WIKIPEDIA
The Free Encyclopedia

Address Resolution Protocol

The **Address Resolution Protocol** (**ARP**) is a communication protocol used for discovering the link layer address, such as a MAC address, associated with a given internet layer address, typically an IPv4 address. This mapping is a critical function in the Internet protocol suite. ARP was defined in 1982 by RFC 826 (https://datatracker.ietf.org/doc/html/rfc826),[1] which is Internet Standard STD 37.

ARP has been implemented with many combinations of network and data link layer technologies, such as IPv4, Chaosnet, DECnet and Xerox PARC Universal Packet (PUP) using IEEE 802 standards, FDDI, X.25, Frame Relay and Asynchronous Transfer Mode (ATM).

In Internet Protocol Version 6 (IPv6) networks, the functionality of ARP is provided by the Neighbor Discovery Protocol (NDP).

Operating scope

The Address Resolution Protocol is a request-response protocol. Its messages are directly encapsulated by a link layer protocol. It is communicated within the boundaries of a single network, never routed across internetworking nodes.

Packet structure

The Address Resolution Protocol uses a simple message format containing one address resolution request or response. The packets are carried at the data link layer of the underlying network as raw payload. In the case of Ethernet, a `0x0806` EtherType value is used to identify ARP frames.

The size of the ARP message depends on the link layer and network layer address sizes. The message header specifies the types of network in use at each layer as well as the size of addresses of each. The message header is completed with the operation code for request (1) and reply (2). The payload of the packet consists of four addresses, the hardware and protocol address of the sender and receiver hosts.

The principal packet structure of ARP packets is shown in the following table which illustrates the case of IPv4 networks running on Ethernet. In this scenario, the packet has 48-bit fields for the sender hardware address (SHA) and target hardware address (THA), and 32-bit fields for the corresponding sender and target protocol addresses (SPA and TPA). The ARP packet size in this case is 28 bytes.

Hardware type (HTYPE)
: This field specifies the network link protocol type. Example: Ethernet is 1.[2]

Protocol type (PTYPE)
: This field specifies the internetwork protocol for which the ARP request is intended. For IPv4, this has the value `0x0800`. The permitted PTYPE values share a numbering space with those for EtherType.[2][3]

Hardware length (HLEN)
: Length (in octets) of a hardware address. Ethernet address length is 6.

Protocol length (PLEN)

Length (in octets) of internetwork addresses. The internetwork protocol is specified in PTYPE. Example: IPv4 address length is 4.

Operation
Specifies the operation that the sender is performing: 1 for request, 2 for reply.

Sender hardware address (SHA)
Media address of the sender. In an ARP request this field is used to indicate the address of the host sending the request. In an ARP reply this field is used to indicate the address of the host that the request was looking for.

Sender protocol address (SPA)
Internetwork address of the sender.

Target hardware address (THA)
Media address of the intended receiver. In an ARP request this field is ignored. In an ARP reply this field is used to indicate the address of the host that originated the ARP request.

Target protocol address (TPA)
Internetwork address of the intended receiver.

Internet Protocol (IPv4) over Ethernet ARP packet

Octet offset	0	1
0	Hardware type (HTYPE)	
2	Protocol type (PTYPE)	
4	Hardware address length (HLEN)	Protocol address length (PLEN)
6	Operation (OPER)	
8	Sender hardware address (SHA) (first 2 bytes)	
10	(next 2 bytes)	
12	(last 2 bytes)	
14	Sender protocol address (SPA) (first 2 bytes)	
16	(last 2 bytes)	
18	Target hardware address (THA) (first 2 bytes)	
20	(next 2 bytes)	
22	(last 2 bytes)	
24	Target protocol address (TPA) (first 2 bytes)	
26	(last 2 bytes)	

ARP protocol parameter values have been standardized and are maintained by the Internet Assigned Numbers Authority (IANA).[2]

The EtherType for ARP is `0x0806`. This appears in the Ethernet frame header when the payload is an ARP packet and is not to be confused with PTYPE, which appears within this encapsulated ARP packet.

