Problem 2: Using Azure PAAS Component

# Azure Services Mapping (No Infrastructure)

## Physical Server Layout (Optimized & Distributed)

|  |  |  |
| --- | --- | --- |
| **Physical Server** | **Main Function** | **Hosted VM/Service** |
| Server 1 | Critical App + Database | VM1 (S1 – Linux App + MySQL) |
| Server 2 | Web Frontend for S1 | VM2 (S2 – PHP/MySQL Web App) |
| Server 3 | Independent Web App | VM3 (S3 – PHP/PostgreSQL Web App) |
| Server 4 | Lightweight App | VM4 (S4 – Python/Django + Azure Files/GlusterFS) |
| Server 5 | Heavy Aggregator App | VM5 (S5 – Java Web App) |
| Server 6 | Dedicated Backup Server | VM6 (Bacula/Restic + NAS Mounts) |
| Server 7 (Optional) | Monitoring & Security Systems | VM7 (Prometheus, Grafana, Security Tools) |

### **Separation** ensures resilience and scalability.

## System Architecture

* Dedicated server for each major workload = better **fault tolerance**.
* Separate **Backup** server (isolated storage zone).
* **Load balancing** at network/proxy level for external traffic.
* **Management traffic** separated via VLAN.
* Dual ISP connectivity for **redundancy**.
* Design ready for **cloud extension** in the future (Hybrid-ready).

## Network Design

|  |  |
| --- | --- |
| **Component** | **Design** |
| Routers/Firewalls | Dual pfSense/OPNsense in HA (High Availability Cluster) |
| Switch | L3 Core Switch with VLAN segmentation |
| Public Access | Reverse Proxy (HAProxy/NGINX) |
| Private Access | VPN required (WireGuard/OpenVPN) |
| DDoS Protection | Router level + Cloudflare (Optional) |
| Internal DB communication | Encrypted (SSL-internal certificates) |

## Public DNS Mapping

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **System** | **Public DNS** | | S1 | s1.example.com | | S2 | s2.example.com | | S3 | s3.example.com | | S4 | s4.example.com | | S5 | s5.example.com | |  |
|  |  |

* Managed through Reverse Proxy.
* SSL certificates via Let's Encrypt.
* Backup & Disaster Recovery

|  |  |
| --- | --- |
| **Strategy** | **Details** |
| Full Backup | Weekly full backup (VMs + Databases) |
| Incremental | Daily |
| Mid-week Differential | Wednesday nights |
| Snapshot Retention | 4 weeks rolling backup cycle |
| Disaster Recovery | Offsite backup copies monthly (if cloud used) |

## Monitoring and Security Plan

|  |  |
| --- | --- |
| **Monitoring** | **Tools** |
| System Health | Prometheus + Grafana |
| Log Management | Loki or Elastic Stack (optional) |
| Alerts | Slack/Email alerts on critical thresholds |
| Security | Tools |
| Firewall | pfSense/OPNsense HA Cluster |
| VPN Access | WireGuard (preferred) |
| Fail2Ban | Installed on all VMs |
| Update Automation | Ansible Playbooks |
| Endpoint Protection | CrowdSec or OSSEC (optional) |

## Scalability and Redundancy

|  |  |
| --- | --- |
| **Aspect** | **Best Practice** |
| Redundancy | HA Firewalls, dual ISPs, RAID storage |
| Scalability | Add more physical servers into VLANs |
| Load Balancing | Front Door Layer (HAProxy) |
| Disaster Recovery | Weekly cloud sync backups (optional) |
| Expansion | Hybrid cloud possible at any point (Azure/AWS ready) |

## Final Advantages Summary

|  |  |
| --- | --- |
| **Advantage** | **Description** |
| High Availability | Dual ISPs, Firewall HA, Server separation |
| Fault Isolation | Failures limited to one server at a time |
| High Security | VLANs, VPN access, Firewall Rules |
| Excellent Performance | No resource contention between apps |
| Disaster Recovery | Robust backup + offsite option |
| Scalability | Easily expand by adding new servers |
| Cloud Ready | Easy migration later if needed |