Wireshark: Ethernet & ARP

Brent Berghmans 1334252

**1. What is the 48-bit Ethernet address of your computer?**

bc:85:56:a0:a2:e9

**2. What is the 48-bit destination address in the Ethernet frame? Is this the Ethernet address of gaia.cs.umass.edu? (Hint: the answer is no). What device has this as its Ethernet address?**

00:17:10:88:48:dd

No. This is the address of the router that my pc uses for access to the internet. We do not know anything about the physical address of gaia.cs.umass.edu, this is handled by the IP layer, the ethernet layer has no knowledge about this.

**3. Give the hexadecimal value for the two-byte Frame type field. What upper layer protocol does this correspond to?**

08 00

This hex value corresponds to the IP protocol.

**4. How many bytes from the very start of the Ethernet frame does the ASCII “G” in “GET” appear in the Ethernet frame?**

The G is the 55th byte from the start.

**5. What is the value of the Ethernet source address? Is this the address of your computer, or of gaia.cs.umass.edu (Hint: the answer is no). What device has this as its Ethernet address?**

00:17:10:88:48:dd

No, this is the address of the router that my pc is connected to. See answer 2.

**6. What is the destination address in the Ethernet frame? Is this the Ethernet address of your computer?**

bc:85:56:a0:a2:e9

Yes this is my ethernet address. I double checked it by using the command ‘getmac’ on windows.

**7. Give the hexadecimal value for the two-byte Frame type field. What upper layer protocol does this correspond to?**

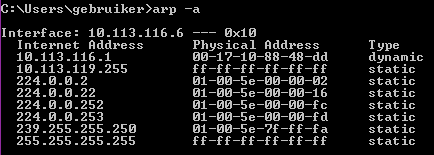
08 00

This corresponds to the IP protocol.

**8. How many bytes from the very start of the Ethernet frame does the ASCII “O” in “OK” (i.e., the HTTP response code) appear in the Ethernet frame?**

The O in “OK” is the 69th byte from the start of the frame.

**9. Write down the contents of your computer’s ARP cache. What is the meaning of each column value?**

****

The first column describes the local ip address, the second column describes the MAC address that belongs to this ip address, and the last column describes the type of the arp entry.

A dynamic entry is added when communicating with another computer or device on the local network.These entries will be deleted after some time of inactivity.

Static entries are entries that have been added manually by the arp command or the OS, and they will stay in the cache until the interface gets restarted or the computer reboots. For instance, calling ipconfig /release and ipconfig /renew will cause a restart to the interface and the cache will be reset.

Both types can be manually deleted by entering the correct command.

**10. What are the hexadecimal values for the source and destination addresses in the Ethernet frame containing the ARP request message?**

Source: bc:85:56:a0:a2:e9

Destination: ff:ff:ff:ff:ff:ff

**11. Give the hexadecimal value for the two-byte Ethernet Frame type field. What upper layer protocol does this correspond to?**

08 06

This corresponds to the ARP protocol.

**12. Download the ARP specification from ftp://ftp.rfc-editor.org/in-notes/std/std37.txt. A readable, detailed discussion of ARP is also at http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html.**

**a) How many bytes from the very beginning of the Ethernet frame does the ARP opcode field begin?**

20 Bytes. The first 14 bytes are part of the ethernet header. The arp message comes after these 14 bytes, but before we get to the opcode we have to pass 2 bytes hardware information, 2 bytes protocol type, 1 byte HLEN and finally 1 byte PLEN. 14 + 2 + 2 + 1 + 1 = 20 bytes.

**b) What is the value of the opcode field within the ARP-payload part of the Ethernet frame in which an ARP request is made?**

Decimal: 1, Hex: 00 01

This indicate that it’s an ARP request.

**c) Does the ARP message contain the IP address of the sender?**

Yes it does, in my case it was “10.113.116.6” which is my local ip address.

**d) Where in the ARP request does the “question” appear – the Ethernet address of the machine whose corresponding IP address is being queried?**

The ARP request contains a target ip of “10.113.116.1” and a target MAC address of “00:00:00:00:00:00”, we are also using the opcode 1 which means we are looking for the MAC address of the host which has 10.113.116.1 as IP address.

**13. Now find the ARP reply that was sent in response to the ARP request.**

**a) How many bytes from the very beginning of the Ethernet frame does the ARP opcode field begin?**

20 Bytes, see 12 a).

**b) What is the value of the opcode field within the ARP-payload part of the Ethernet frame in which an ARP response is made?**

Decimal: 2, Hex: 00 02.

This means ‘reply’.