Layering

ARP's placement within the Internet protocol suite and the OSI model may be a matter of confusion or even of dispute. RFC 1122 (https://datatracker.ietf.org/doc/html/rfc1122) mentions ARP within its link layer section without explicitly placing it within that layer.[4] Some older references place ARP in OSI's data link layer[5] while newer editions associate it with the network layer or introduce an intermediate OSI layer 2.5.[6]

Example

Two computers in an office (*Computer 1* and *Computer 2*) are connected to each other in a local area network by Ethernet cables and network switches, with no intervening gateways or routers. *Computer 1* has a packet to send to *Computer 2*. Through DNS, it determines that *Computer 2* has the IP address *192.168.0.55*.

To send the message, it also requires *Computer 2*'s MAC address. First, *Computer 1* uses a cached ARP table to look up *192.168.0.55* for any existing records of *Computer 2*'s MAC address (*00:EB:24:B2:05:AC*). If the MAC address is found, it sends an Ethernet frame containing the IP packet onto the link with the destination address *00:EB:24:B2:05:AC*. If the cache did not produce a result for *192.168.0.55*, *Computer 1* has to send a broadcast ARP request message (destination *FF:FF:FF:FF:FF:FF* MAC address), which is accepted by all computers on the local network, requesting an answer for *192.168.0.55*.

Computer 2 responds with an ARP response message containing its MAC and IP addresses. As part of fielding the request, *Computer 2* may insert an entry for *Computer 1* into its ARP table for future use.

Computer 1 receives and caches the response information in its ARP table and can now send the packet.[7]

ARP probe

An **ARP probe** in IPv4 is an ARP request constructed with the SHA of the probing host, an SPA of all 0s, a THA of all 0s, and a TPA set to the IPv4 address being probed for. If some host on the network regards the IPv4 address (in the TPA) as its own, it will reply to the probe (via the SHA of the probing host) thus informing the probing host of the address conflict. If instead there is no host which regards the IPv4 address as its own, then there will be no reply. When several such probes have been sent, with slight delays, and none receive replies, it can reasonably be expected that no conflict exists. As the original probe packet contains neither a valid SHA/SPA nor a valid THA/TPA pair, there is no risk of any host using the packet to update its cache with problematic data. Before beginning to use an IPv4 address (whether received from manual configuration, DHCP, or some other means), a host implementing this specification must test to see if the address is already in use, by broadcasting ARP probe packets.[8][9]

ARP announcements

ARP may also be used as a simple announcement protocol. This is useful for updating other hosts' mappings of a hardware address when the sender's IP address or MAC address changes. Such an announcement, also called a **gratuitous ARP** (GARP) message, is usually broadcast as an *ARP request* containing the SPA in the target field (TPA=SPA), with THA set to zero. An alternative way is to broadcast an *ARP reply* with the sender's SHA and SPA duplicated in the target fields (TPA=SPA, THA=SHA).

The *ARP request* and *ARP reply* announcements are both standards-based methods,[10][11] but the *ARP request* method is preferred.[12] Some devices may be configured for the use of either of these two types of announcements.[13]

An ARP announcement is not intended to solicit a reply; instead, it updates any cached entries in the ARP tables of other hosts that receive the packet. The operation code in the announcement may be either request or reply; the ARP standard specifies that the opcode is only processed after the ARP table has been updated from the address fields.[14][15][16]

Many operating systems issue an ARP announcement during startup. This helps to resolve problems which would otherwise occur if, for example, a network card was recently changed (changing the IP-address-to-MAC-address mapping) and other hosts still have the old mapping in their ARP caches.

ARP announcements are also used by some network interfaces to provide load balancing for incoming traffic. In a team of network cards, it is used to announce a different MAC address within the team that should receive incoming packets.

ARP announcements can be used in the Zeroconf protocol to allow automatic assignment of a link-local address to an interface where no other IP address configuration is available. The announcements are used to ensure an address chosen by a host is not in use by other hosts on the network link.[17]

This function can be dangerous from a cybersecurity viewpoint since an attacker can obtain information about the other hosts of its subnet to save in their ARP cache (ARP spoofing) an entry where the attacker MAC is associated, for instance, to the IP of the default gateway, thus allowing them to intercept all the traffic to external networks.

