

# *Round-robin DNS*

**Round-robin DNS** is a technique of [load distribution](#), [load balancing](#), or [fault-tolerance](#) provisioning multiple, redundant [Internet Protocol](#) service hosts, e.g., [Web server](#), [FTP servers](#), by managing the [Domain Name System](#)'s (DNS) responses to address requests from client computers according to an appropriate statistical model.<sup>[1]</sup>

In its simplest implementation, round-robin DNS works by responding to DNS requests not only with a single potential [IP address](#), but with a list of potential IP addresses corresponding to several servers that host identical services.<sup>[2][3]</sup> The order in which IP addresses from the list are returned is the basis for the term *round robin*. With each DNS response, the IP address sequence in the list is [permuted](#).<sup>[4]</sup> Traditionally, IP clients initially attempt connections with the first address returned from a DNS query,<sup>[5]</sup> so that on different connection attempts, clients would receive service from different providers, thus distributing the overall load among servers.

Some resolvers attempt to re-order the list to give priority to numerically "closer" networks. This behaviour was [standardized](#) during the definition of IPv6, and has been blamed for defeating round-robin load-balancing.<sup>[6][7][8]</sup> Some desktop clients do try alternate addresses after a connection timeout of up to 30 seconds.<sup>[2]</sup>

Round-robin DNS is often used to load balance requests among a number of [Web servers](#). For example, a company has one domain name and three identical copies of the same web site residing on three servers with three IP addresses. The DNS server will be set up so that domain name has multiple A records, one for each IP address. When one user accesses the home page it will be sent to the first IP address. The second user who accesses the home page will be sent

to the next IP address, and the third user will be sent to the third IP address. In each case, once the IP address is given out, it goes to the end of the list. The fourth user, therefore, will be sent to the first IP address, and so forth.

A round-robin DNS name is, on rare occasions, referred to as a "rotor" due to the rotation among alternative A records.

## Drawbacks

---

Although easy to implement, round-robin DNS has a number of drawbacks, such as those arising from record caching in the DNS hierarchy itself, as well as client-side address caching and reuse, the combination of which can be difficult to manage. Round-robin DNS should not solely be relied upon for service availability. If a service at one of the addresses in the list fails, the DNS will continue to hand out that address and clients will still attempt to reach the inoperable service.

Round-robin DNS may not be the best choice for [load balancing](#) on its own, since it merely alternates the order of the address records each time a name server is queried. Because it does not take transaction time, server load, and network congestion into consideration, it works best for services with a large number of uniformly distributed connections to servers of equivalent capacity. Otherwise, it just does [load distribution](#).<sup>[9]</sup>

Methods exist to overcome such limitations. For example, modified DNS servers (such as [Ibnamed](#)<sup>[10]</sup>) can routinely poll mirrored servers for availability and load factor. If a server does not reply as required, the server can be temporarily removed from the DNS pool, until it reports that it is once again operating within specs.

## References

---

1. ["Round-Robin DNS"](https://www.cloudflare.com/learning/dns/glossary/round-robin-dns/) (<https://www.cloudflare.com/learning/dns/glossary/round-robin-dns/>) . Cloudflare. Cloudflare, Inc. Retrieved 16 August 2019.
2. Feenberg, Daniel (July 25, 2016). ["DNS round robin for web server failover"](http://www.nber.org/sys-admin/dns-failover.html) (<http://www.nber.org/sys-admin/dns-failover.html>) . Retrieved January 10, 2020.
3. ["What is the difference among a multivalued answer routing policy and a simple routing policy?"](https://aws.amazon.com/premiumsupport/knowledge-center/multivalued-versus-simple-policies/) (<https://aws.amazon.com/premiumsupport/knowledge-center/multivalued-versus-simple-policies/>) . Amazon Web Services, Inc. August 3, 2018. Retrieved January 10, 2020.

4. Liska, A.; Stowe, G. (2016). "Understanding DNS". *DNS Security*. p. 19.
5. Pan, Jianping; Hou, Y. Thomas; Li, Bo (2003). "An overview of DNS-based server selections in content distribution networks". *Computer Networks*. **43** (6): 695–711. doi:10.1016/S1389-1286(03)00293-7 (<https://doi.org/10.1016%2FS1389-1286%2803%2900293-7>) .<sup>6</sup>
6. "getaddrinfo with round robin DNS and happy eyeballs" (<https://daniel.haxx.se/blog/2012/01/03/getaddrinfo-with-round-robin-dns-and-happy-eyeballs/>) .
7. "RFC 6724 - Default Address Selection for Internet Protocol Version 6 (IPv6)" (<https://tools.ietf.org/html/rfc6724>) .
8. "Rule 9 for IPv4 caused serious operational problems ([the Debian Linux] ftp sites failed!" (<https://lists.debian.org/debian-ctte/2007/09/msg00067.html>) .
9. Brisco, Thomas P. (April 1995). "RFC 1794 - DNS Support for Load Balancing" (<https://tools.ietf.org/html/rfc1794>) . Retrieved 24 August 2019.
10. *lbname*d (<http://www.stanford.edu/~riepel/lbname/>) , a load-balanced DNS server implemented in the Perl programming language

Retrieved from

["https://en.wikipedia.org/w/index.php?title=Round-robin\\_DNS&oldid=1046790621"](https://en.wikipedia.org/w/index.php?title=Round-robin_DNS&oldid=1046790621)

WIKIPEDIA

---