In HIS name

New algorithm for log4shell detection

Ali Sheikh Attar

★ Decrease the filtering time by narrowing the filtering range to only ldap traffics

NOTE

- If capturing all connections, and then look for *jndi* string among them after capture *ldap* connection, the performance will slightly be improved, compared to regex search every quantum of time (3 seconds),
- We cannot save only ldap connections, with option -Y, it cannot simultaneously filters and writes to file,
- We can save all connections and search for ldap connection which contains the *jdni* string, which would increase the performance again compared to search for regex among all connections,

★ In this implementation we would save all connections and search for ldap connections containing *jndi* string in every quantum of time (3 seconds).

```
"tshark -r merged_filtered_output.pcap -Y 'ldap && tcp contains
\"jndi\"' -w \"filtered_file\""
```

capturing ldap traffics from wireless interface using tshark

5. Display LDAP-specific Information:

If we want to see more details specific to LDAP, you can use a display filter while reading the captured file, it filters the captured .pcap files:

```
tshark -r ldap_capture.pcap -Y ldap
```

6. Real-time Capture and Filtering:

Ifor better performance and filter the captures real-time we can use the following command:

```
sudo tshark -i wlan0 -Y ldap
```

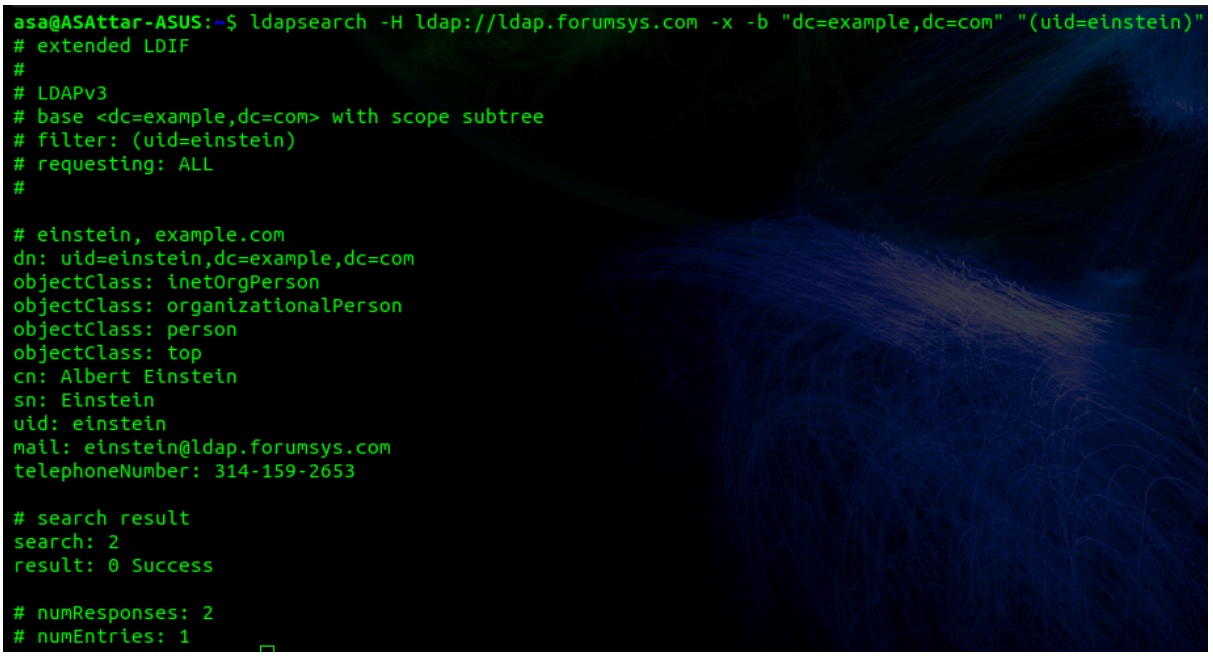
This command captures and displays LDAP traffic in real-time, filtering directly on the protocol.