**c) Where in the ARP message does the “answer” to the earlier ARP request appear – the IP address of the machine having the Ethernet address whose corresponding IP address is being queried?**

The sender’s MAC address is set to 00:17:10:88:48:dd and the sender’s IP address is 10.113.116.1, which means: The host with IP address 10.113.116.1 has the MAC address 00:17:10:88:48:dd.

**14. What are the hexadecimal values for the source and destination addresses in the Ethernet frame containing the ARP reply message?**

Destination: bc:85:56:a0:a2:e9

Source: 00:17:10:88:48:dd

**15. Open the ethernet-ethereal-trace-1 trace file in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip. The first and second ARP packets in this trace correspond to an ARP request sent by the computer running Wireshark, and the ARP reply sent to the computer running Wireshark by the computer with the ARP-requested Ethernet address. But there is yet another computer on this network, as indicated by packet 6 – another ARP request. Why is there no ARP reply (sent in response to the ARP request in packet 6) in the packet trace?**

Whenever an ARP request is sent, all hosts on the network receive this broadcast (because the sender does not know the right address in the first place, hence the request). The host that replies however does know the requester’s MAC and IP address because of the request and thus can send the reply to that address. We weren’t the ones who sent the request so if there even was a reply we wouldn’t have been able to see it. (Our IP address is 192.168.1.105, confirmed by the ARP reply. The sender of the second request has IP address 192.168.1.104).

**EX-1. The arp command: arp -s InetAddr EtherAddr allows you to manually add an entry to the ARP cache that resolves the IP address InetAddr to the physical address EtherAddr. What would happen if, when you manually added an entry, you entered the correct IP address, but the wrong Ethernet address for that remote interface?**

Whenever we’d send something to that IP address, our data would never arrive at the right device (if the “wrong Ethernet address” does exist) or not arrive at all (if it doesn’t exists). To make things worse, every manual added ARP cache entry is a static entry as mentioned in answer 9). This means that this entry does not get refreshed until the interface gets restarted or the computer reboots. (The interface gets restarted automatically on certain occasions like waking up from sleep.). So until this happens the device with this IP address will be unreachable.

**EX-2. What is the default amount of time that an entry remains in your ARP cache before being removed. You can determine this empirically (by monitoring the cache contents) or by looking this up in your operation system documentation. Indicate how/where you determined this value.**

According to <https://technet.microsoft.com/nl-BE/library/cc786759(v=ws.10).aspx>

***Dynamische ARP-cachevermeldingen***

*Dit zijn vermeldingen die automatisch worden toegevoegd en verwijderd tijdens het normale gebruik van TCP/IP-sessies met externe computers. Dynamische vermeldingen verouderen en worden uit de cache verwijderd wanneer ze langer dan 2 minuten niet worden gebruikt. Als een dynamische vermelding binnen 2 minuten opnieuw wordt gebruikt, kan deze maximaal 10 minuten in de cache blijven staan. Als de cache daarna niet wordt vernieuwd met behulp van het ARP-broadcastproces, wordt de vermelding verwijderd.*

How I interpret it:

Dynamic ARP cache entries will be deleted after 2 minutes of inactivity after the TCP/IP session which caused the addition has ended. If however, this entry gets re-used within these 2 minutes, it will stay in the cache for a maximum of 10 minutes. After these 10 minutes, an ARP broadcast will be send out, if there is no reply to this request the entry will get deleted.

However according to the english version of previous said link (<https://technet.microsoft.com/en-us/library/cc758357(v=ws.10).aspx>)

*Each dynamic ARP cache entry has a potential lifetime of 10 minutes. New entries added to the cache are timestamped. If an entry is not reused within 2 minutes of being added, it expires and is removed from the ARP cache. If an entry is used, it receives two more minutes of lifetime. If an entry keeps getting used, it receives an additional two minutes of lifetime up to a maximum lifetime of 10 minutes.*

This is in conflict with my interpretation. So I tested this and after 15 minutes of the first ARP request I did not see any further ARP requests which would indicate that the maximum lifetime is outdated.

So I looked further and found this link <https://support.microsoft.com/en-us/kb/949589>

Unfortunately I use windows 10, which is not in the list down below, so it might have changed again.

