Highly scalable and secure WordPress website

09/18/2018 • 5 minutes to read • Contributors

In this article

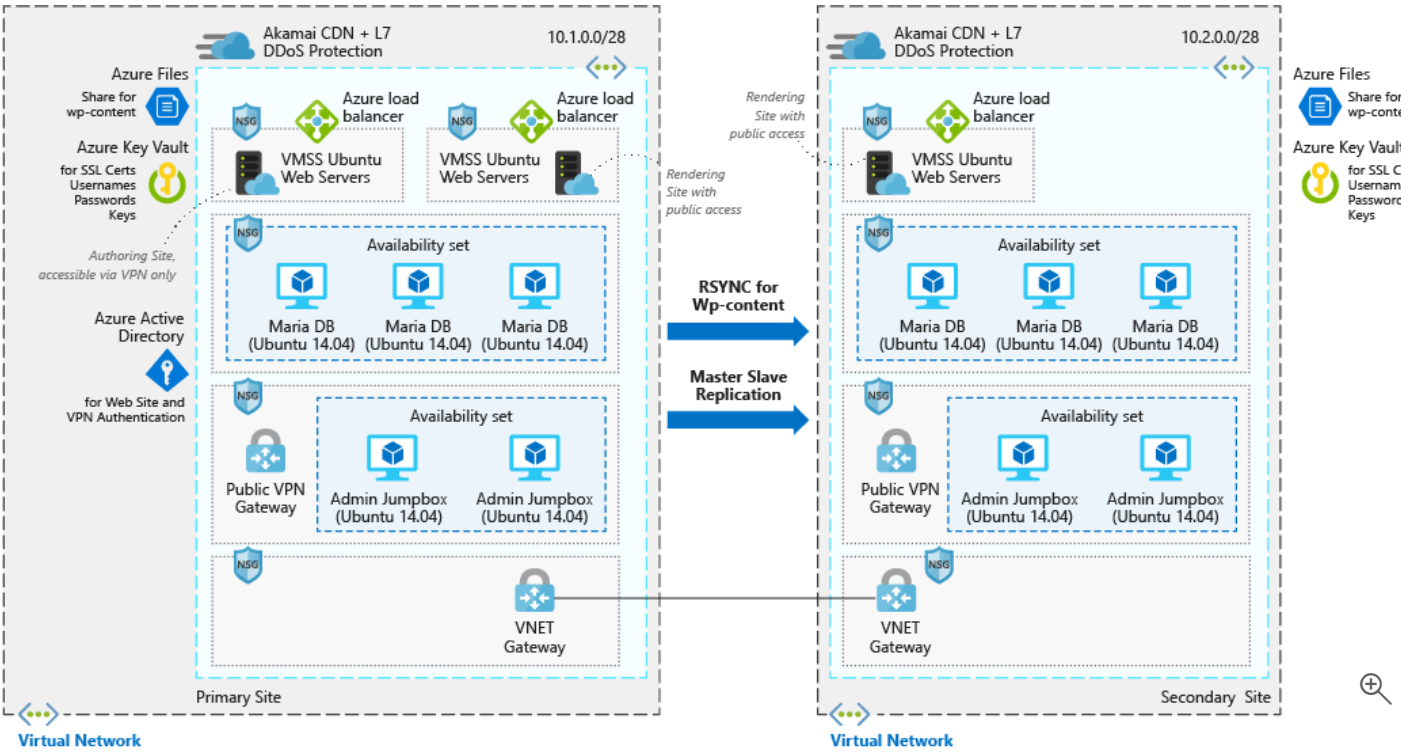
This example scenario is applicable to companies that need a highly scalable and secure installation of WordPress. This scenario is based on a deployment that was used for a large convention and was successfully able to scale to meet the spike traffic that sessions drove to the site.

Relevant use cases

Other relevant use cases include:

- Media events that cause traffic surges.
- Blogs that use WordPress as their content management system.
- Business or e-commerce websites that use WordPress.
- Web sites built using other content management systems.

Architecture



This scenario covers a scalable and secure installation of WordPress that uses Ubuntu web servers and MariaDB. There are two distinct data flows in this scenario the first is users access the website:

1. Users access the front-end website through a CDN.
2. The CDN uses an Azure load balancer as the origin, and pulls any data that isn't cached from there.
3. The Azure load balancer distributes requests to the virtual machine scale sets of web servers.

4. The WordPress application pulls any dynamic information out of the Maria DB clusters, all static content is hosted in Azure Files.
5. SSL keys are stored Azure Key Vault.

