A black background with white text

Description automatically generatedStep-by-step HTTP request

You want to access the Raspberry Pi Foundation website, so you type “raspberrypi.org” into the address bar of your browser. Let’s follow the journey of this request as it travels from your web browser and across the internet to the web server application at the Raspberry Pi Foundation.



|  |  |
| --- | --- |
| Each different type of application uses one or more different protocols. These protocols are implemented in software as subprograms, procedures, or subroutines.  Your web browser (the client) will use the **Hypertext Transfer Protocol** (HTTP) to communicate with the application layer on the Raspberry Pi Foundation’s web server.  The application layer contains a whole suite of different protocols designed to encode/decode messages into a form that is understood by the sender and the recipient devices. |  |
| At the transport layer, the two main protocols that are used are **Transmission Control Protocol (TCP)** and **User Datagram Protocol (UDP).**  Web browsing requires complete (reliable) data, so uses **TCP**. The data from the application layer, i.e. your HTTP request, is broken down and formatted into one or more **segments**.  Each segment is given a header that contains a **sequence number** so that TCP can keep track of the segments**.** A **checksum** is calculated and added to the header to help with error detection. |  |
| The internet layer (sometimes called the network layer) receives the TCP segments from the transport layer. It takes these segments and encapsulates them into **IP packets.**  Each **IP packet** is given a header containing data to assist with delivery. The header will contain the IP address of the packet source and the IP address of its destination. This is so that the packets can be routed to the correct destination, i.e. the Raspberry Pi Foundation web server. |  |
| The link layer (sometimes called the data link layer) enables the **physical transfer** of the packets.  Your device may have many technologies to connect to a network such as **Ethernet**, **Wi-Fi**,or **Bluetooth**. Once the link has been chosen, the appropriate **link layer protocol** is selected.  For example, if your device is connected to a network using Ethernet, the IP packets from the internet layer are further encapsulated in **Ethernet frames** which are designed to be transmitted across a local network. |  |
| An IP packet can be encapsulated and decapsulated several times as it traverses the internet.  Each time a packet encounters a new link, it will be encapsulated in a different frame suitable for that link technology.  Each frame will be given the address of the **next node**. This address is the media access control address or **MAC address** of that node. |  |
| Eventually, the packets containing the HTTP request will reach their final destination, i.e. the Raspberry Pi Foundation web server.  Each frame, as it arrives, will be decapsulated and the IP packets will be passed from the link layer to the internet layer. |  |
| At the internet layer on the web server, each IP packet will be decapsulated and the TCP segment will be passed to the transport layer. |  |
| At the transport layer on the web server, the segments will be checked and put into sequence (if they arrived out of order).  Each segment is acknowledged and any missing segments will be retransmitted.  The complete HTTP request is passed to the application layer. |  |
| At the application layer on the web server, the request for the webpage is received.  The web server will respond to the request by getting all of the data needed for the webpage.  This data is then formatted as an HTTP response and passed down to the transport layer to start its journey back to the client that requested the page. |  |
| Eventually the data will be received back at the application layer on the client and displayed to the user. |  |