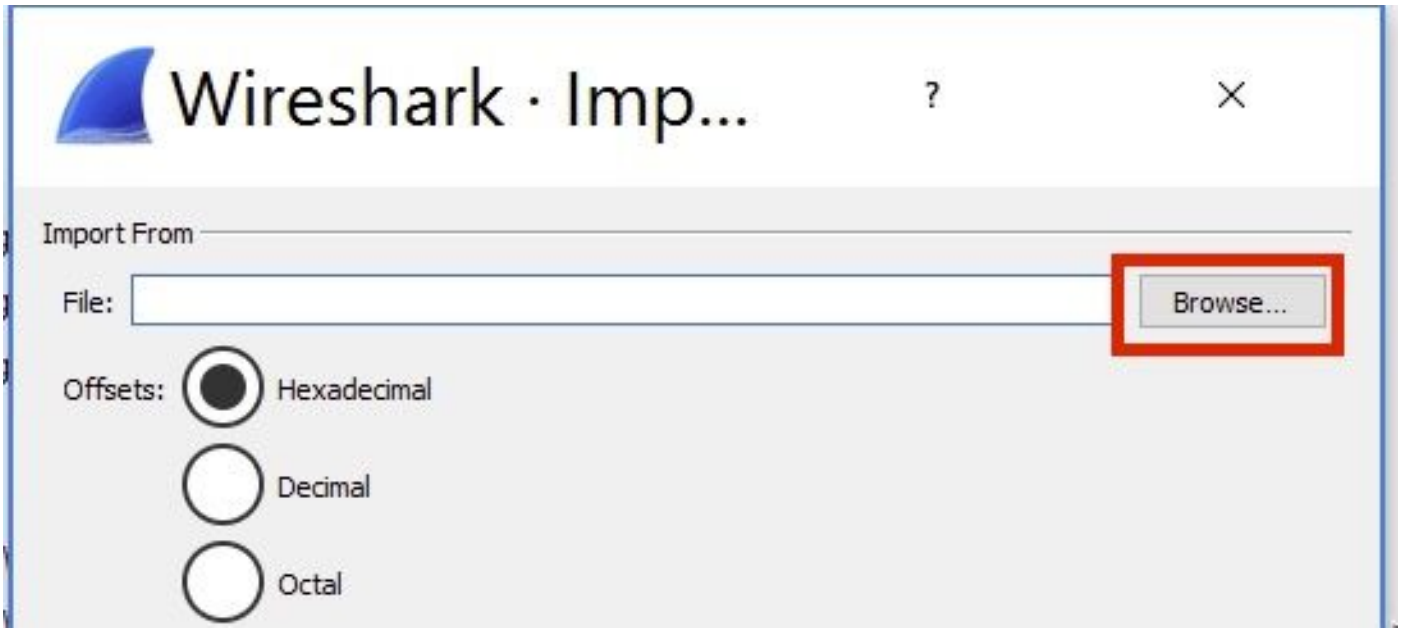
Step 1. Once the output finishes, collect it and save it to a text file.

Ensure that you gather a clean log, otherwise Wireshark might show corrupted packets.

