Architectural design

Introduction

This document describes the architecture and infrastructure's migration conceptual and logical level in such a manner that it can be used as a starting point for further detailed designs. This document states functional components (building blocks) which provide the required functionality and the relations between these components. This document provides technical and/or product specifications, detailed interface descriptions, etc.

Background

Azure cloud provides hosting and technical application management capabilities for a number of applications. This document provides the conceptual migration design for the on premise WordPress application environment to the Azure public cloud.

Aim of this document

This document provides a detail level overview of the required solution. This solution is based upon the business requirements, starting points and assumptions which are stated in this document. The customer agrees to these business requirements, starting points and assumptions.

Situation

Customer is looking for scalable and flexible solution to be future proof and expandable with new WordPress sites with minimal effort. They are currently hosting 10 WordPress sites using WordPress Multisite in a private datacenter. They achieve high availability (HA) by using 2 servers and having two copies of their Multisite. For the database, they are using 2 MySQL servers behind and HAProxy to achieve HA.

The past few months, they have been having a lot of issues because some of their websites have increased in popularity, especially during certain timeframes. For the future state, they have agreed that they want to move away from Multisite, and have independent Wordpress applications. They have also pointed out that they have 5 more sites in the making that will reach Production in the next 12 months.

The client is only interested in developing the WordPress sites from an application perspective. They work using GIT repositories, and they have agreed to provide access to the application source code in one or more repositories.

Aim of the design

This project is intended to create and implement scalable, flexible and future proof re-architecture to run application with high availability in Azure infrastructure. All assumptions and decisions are recorded in this document.

Preconditions and Assumptions

Sometimes preconditions exist to which the design must adhere. If any preconditions exist they are listed here.

Preconditions

Customer maintains application code in Git repository and willing to have solution which can help with CI/CD to have agile way of working.

Provided solution should enable customer to develop only WordPress application.

Assumptions

Develop a cloud adoption strategy

Executive summary

The Cloud Adoption Framework helps customers undertake a simplified cloud adoption journey. This framework contains detailed information covering an end-to-end cloud adoption journey, starting with targeted business outcomes and aligning cloud readiness and assessments with clearly defined business objectives. Those outcomes are achieved through a defined path for cloud adoption. With migration-based adoption, the defined path focuses largely on completing a migration of on-premises workloads to the cloud. Sometimes this journey includes modernization of workloads to increase the return on investment from the migration effort.

Cloud adoption strategy assumptions

Motivation: Assuming this case as a Critical Business Event to reduce disruptions and improve IT stability. When a response to critical business events is the highest priority, it is important to engage in cloud implementation early, often in parallel with strategy and planning efforts. I assume customer has a growth mindset and a willingness to iteratively improve processes, based on direct lessons learned.

Business outcomes: Assuming this case as a performance outcome is required. Performance and reliability are assumed. When either falters, reputation damage can be painful and long-lasting. Since customer had issues after their site increased in the popularity.

Business justification: Assuming factor mentioned here is part of business justification presented to customer and there is agreement with customer on the same. Customer is considering reliability of application over cost, customer want to restructure application design during migration to cloud. It's known to customer that redesigning and migration of application can take time because of potential road blocks. It also required team effort between cloud migration team and current service providers for customer from internal employees or external vendors.

First cloud adoption project: Provided case in the part one is considered as project chosen based on available hands on experience, knowledge and skills required to achieve goal.

Develop a cloud adoption plan

Executive summary

Cloud adoption plans convert the aspirational goals of a cloud adoption strategy into an actionable plan. The collective cloud teams can use the cloud adoption plan to guide their technical efforts and align them with the business strategy

Cloud adoption plan assumptions

Digital estate: Cloud rationalization is the process of evaluating assets to determine the best approach to hosting them in the cloud. Assuming provided cloud migration approach is determined and aggregated an inventory based on that. Incremental approach will be used for digital estate planning.

Initial organization alignment: Assuming organization alignment is already established to support to the cloud adoption plan and cloud governance.

Skill readiness plan: As per given information assuming customer have required skills to work with CI/CD created through Git repository. Deploying and managing infrastructure deployed in Azure will be responsibility of Sentia.

Cloud adoption plan: Assuming cloud adoption plan is developed to manage change across the digital estate, skill and organization.

Ensure the environment is prepared for the cloud adoption plan

Before adoption can begin, you must create a landing zone to host the workloads that you plan to build in the cloud or migrate to the cloud.

Landing zone exercises assumptions

Azure Readiness: Review for the Azure Readiness to create a landing zone is completed.

First landing zone: Evaluating the Cloud Adoption Framework to migrate and create the landing zone with blueprint is completed.