ARP mediation

ARP mediation refers to the process of resolving Layer-2 addresses through a virtual private wire service (VPWS) when different resolution protocols are used on the connected circuits, e.g., Ethernet on one end and Frame Relay on the other. In IPv4, each provider edge (PE) device discovers the IP address of the locally attached customer edge (CE) device and distributes that IP address to the corresponding remote PE device. Then each PE device responds to local ARP requests using the IP address of the remote CE device and the hardware address of the local PE device. In IPv6, each PE device discovers the IP address of both local and remote CE devices and then intercepts local Neighbor Discovery (ND) and Inverse Neighbor Discovery (IND) packets and forwards them to the remote PE device.[18]

Inverse ARP and Reverse ARP

Inverse Address Resolution Protocol (**Inverse ARP** or **InARP**) is used to obtain network layer addresses (for example, IP addresses) of other nodes from data link layer (Layer 2) addresses. Since ARP translates layer-3 addresses to layer-2 addresses, InARP may be described as its inverse. In addition, InARP is implemented as a protocol extension to ARP: it uses the same packet format as ARP, but different operation codes.

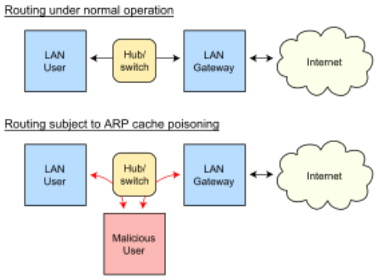
InARP is primarily used in Frame Relay (DLCI) and ATM networks, in which layer-2 addresses of virtual circuits are sometimes obtained from layer-2 signaling, and the corresponding layer-3 addresses must be available before those virtual circuits can be used.[19]

The Reverse Address Resolution Protocol (Reverse ARP or RARP), like InARP, translates layer-2 addresses to layer-3 addresses. However, in InARP the requesting station queries the layer-3 address of another node, whereas RARP is used to obtain the layer-3 address of the requesting station itself for address configuration purposes. RARP is obsolete; it was replaced by BOOTP, which was later superseded by the Dynamic Host Configuration Protocol (DHCP).[20]

ARP spoofing and proxy ARP

Because ARP does not provide methods for authenticating ARP replies on a network, ARP replies can come from systems other than the one with the required Layer 2 address. An ARP *proxy* is a system that answers the ARP request on behalf of another system for which it will forward traffic, normally as a part of the network's design, such as for a dialup internet service. By contrast, in ARP *spoofing* the

answering system, or *spoofer*, replies to a request for another system's address with the aim of intercepting data bound for that system. A malicious user may use ARP spoofing to perform a man-in-the-middle or denial-of-service attack on other users on the network. Various software exists to both detect and perform ARP spoofing attacks, though ARP itself does not provide any methods of protection from such attacks.[21]



A successful ARP spoofing attack allows an attacker to perform a man-in-the-middle attack.

Alternatives

IPv6 uses the Neighbor Discovery Protocol and its extensions such as Secure Neighbor Discovery, rather than ARP.

Computers can maintain lists of known addresses, rather than using an active protocol. In this model, each computer maintains a database of the mapping of Layer 3 addresses (e.g., IP addresses) to Layer 2 addresses (e.g., Ethernet MAC addresses). This data is maintained primarily by interpreting ARP packets from the local network link. Thus, it is often called the *ARP cache*. Since at least the 1980s,[22] networked computers have a utility called *arp* for interrogating or manipulating this database.[23][24][25]

Historically, other methods were used to maintain the mapping between addresses, such as static configuration files,[26] or centrally maintained lists.

ARP stuffing

Embedded systems such as networked cameras[27] and networked power distribution devices,[28] which lack a user interface, can use so-called *ARP stuffing* to make an initial network connection, although this is a misnomer, as ARP is not involved.