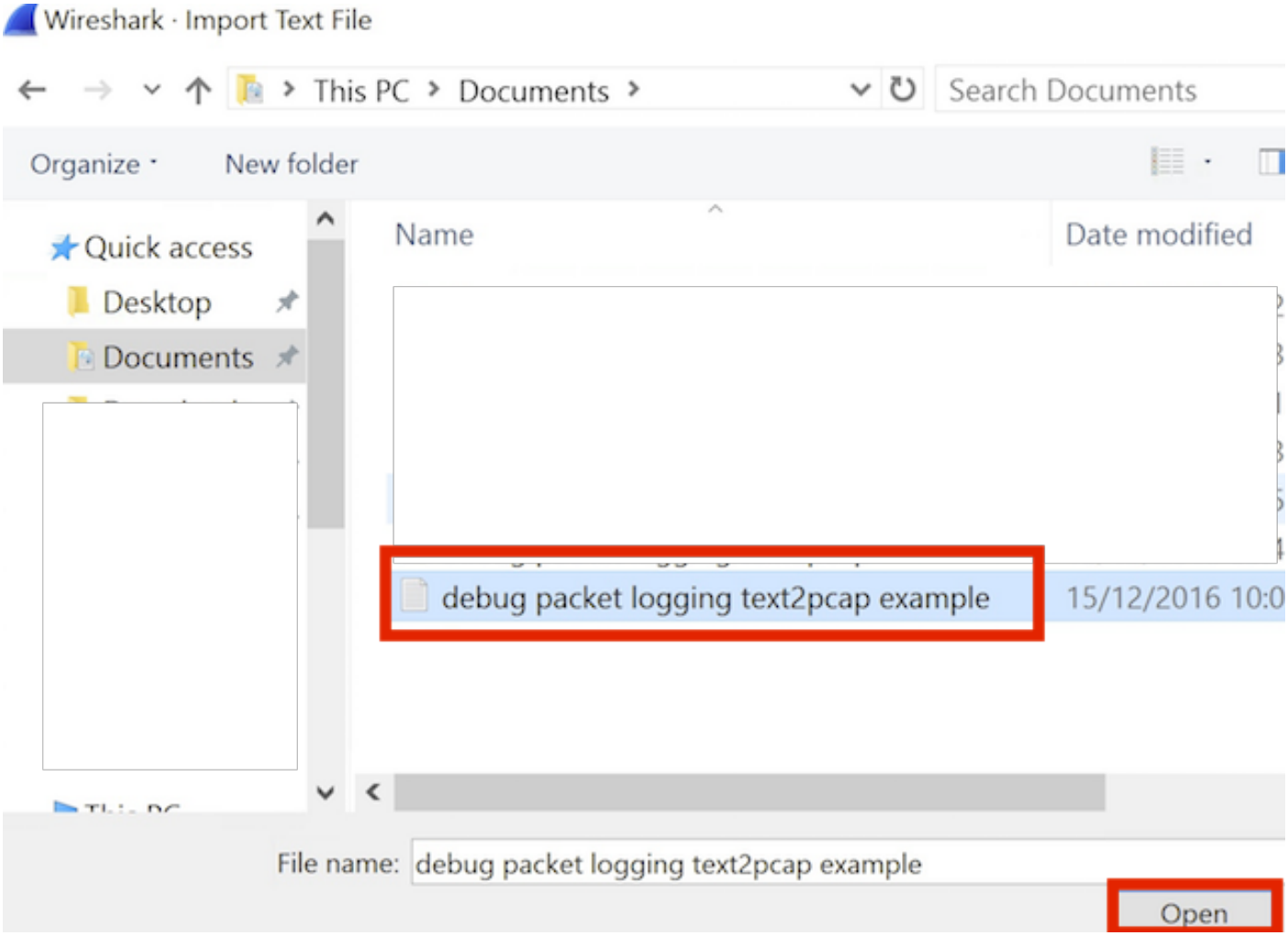
Step 2. Open Wireshark and navigate to **File>Import from Hex Dump...**