Expand the blueprint: Use the landing zone considerations to identify and make any necessary modifications to the blueprint template based on customer's business governance considerations and hosting considerations.

Best practices: Assuming landing zone modifications are validated against the best practices sections to ensure the proper configuration of current and future landing zone. Determining Azure fundamentals (example: subscription, naming and tagging conventions, organizing resources with Azure management groups etc.) networking decisions, identity and access control (RBAC), storage, databases, and cost management.

Decisions

Compute: To provide maximum availability and scalability Docker Swarm on Linux VM scale set is configured.

Storage: Production VM scale set is running on managed disk, DTA environment is using standard LRS disk to provide cost savings.

Networking: Entire infrastructure is divided with Vnet and secured with Network Security Group, subnet provides logical boundary between front-end web server and backend database servers. This solution prevent from malicious traffic and provide enhance security. Use of Content Delivery Network (CDN) insures security and quality of service.

Database: Database solution is implemented to insure high availability and resiliency, using MySql DB on VM availability set is cheaper than Azure MySql DB service.

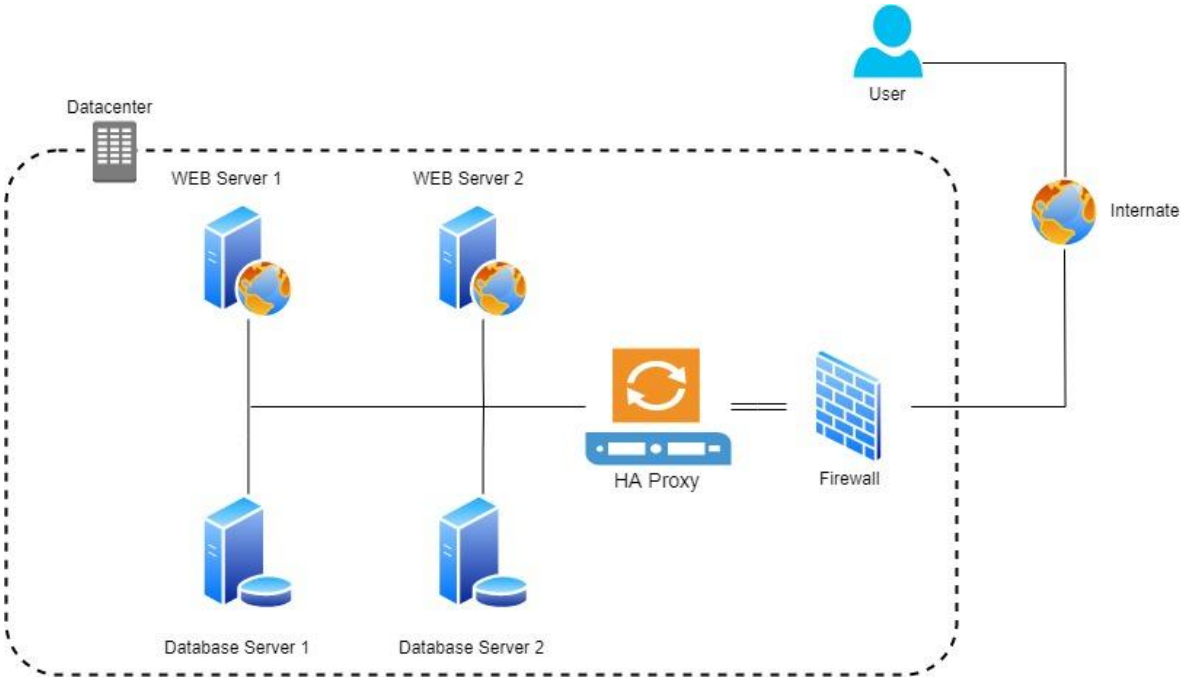
Containerization: Using Docker Hub for this task is because of available support, documentation and community support along with allowing multiple micro-services to function together without in-depth orchestration at the VM level.

Rapid deployment: A developer with appropriate access can simply push to the correct remote and the deploy happens automatically from there

