Overview of Traffic Manager

06/

Microsoft Azure Traffic Manager allows you to control the distribution of user traffic for service endpoints in different datacenters. Service endpoints supported by Traffic Manager include Azure VMs, Web Apps, and cloud services. You can also use Traffic Manager with external, non-Azure endpoints.

Traffic Manager uses the Domain Name System (DNS) to direct client requests to the most appropriate endpoint based on a traffic-routing method and the health of the endpoints. Traffic Manager provides a range of [traffic-routing methods](https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-routing-methods) and [endpoint monitoring options](https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-monitoring) to suit different application needs and automatic failover models. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

Traffic Manager benefits

Traffic Manager can help you:

* **Improve availability of critical applications**

Traffic Manager delivers high availability for your applications by monitoring your endpoints and providing automatic failover when an endpoint goes down.

* **Improve responsiveness for high-performance applications**

Azure allows you to run cloud services or websites in datacenters located around the world. Traffic Manager improves application responsiveness by directing traffic to the endpoint with the lowest network latency for the client.

* **Perform service maintenance without downtime**

You can perform planned maintenance operations on your applications without downtime. Traffic Manager directs traffic to alternative endpoints while the maintenance is in progress.

* **Combine on-premises and Cloud-based applications**

Traffic Manager supports external, non-Azure endpoints enabling it to be used with hybrid cloud and on-premises deployments, including the "burst-to-cloud," "migrate-to-cloud," and "failover-to-cloud" scenarios.

* **Distribute traffic for large, complex deployments**

Using [nested Traffic Manager profiles](https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-nested-profiles), traffic-routing methods can be combined to create sophisticated and flexible rules to support the needs of larger, more complex deployments.

How Traffic Manager works

Azure Traffic Manager enables you to control the distribution of traffic across your application endpoints. An endpoint is any Internet-facing service hosted inside or outside of Azure.

Traffic Manager provides two key benefits:

1. Distribution of traffic according to one of several [traffic-routing methods](https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-routing-methods)
2. [Continuous monitoring of endpoint health](https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-monitoring) and automatic failover when endpoints fail

When a client attempts to connect to a service, it must first resolve the DNS name of the service to an IP address. The client then connects to that IP address to access the service.

**The most important point to understand is that Traffic Manager works at the DNS level.**Traffic Manager uses DNS to direct clients to specific service endpoints based on the rules of the traffic-routing method. Clients connect to the selected endpoint **directly**. Traffic Manager is not a proxy or a gateway. Traffic Manager does not see the traffic passing between the client and the service.

Traffic Manager example

Contoso Corp have developed a new partner portal. The URL for this portal is <https://partners.contoso.com/login.aspx>. The application is hosted in three regions of Azure. To improve availability and maximize global performance, they use Traffic Manager to distribute client traffic to the closest available endpoint.

To achieve this configuration, they complete the following steps:

1. Deploy three instances of their service. The DNS names of these deployments are 'contoso-us.cloudapp.net', 'contoso-eu.cloudapp.net', and 'contoso-asia.cloudapp.net'.
2. Create a Traffic Manager profile, named 'contoso.trafficmanager.net', and configure it to use the 'Performance' traffic-routing method across the three endpoints.
3. Configure their vanity domain name, 'partners.contoso.com', to point to 'contoso.trafficmanager.net', using a DNS CNAME record.

1

Note

When using a vanity domain with Azure Traffic Manager, you must use a CNAME to point your vanity domain name to your Traffic Manager domain name. DNS standards do not allow you to create a CNAME at the 'apex' (or root) of a domain. Thus you cannot create a CNAME for 'contoso.com' (sometimes called a 'naked' domain). You can only create a CNAME for a domain under 'contoso.com', such as 'www.contoso.com'. To work around this limitation, we recommend using a simple HTTP redirect to direct requests for 'contoso.com' to an alternative name such as 'www.contoso.com'.

How clients connect using Traffic Manager

Continuing from the previous example, when a client requests the page <https://partners.contoso.com/login.aspx>, the client performs the following steps to resolve the DNS name and establish a connection:

1

1. The client sends a DNS query to its configured recursive DNS service to resolve the name 'partners.contoso.com'. A recursive DNS service, sometimes called a 'local DNS' service, does not host DNS domains directly. Rather, the client off-loads the work of contacting the various authoritative DNS services across the Internet needed to resolve a DNS name.
2. To resolve the DNS name, the recursive DNS service finds the name servers for the 'contoso.com' domain. It then contacts those name servers to request the 'partners.contoso.com' DNS record. The contoso.com DNS servers return the CNAME record that points to contoso.trafficmanager.net.
3. Next, the recursive DNS service finds the name servers for the 'trafficmanager.net' domain, which are provided by the Azure Traffic Manager service. It then sends a request for the 'contoso.trafficmanager.net' DNS record to those DNS servers.
4. The Traffic Manager name servers receive the request. They choose an endpoint based on:
   * The configured state of each endpoint (disabled endpoints are not returned)
   * The current health of each endpoint, as determined by the Traffic Manager health checks. For more information, see [Traffic Manager Endpoint Monitoring](https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-monitoring).
   * The chosen traffic-routing method. For more information, see [Traffic Manager Routing Methods](https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-routing-methods).
5. The chosen endpoint is returned as another DNS CNAME record. In this case, let us suppose contoso-us.cloudapp.net is returned.
6. Next, the recursive DNS service finds the name servers for the 'cloudapp.net' domain. It contacts those name servers to request the 'contoso-us.cloudapp.net' DNS record. A DNS 'A' record containing the IP address of the US-based service endpoint is returned.
7. The recursive DNS service consolidates the results and returns a single DNS response to the client.
8. The client receives the DNS results and connects to the given IP address. The client connects to the application service endpoint directly, not through Traffic Manager. Since it is an HTTPS endpoint, the client performs the necessary SSL/TLS handshake, and then makes an HTTP GET request for the '/login.aspx' page.

The recursive DNS service caches the DNS responses it receives. The DNS resolver on the client device also caches the result. Caching enables subsequent DNS queries to be answered more quickly by using data from the cache rather than querying other name servers. The duration of the cache is determined by the 'time-to-live' (TTL) property of each DNS record. Shorter values result in faster cache expiry and thus more round-trips to the Traffic Manager name servers. Longer values mean that it can take longer to direct traffic away from a failed endpoint. Traffic Manager allows you to configure the TTL used in Traffic Manager DNS responses to be as low as 0 seconds and as high as 2,147,483,647 seconds (the maximum range compliant with [RFC-1035](https://www.ietf.org/rfc/rfc1035.txt)), enabling you to choose the value that best balances the needs of your application.