Use Public LDAP Servers:

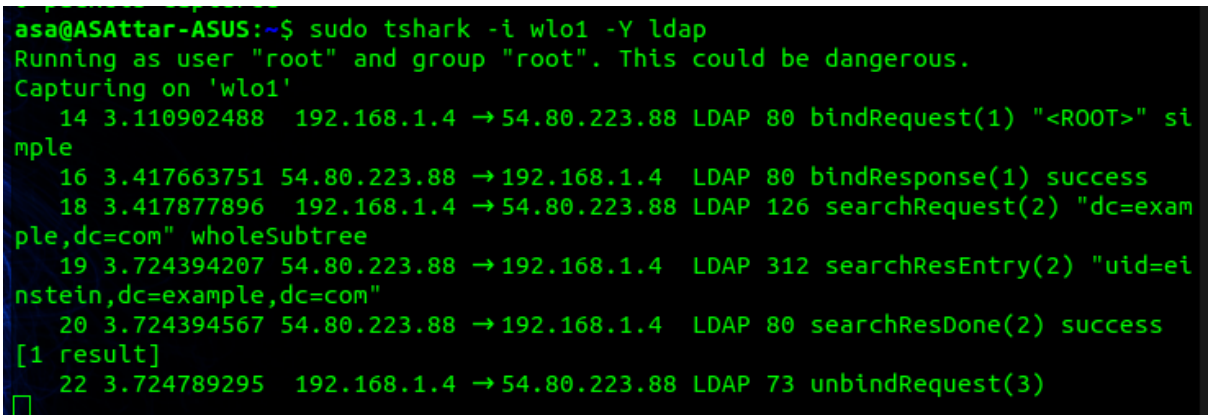
Some organizations maintain publicly accessible LDAP servers for testing purposes. You could look for one of these, but use caution, as publicly available servers might have restricted access or could be unreliable.
Example:

```
ldapsearch -H ldap://ldap.forumsys.com -x -b "dc=example,dc=com"
"(uid=einstein)"
```

This connects to a public test LDAP server.



Result: captures



why dont filtering ldap using -f filter option

Great question! The difference between the `-f` (capture filter) and `-Y` (display filter) options in `tshark` is crucial to understanding why filtering LDAP traffic specifically with the `-f` filter might not work as expected.

Capture Filter (`-f`) vs. Display Filter (`-Y`)

- **Capture Filter (`-f`)**:
  - Applied by the packet capture library (libpcap).
  - Filters packets as they are being captured, meaning only packets that match the filter criteria are saved to the capture file or displayed.
  - Uses a more limited syntax that is generally focused on basic packet properties like IP addresses, ports, protocols, etc.
  - Example: `tcp port 389 or tcp port 636`.
- **Display Filter (`-Y`)**:
  - Applied by `tshark` after the packets have been captured.
  - Filters which packets are displayed or analyzed, but all packets are still captured.
  - Supports a much richer set of filtering options, including filtering by protocol specifics (e.g., LDAP-specific fields).
  - Example: `ldap`.

Why LDAP Filtering with `-f` Might Not Work

The `-f` filter works at a lower level (at the packet capture level), and is limited to filtering on basic packet properties. LDAP operates at the application layer (layer 7 in the OSI model), and although it typically runs over TCP on port 389 or 636, not all traffic on these ports will necessarily be LDAP.

Here are some potential issues with using `-f` to filter LDAP:

1. **Protocol Overhead**: The `-f` filter cannot inspect the payload of the packet to determine if it's truly LDAP. It only filters by properties like port numbers. If an

application uses port 389 or 636 but doesn't actually carry LDAP traffic, those packets will still be captured.

2. **Encrypted Traffic**: If LDAP is running over TLS/SSL (on port 636), the contents of the packet are encrypted. The `-f` filter would capture the packet based on the port number, but it can't confirm whether the encrypted payload is indeed LDAP traffic.
3. **Precision**: The `-f` filter can only capture traffic on specific ports, so any LDAP traffic running on non-standard ports or encapsulated differently would be missed.

Best Practice

- Use `-f` filters to narrow down the capture to relevant packets based on IP addresses, ports, or protocols.
- Use `-Y` filters for more detailed analysis once the packets are captured, especially if you need to inspect application-layer protocols like LDAP.