

Basics of Name Resolution and Internal Network Name Resolution

This article is reserved for the next person I mentor in technology.

Why Do We Use Name Resolution

... (link to Basics of IP Address article)

Bonjour for Name Resolution

Bonjour for Ubuntu,

```
sudo apt-get install avahi-daemon
```

Once this is installed, you may access your machine by adding .local to the machine's host name. For example if the hostname is **plex**, it can be access as **plex.local**.

Quick Summary

The key to understanding how data is transmitted on to the internet and how text files populate information onto the web we first have to start with understanding the basics.

1. The user (being you) makes a request through your computer to find a particular site (ie. michaelerik.duckdns.org)
2. From your computer the request object is sent directly to the webserver.

Webserver: is a program that served the files that form web pages to the users, in response to their request, which is then forwarded by their computers.

1. This request contains important information for the webserver to interpret and send back the correct response object
2. The request object contains:
 - a. Source ip
 - b. Page request (hostname)
 - c. Path
 - d. Browser version
 - e. Os server
 - f. Key value pair
 - g. Port
 - h. Protocol (ex. http or https)
3. Once the information is received the webserver sends back a response object which contains:
 - a. Webserver IP
 - b. Page Url
 - c. Cookies
4. This information is retrieved and populated on your screen

At the most basic level possible, the following diagram shows the steps that brought that page to your screen.

Introduction- Resolving DNS Process

- 1) When using the internet most people connect to websites or other internet servers by connecting to a domain name, as in "[michaelerik.duck](https://michaelerik.duckdns.org)

[dns.org](https://www.dns.org)". Internet applications, though, do not communicate via domain names, but rather uses IP address such as 10.66.12.104.

IP Address – Every machine on the Internet has a unique IP address. A server has a static IP address that does not change very often. An IP is a unique string of numbers separated by periods that identifies each computer using the Internet to communicate over a network, we can also refer to it as our postal code.

Example of this is: 10.66.12.104

how to find an IP address on my computer	how to find an IP address on from command prompt
<ul style="list-style-type: none">• control panel• network and sharing• click local area connection• details• IPv4	<ul style="list-style-type: none">• Press the Windows Start key to open the Start screen.• Type cmd and press Enter to launch the command prompt. ...• Type ipconfig /all at the command prompt to check the network card settings.• Scroll up till you see (see picture)

When you enter a URL into your Web browser, the user's local host file is first checked to resolve the DNS to an IP address. If this does not happen, it then moves to the second checkpoint.

2) When you type a domain name in your program that you wish to connect to, your application must first convert it to an IP address that it will use to connect to. The way these hostnames are resolved to their mapped IP address is called Domain Name Resolution (DNS). Think of it like our computers GPS for the internet.

Computers and other network devices on the Internet use an IP address to route your request to the site you're trying to reach. This is similar to dialing a phone number to connect to the person you're trying to call. Thanks to DNS, though, you don't have to keep your own address book of IP addresses. Instead, you just connect through a **domain name resolution**, also called a **DNS server** or **name server**, which manages a massive database that maps domain names to IP addresses

When you enter a URL into your Web browser, your DNS server uses its resources to resolve the name into the IP address for the appropriate Web server. A DNS is the task of converting domain names to their corresponding IP address. This is all done behind the scenes and all operating systems have the below default order for domain name resolution.

1. (OS) – Host Files: (which resides on your local computer/device) **NOTE: This has already happened and if not resolved the second checkpoint happens.**
2. (Router) – DNS Gateway: can always override DNS to where it needs to be (ex. Rogers modem)
3. (Internet) – Registrar: (example go daddy) **NOTE: This happens if not resolved in 1 or 2.**

The browser communicated with a [name server](#) to translate the server name "[michaelerik.duckdns.org](#)" nto an **IP Address**, which it uses to connect to the server machine. The browser then formed a connection to the server at that IP address on **port 80**.

Your browser formed a connection to a Web server, requested a page and received it.

The browser broke the URL into three parts:

- The protocol ("http")
- The server name ("[www.howstuffworks.com](#)")
- The file name ("web-server.htm")

The server then sent the [HTML text](#) for the Web page to the browser.

The browser read the [HTML tags](#) and formatted the page onto your screen.