ARP stuffing is accomplished as follows:

1. The user's computer has an IP address *stuffed* manually into its address table (normally with the *arp* command with the MAC address taken from a label on the device)
2. The computer sends special packets to the device, typically a ping packet with a non-default size.
3. The device then adopts this IP address
4. The user then communicates with it by telnet or web protocols to complete the configuration.

Such devices typically have a method to disable this process once the device is operating normally, as the capability can make it vulnerable to attack.

Standards documents

- RFC 826 (https://datatracker.ietf.org/doc/html/rfc826) - Ethernet Address Resolution Protocol, Internet Standard STD 37.
- RFC 903 (https://datatracker.ietf.org/doc/html/rfc903) - Reverse Address Resolution Protocol, Internet Standard STD 38.
- RFC 2390 (https://datatracker.ietf.org/doc/html/rfc2390) - Inverse Address Resolution Protocol, draft standard

- RFC 5227 (https://datatracker.ietf.org/doc/html/rfc5227) - IPv4 Address Conflict Detection, proposed standard

See also

- Arping
- Arptables
- Arpwatch
- Bonjour Sleep Proxy
- Cisco HDLC

References

1. David C. Plummer (November 1982). "RFC 826, An Ethernet Address Resolution Protocol -- or -- Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware" (http://tools.ietf.org/html/rfc826). Internet Engineering Task Force, Network Working Group.
2. "Address Resolution Protocol (ARP) Parameters" (https://www.iana.org/assignments/arp-parameters/arp-parameters.xhtml). *www.iana.org*. Retrieved 2018-10-16.
3. RFC 5342 (https://datatracker.ietf.org/doc/html/rfc5342)
4. RFC 1122 (https://datatracker.ietf.org/doc/html/rfc1122)
5. W. Richard Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, Addison Wesley, 1994, ISBN 0-201-63346-9.
6. W. Richard Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, Addison Wesley, 2011, ISBN 0-321-33631-3, page 14
7. Chappell, Laura A.; Tittel, Ed (2007). *Guide to TCP/IP* (Third ed.). Thomson Course Technology. pp. 115–116. ISBN 9781418837556.
8. Cheshire, S. (July 2008). *IPv4 Address Conflict Detection* (https://datatracker.ietf.org/doc/html/rfc5227). Internet Engineering Task Force. doi:10.17487/RFC5227 (https://doi.org/10.17487%2FRFC5227). RFC 5227 (https://datatracker.ietf.org/doc/html/rfc5227).
9. Harmoush, Ed. "ARP Probe and ARP Announcement" (https://www.practicalnetworking.net/series/arp/arp-probe-arp-announcement). *Practical Networking*. PracticalNetworking .net. Retrieved 3 August 2022.
10. Perkins, C. (November 2010). "RFC 5944 - IP Mobility Support for IPv4, Revised" (https://tools.ietf.org/html/rfc5944#section-4.6). Internet Engineering Task Force. "A gratuitous ARP MAY use either an ARP Request or an ARP Reply packet. [...] any node receiving any ARP packet (Request or Reply) MUST update its local ARP cache with the Sender Protocol and Hardware Addresses in the ARP packet [...]"
11. Perkins, C. (October 1996). "RFC 2002 - IP Mobility Support" (https://tools.ietf.org/html/rfc2002#section-4.6). Internet Engineering Task Force.
12. Cheshire, S. (July 2008). "RFC 5227 - IPv4 Address Conflict Detection" (https://tools.ietf.org/html/rfc5227#section-3). Internet Engineering Task Force. "Why Are ARP Announcements Performed Using ARP Request Packets and Not ARP Reply Packets?"
13. "FAQ: The Firewall Does not Update the Address Resolution Protocol Table" (http://support.citrix.com/article/CTX112701). Citrix. 2015-01-16. "[...] garpReply enabled [...] generates ARP packets that [...] are of OPCODE type REPLY, rather than REQUEST."