The second workflow is how authors contribute new content:

1. Authors connect securely to the public VPN gateway.
2. VPN authentication information is stored in Azure Active Directory.
3. A connection is then established to the Admin jump boxes.
4. From the admin jump box, the author is then able to connect to the Azure load balancer for the authoring cluster.
5. The Azure load balancer distributes traffic to the virtual machine scale sets of web servers that have write access to the Maria DB cluster.
6. New static content is uploaded to Azure files and dynamic content is written into the Maria DB cluster.
7. These changes are then replicated to the alternate region via rsync or master/slave replication.

Components

- Azure Content Delivery Network (CDN) is a distributed network of servers that efficiently delivers web content to users. CDNs minimize latency by storing cached content on edge servers in point-of-presence locations near to end users.
- Virtual networks allow resources such as VMs to securely communicate with each other, the Internet, and on-premises networks. Virtual networks provide isolation and segmentation, filter and route traffic, and allow connection between locations. The two networks are connected via Vnet peering.
- Network security groups contain a list of security rules that allow or deny inbound or outbound network traffic based on source or destination IP address, port, and protocol. The virtual networks in this scenario are secured with network security group rules that restrict the flow of traffic between the application components.
- Load balancers distribute inbound traffic according to rules and health probes. A load balancer provides low latency and high throughput, and scales up to millions of flows for all TCP and UDP applications. A load balancer is used in this scenario to distribute traffic from the content deliver network to the front-end web servers.
- Virtual machine scale sets let you create and manage a group of identical load-balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Two separate virtual machine scale sets are used in this scenario - one for the front-end web-servers serving content, and one for the front-end webservers used to author new content.
- Azure Files provides a fully-managed file share in the cloud that hosts all of the WordPress content in this scenario, so that all of the VMs have access to the data.
- Azure Key Vault is used to store and tightly control access to passwords, certificates, and keys.
- Azure Active Directory (Azure AD) is a multitenant, cloud-based directory and identity management service. In this scenario, Azure AD provides authentication services for the website and the VPN tunnels.

Alternatives

- SQL Server for Linux can replace the MariaDB data store.
- Azure database for MySQL can replace the MariaDB data store if you prefer a fully managed solution.

Considerations

Availability

The VM instances in this scenario are deployed across multiple regions, with the data replicated between the two via RSYNC for the WordPress content and master slave replication for the MariaDB clusters.

Scalability

This scenario uses virtual machine scale sets for the two front-end web server clusters in each region. With scale sets, the number of VM instances that run the front-end application tier can automatically scale in response to customer demand, or based on a defined schedule. For more information, see Overview of autoscale with virtual machine scale sets.

The back end is a MariaDB cluster in an availability set. For more information, see the MariaDB cluster tutorial.

For other scalability topics, see the [scalability checklist][scalability] in the Azure Architecture Center.

Security

All the virtual network traffic into the front-end application tier and protected by network security groups. Rules limit the flow of traffic so that only the front-end application tier VM instances can access the back-end database tier. No outbound Internet traffic is allowed from the database tier. To reduce the attack footprint, no direct remote management ports are open. For more information, see Azure network security groups.

For general guidance on designing secure scenarios, see the Azure Security Documentation.

Resiliency

In combination with the use of multiple regions, data replication and virtual machine scale sets, this scenario uses Azure load balancers. These networking components distribute traffic to the connected VM instances, and include health probes that ensure traffic is only distributed to healthy VMs. All of these networking components are fronted via a CDN. This makes the networking resources and application resilient to issues that would otherwise disrupt traffic and impact end-user access.

For general guidance on designing resilient scenarios, see Designing reliable Azure applications.

Pricing

To explore the cost of running this scenario, all of the services are pre-configured in the cost calculator. To see how the pricing would change for your particular use case, change the appropriate variables to match your expected traffic.

We have provided a pre-configured cost profile based on the architecture diagram provided above. To configure the pricing calculator for your use case, there are a couple main things to consider:

- How much traffic are you expecting in terms of GB/month? The amount of traffic will have the biggest impact on your cost, as it will impact the number of VMs that are required to surface the data in the virtual machine scale set. Additionally, it will directly correlate with the amount of data that is surfaced via the CDN.
- How much new data are you going to be writing to your website? New data written to your website correlates with how much data is mirrored across the regions.
- How much of your content is dynamic? How much is static? The variance around dynamic and static content influences how much data has to be retrieved from the database tier versus how much will be cached in the CDN.