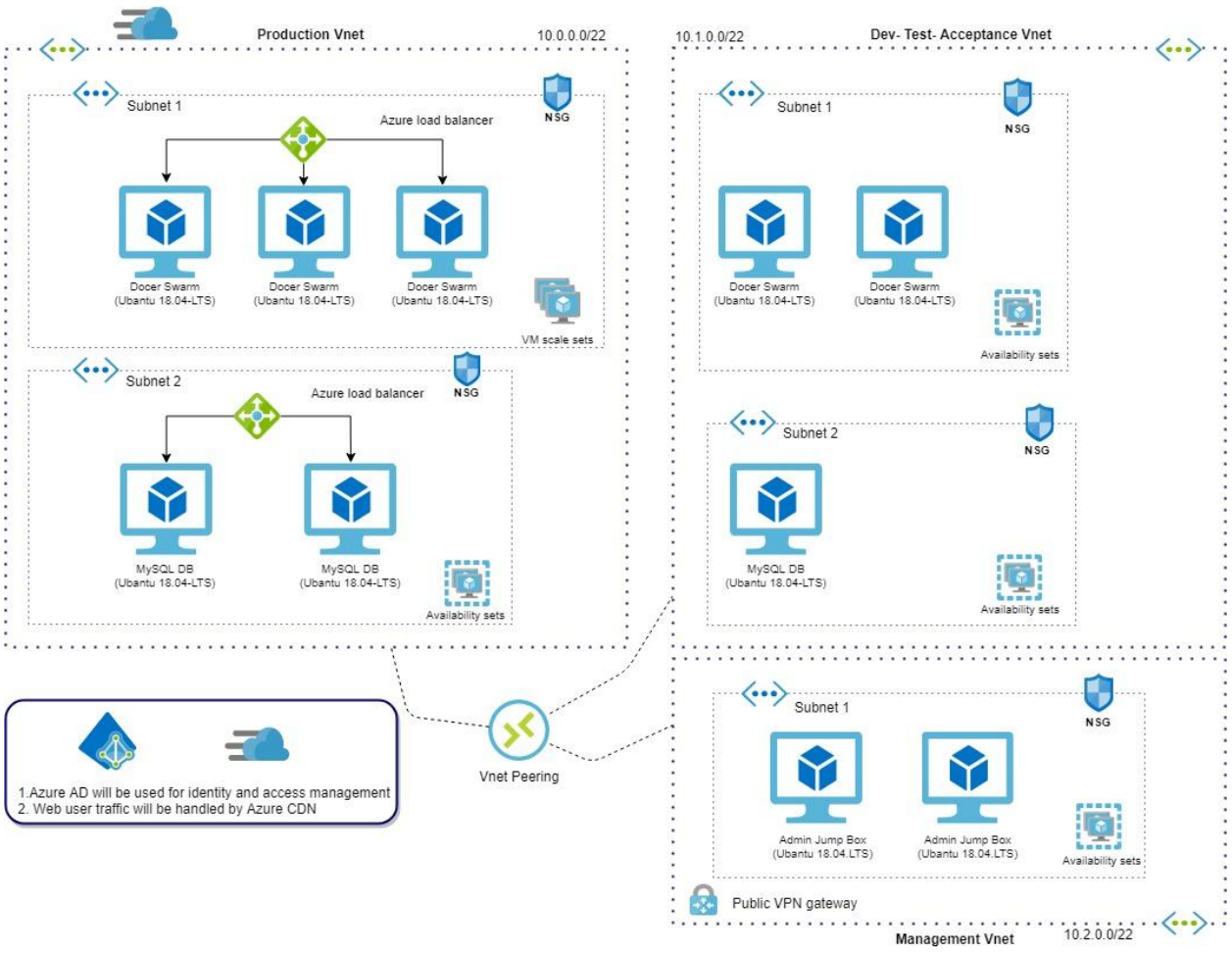
Decouples provisioning from VM administration: Allows containers to be deployed (more), regardless of VM set up.

Note: Azure Kubernetes can be used as a robust solution; however, knowledge for a particular product needs to learn on my personal level.

Current Architectural design



Architecture



Conceptual design after migration

This architectural design covers a scalable and secure installation of WordPress that uses Ubuntu web servers and Mysql DB. There are three 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 on Docker containers.
4. The WordPress application pulls any dynamic information out of the Mysql DB cluster.

The second workflow is how application/system administrators connect for daily operations.

1. Administrators connect securely to the public VPN gateway.
2. User Identity and authentication occurs with the help of Azure Active Directory.

3. A connection is then established to the Admin jump boxes.
4. From the admin jump box, administrator 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 Docker container running web servers that have write access to the Mysql DB cluster.

Third workflow is how administrators connect to Development, Test and Acceptance (DTA) environment.

1. Administrators use same jump boxes to connect DTA environment as production.

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 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.

Considerations

Availability

The VM instances in this scenario are deployed in VM scale sets across 3 zones for high availability. Docker Swarm configured to provide high availability to containers running WordPress application. WordPress application is backed up by the MySql DB cluster running in the availability set.

Scalability

This scenario uses virtual machine scale sets for the three front-end web server running Docker Swarm. 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.

The back end is a Mysql DB cluster in the scale set.

Security

The entire 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.

Resiliency

In combination with the use of 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.