Useful snippet:

*ARP caching behavior has been changed in Windows Vista. The TCP/IP stack implementations in Windows Vista comply with RFC4861 (Neighbor Discovery protocol for IP version 6 [Ipv6]) for both the IPv4 and the IPv6 Neighbor Discovery process.*

*The ArpCacheLife and ArpCacheMinReferencedLife registry entries determine how the ARP cache is maintained in Windows XP and in Windows Server 2003. These registry entries no longer apply to Windows Vista.*

*In the new Windows Vista TCP/IP stack implementation, hosts create the neighbor cache entries when there is no matching entry in the neighbor cache. ARP cache entry for IPv4 is an example of a neighbor cache entry. After the entry is successfully created in the neighbor cache, the entry may change to the "Reachable" state if the entry meets certain conditions. If the entry is in the "Reachable" state, Windows Vista TCP/IP hosts do not send ARP requests to the network. Therefore, Windows Vista TCP/IP hosts use the information in the cache. If an entry is not used, and it stays in the "Reachable" state for longer than its "Reachable Time" value, the entry changes to the "Stale" state. If an entry is in the "Stale" state, the Windows Vista TCP/IP host must send an ARP request to reach that destination.*

*The "Reachable Time" value is calculated as follows:*

*Reachable Time = BaseReachable Time × (A random value between MIN\_RANDOM\_FACTOR and MAX\_RANDOM\_FACTOR)*

*RFC provides the following calculated results.*

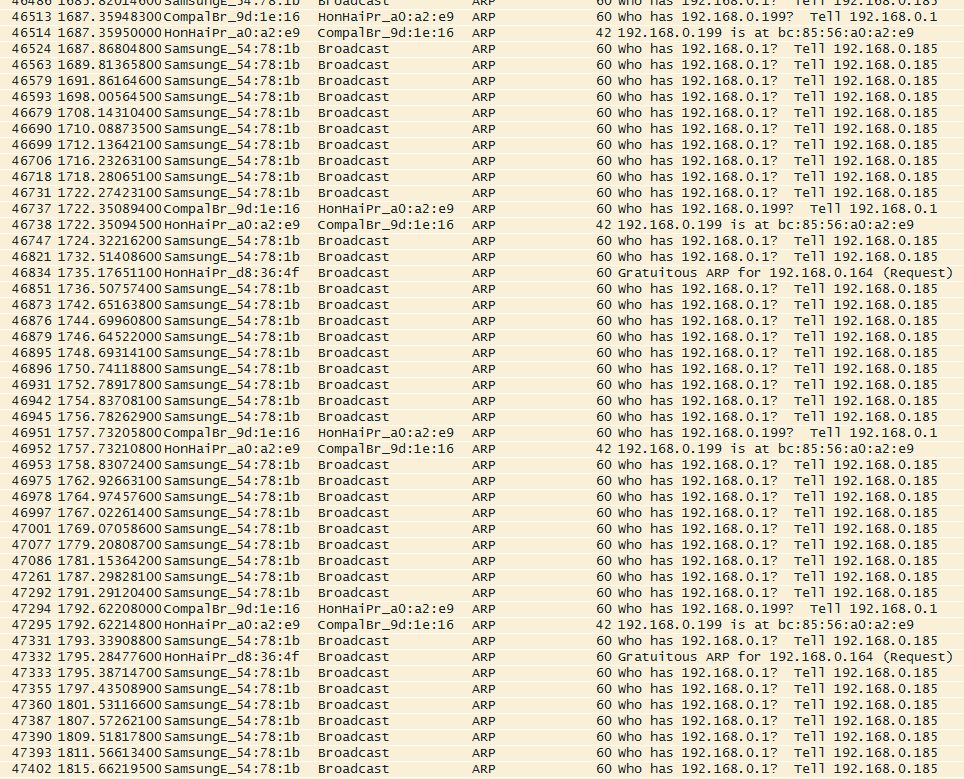
|  |  |
| --- | --- |
| *BaseReachable Time* | *30,000 milliseconds (ms)* |
| *MIN\_RANDOM\_FACTOR* | *0.5* |
| *MAX\_RANDOM\_FACTOR* | *1.5* |

*Therefore, the "Reachable Time" value is somewhere between 15 seconds (30 × 0.5 seconds) and 45 seconds (30 × 1.5 seconds). If an entry is not used for a time between 15 to 45 seconds, it changes to the "Stale" state. Then, the host must send an ARP Request for IPV4 to the network when any IP datagram is sent to that destination.*

**Interesting discovery?**

To determine if the 2 minute timeout was still applicable to windows 10 I thought of a plan: Delete the ARP cache, ping the TV in the room, wait a couple of minutes and look at the cache again to see if it disappeared. I chose the TV in the hope that windows wouldn’t make connections with it on it’s own.

While doing this I noticed that my TV sends out **a lot** of ARP request messages. (trace: smart-tv-bug.pcap)



SamsungE\_54:78:1b is my TV.

Could there be a bug with the ARP caching on Samsung Smart TV’s?