(Browser Cache- keeps a memory of what you have looked at, it's helpful because you don't need to download again and it's a default setting in the computer that captures what you have looked at. So as you start to type a few characterises pages will automatically start to populate until you find what you're looking for. Sometimes you have to force cache request (control r) or delete cache. To be able to see updated page content.)

3) Any server machine makes its services available to the Internet using numbered **ports**, one for each service that is available on the server. For example, if a server machine is running a Web server and an FTP server, the Web server would typically be available on port 80, and the FTP server would be available on port 21. Clients connect to a service at a specific IP address and on a specific port.

Here are some common port numbers:

- tp 21
- telnet 23
- smtp 25 (Simple Mail Transfer, meaning e-mail)

- time 37
- nameserver 53
- gopher 70
- finger 79
- www 80

In this particular case, we are making a general website search which will be accessed via port 80. A "request object" or "request header" is sent from the user (which happens "behind the scenes" and requires no effort from the user) to the server that hosts the files, images, content, etc. that the user is requesting to access.

Here are the common pieces of information sent in a request header:

Browser Type: Chrome (user's browser)
Source IP: 124.63.145.113 (user's IP)
OS: Windows (user's OS)
Domain Request: michaelerik.duckdns.org
Key Value Pair: Public

4) What is a web server?

A Web [server](#) is a computer system that processes request using [HTTP](#) (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. All computers that host Web sites must have Web server programs. Most commonly used web server is Apache. Webservers are usually set up in facilities called Data Centers, and operated by hosting companies. The purpose of these computers is to store websites so that people can visit them on the internet. The files stored on web servers are read by browsers (such as Firefox, Safari, Chrome, or Internet Explorer) which converts these files into images and text for you to view. Your browser communicates with web servers to bring you information from the internet.

How you would add styling to your html:

```
<p style="font-family:courier;">This is Michael at <b>10 minutes</b> old. Born just before midnight, our little Sagitarius night owl tuckered mama out.</p>
```

To add a link in html:

Specify the target in the .

Then add the text that should work as a link.

Finally add an tag to indicate where the link ends

Example:

```
<a href= "text.html"> my first link</a>!
```

5) After the server has received the request header information and has retrieved the requested website content, the server then sends a "response object" or "response header" along with the content. The response essentially validates the user's request.

Here are the common pieces of information sent in a response header:

Browser Type: Chrome
Server IP: 174.117.163.194
Domain: tin.homeip.net
OS: ubuntu
Cookies: Optional
Content Type: text/HTML

6) Public and Private Keys

1. Using Keys to Upload Content to a Host

To be able to host content on a server, you must first create both a public and private key. The private key is never shared and resides on the user's machine. The public key, however, is shared with the server host and resides on the server. When the user makes a request to connect with the host server with the intent to upload website content, the users sends the public key information that gets validated by the server.

2. Using Keys for Transactional Activity (example: banking)

Process



HTML Basics

The Web is basically a system of formatted documents in a language called HTML (HyperText Markup Language) that supports links to other documents, links, as well as graphics, audio, and video files

HTML (HyperText Markup Language)

The interesting thing about HTML is it is a system for tagging text files to achieve font, color, graphics and hyperlinks on web pages.

For instance

```
<h1> Cassandra Edwards HTML </h1>
```

You will write the HTML document on Notepad. When you are finished creating the HTML document, you'll then open the document in a browser. The browser then takes that information and interprets the HTML for you and displays it to the web page.

Now what you use to set certain sections apart as heading, subtitles, bold text, underlined text etc is a series of what is called 'tags'

Think of tagging like a structure of an essay.

You will have a heading for instance.

```
<html>
```

```
<head>
```

```
<h1> Cassandra Edwards </h1>
```

```
<p> Scotiabank has a requirements that all employees and contractors complete mandatory learning, on an annual basis, covering new and updated critical information regarding compliance, security and risk </p>
```

We also learned to attach a link

```
<html>
```

```
<head>
```

```
<h1> Cassandra Edwards </h1>
```

```
</head>
```

```
<body>
```

```
<p> April <i> 24 </i> 2017 </p>
```

</body>

</html>

We also learned to attach a pic