14. "Gratuitous ARP in DHCP vs. IPv4 ACD Draft" (https://web.archive.org/web/20071012093401/http://www1.ietf.org/mail-archive/web/dhcwg/current/msg03797.html). Archived from the original (http://www1.ietf.org/mail-archive/web/dhcwg/current/msg03797.html) on October 12, 2007.
15. Perkins, Charles E. (October 1996). "RFC 2002 Section 4.6" (http://tools.ietf.org/html/rfc2002#section-4.6).
16. Droms, Ralph (March 1997). "RFC 2131 DHCP – Last lines of Section 4.4.1" (http://tools.ietf.org/html/rfc2131#section-4.4.1).
17. RFC 3927 (https://datatracker.ietf.org/doc/html/rfc3927)
18. Shah, H.; et al. (June 2012). *Address Resolution Protocol (ARP) Mediation for IP Interworking of Layer 2 VPNs* (https://datatracker.ietf.org/doc/html/rfc6575). Internet Engineering Task Force. doi:10.17487/RFC6575 (https://doi.org/10.17487%2FRFC6575). RFC 6575 (https://datatracker.ietf.org/doc/html/rfc6575).
19. T. Bradley; et al. (September 1998). "RFC 2390 - Inverse Address Resolution Protocol" (http://tools.ietf.org/html/rfc2390). Internet Engineering Task Force.
20. Finlayson; Mann; Mogul; Theimer (June 1984). *A Reverse Address Resolution Protocol* (https://datatracker.ietf.org/doc/html/rfc903). Internet Engineering Task Force. doi:10.17487/RFC0903 (https://doi.org/10.17487%2FRFC0903). RFC 903 (https://datatracker.ietf.org/doc/html/rfc903).
21. Steve Gibson (2005-12-11). "ARP Cache Poisoning" (https://www.grc.com/nat/arp.htm). GRC.
22. University of California, Berkeley. "BSD manual page for arp(8C) command" (http://www.freebsd.org/cgi/man.cgi?query=arp&apropos=0&sektion=0&manpath=2.10+BSD&arch=default&format=html). Retrieved 2011-09-28.
23. Canonical. "Ubuntu manual page for arp(8) command" (https://web.archive.org/web/20120316213518/http://manpages.ubuntu.com/manpages/lucid/man8/arp.8.html). Archived from the original (http://manpages.ubuntu.com/manpages/lucid/man8/arp.8.html) on 2012-03-16. Retrieved 2011-09-28.
24. Apple Computer. "Mac OS X manual page for arp(8) command" (https://developer.apple.com/library/mac/#documentation/Darwin/Reference/ManPages/man8/arp.8.html). Retrieved 2011-09-28.
25. Microsoft. "Windows help for arp command" (https://technet.microsoft.com/en-us/library/cc786759%28WS.10%29.aspx). Retrieved 2011-09-28.
26. Sun Microsystems. "SunOS manual page for ethers(5) file" (http://www.freebsd.org/cgi/man.cgi?query=ethers&sektion=5&apropos=0&manpath=SunOS+4.1.3). Retrieved 2011-09-28.
27. Axis Communication. "Axis P13 Network Camera Series Installation Guide" (http://www.axis.com/files/manuals/ig_p13Series_38731_en_1006.pdf) (PDF). Retrieved 2011-09-28.
28. American Power Corporation. "Switched Rack Power Distribution Unit Installation and Quick Start Manual" (https://web.archive.org/web/20111125012617/http://www.apcmedia.com/salestools/ASTE-6Z6K56_R0_EN.pdf) (PDF). Archived from the original (http://www.apcmedia.com/salestools/ASTE-6Z6K56_R0_EN.pdf) (PDF) on 2011-11-25. Retrieved 2011-09-28.

External links

- "ARP Sequence Diagram (pdf)" (https://web.archive.org/web/20210301060206/http://www.eventhelix.com/RealtimeMantra/Networking/Arp.pdf) (PDF). Archived from the original (http://www.eventhelix.com/RealtimeMantra/Networking/Arp.pdf) (PDF) on 2021-03-01.
- Gratuitous ARP (https://wiki.wireshark.org/Gratuitous_ARP)
- Information and sample capture from Wireshark (https://gitlab.com/wireshark/wireshark/-/wikis/AddressResolutionProtocol)
- ARP-SK ARP traffic generation tools (https://web.archive.org/web/20090903074149/http://sid.rstack.org/arp-sk/)