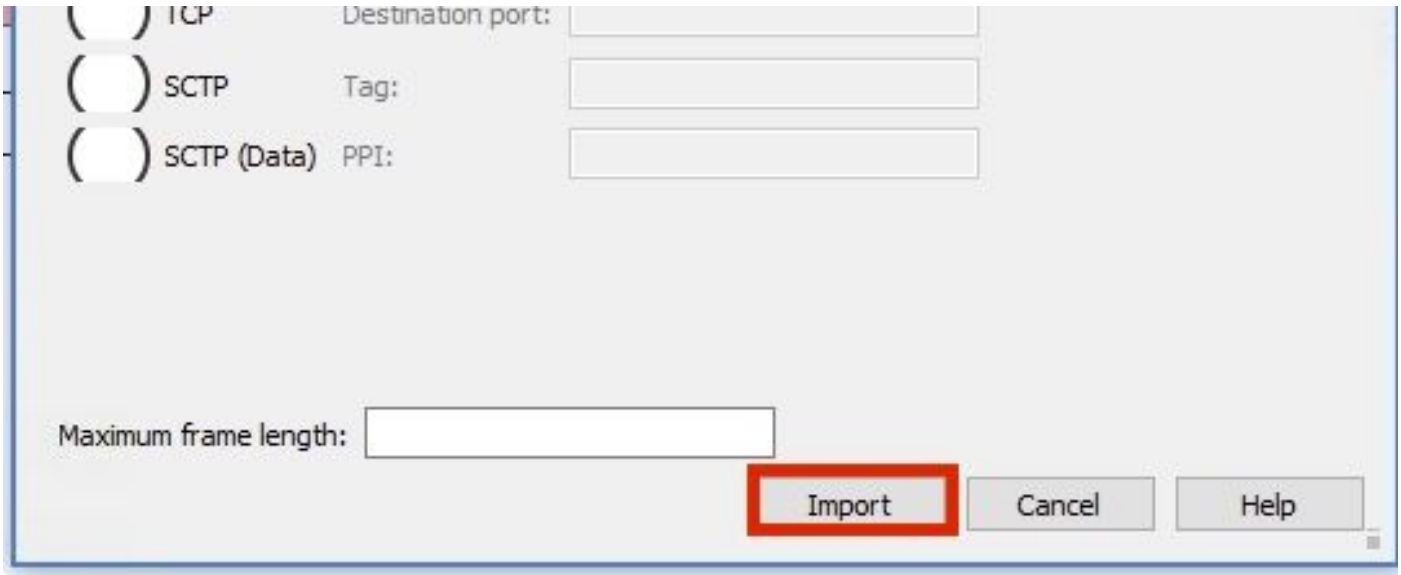
Step 3. Click **Browse.**



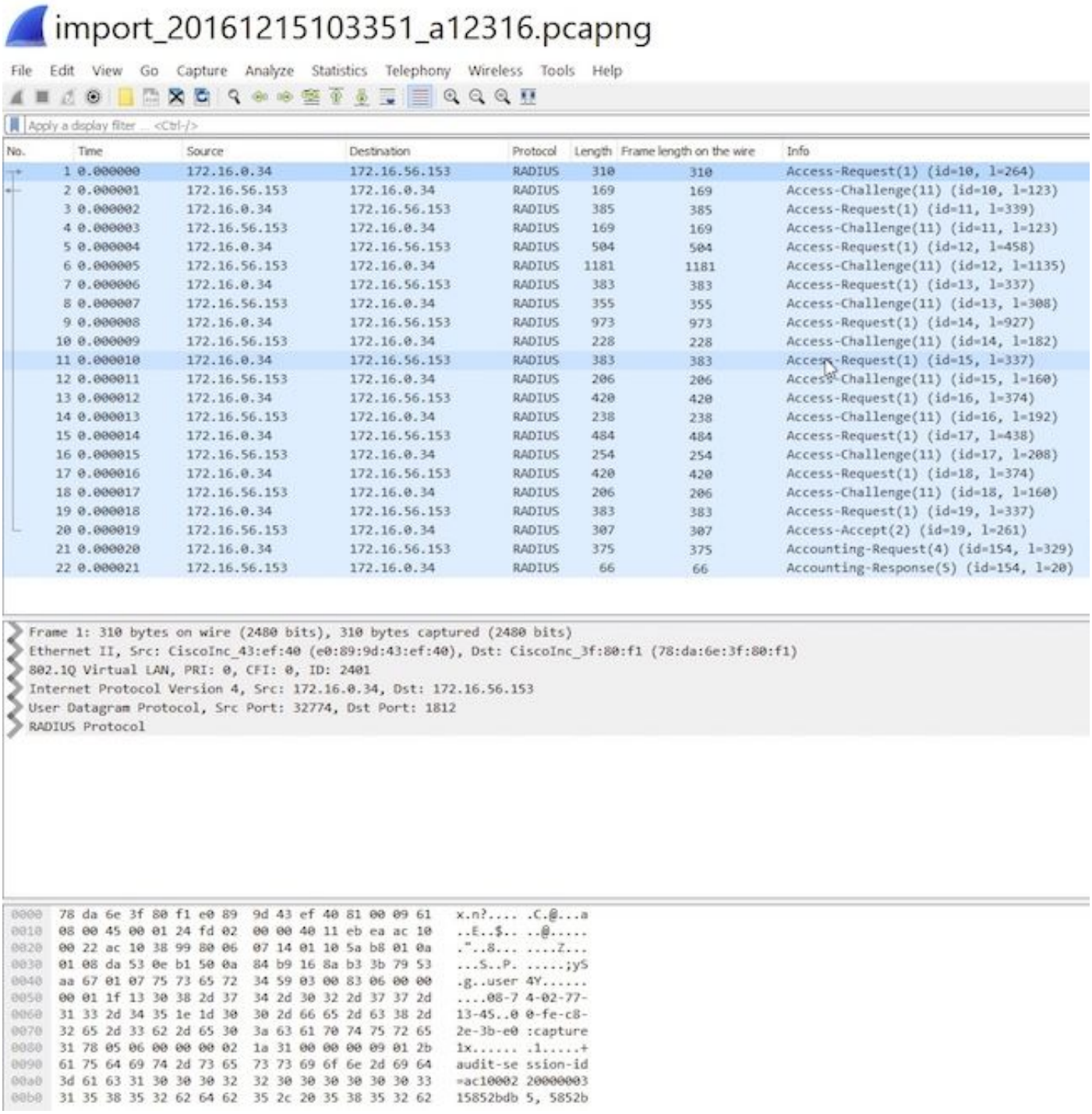
Step 4. Select the text file where you saved the packet logging output.



Step 5. Click **Import.**



Wireshark shows the file as .pcap.



Note: Be aware that the time stamps are not accurate nor the delta time between the frames.

Troubleshoot

There is currently no specific troubleshooting information available for this configuration.

Related Information

- [AP Packet Dump]
- [Fundamentals of 802.11 Wireless Sniffing]
- [Technical Support & Documentation - Cisco Systems]