

## ***Cloud architecture design and proposal for Victorino's Secrets***

By: Julio César Ortega Martínez

### **Introduction:**

Victorino's secrets, a medium size company located in Bogotá, has been successful on it's local retail market and they are growing towards the digital market. They have a small but very loyal customer base, and, they want to make their firsts steps into the cloud. Victorino's secret has very little knowledge and information on how to take advantage of cloud based services, and, they are seeking our expertise to be able to take the right decisions on moving to the cloud.

### Existing platform:

#### *Physical layer*

Victorino's secret current platform consists of the following on-premise components:

- One (1) NetScreen-5GT firewall (*firewall*)
- One (1) Quad AMD Opteron barcelona @ 2.5GHz, 8GB RAM, 4x320GB LVM Striped (*web server*)
- One (1) Dual Intel Pentium 4 netburst @ 2.0 GHz, 2GB RAM, 2x80GB Disks on RAID 1 (*database server*)

#### *Logical layer*

The web server runs Apache web server, and also a node.js instance that runs an api microservice

The database server runs MySQL

The firewall has the following configuration:

- DMZ and private zones
- The DMZ, where the web server is located, has two public ports open: HTTP (port 80) and SSH (port 22)
- The private zone currently encloses only the database server.

The solution responds using two DNS zones:

- glbacademies.com pointing to the web server.
- api.glbacademies.com pointing to the node.js microservice, wich is proxied via the web server.

The current infrastructure presents a number of issues, being the most important ones:

1. Few load peaks through the year , customers tend to be angry with site responsiveness on these dates.
2. Patch management implies manual reboots or (remote hands) assistance.

## CloudAcademy 2.0 – Exercise 1

3. Maintenance windows for the database are lasting more than scheduled. Even when the server has 15000 RPM SCSI disks.
4. There is an additional service with the colocation provider for a NAS with 1TB capacity that is being used for application assets (images,multimedia,user files, etc) and these assets are growing, the service is quite expensive.

### **Proposal**

Based on the aforementioned platform, and the availability of Amazon Web Services (AWS) as a cloud provider, these are our considerations and recommendations for migrating the existing platform to the AWS cloud.

#### *Data Management*

Database service should be moved to a dedicated AWS RDS instance running MySQL, given the capacity that is currently in use on their on-premises solution and, projecting for better performance and reduced maintenance windows, the initial provisioning for this service should be from a db.t2.medium to a db.t2.large instance with 80GB of EBS SSD provisioned storage.

All static content currently on the existing NAS should be moved over to an AWS S3 bucket, where it would be statically served, as well as updating references to these static objects on the main WordPress website. This would greatly reduce load on the main web server, and, it will result in lower transfer cost for the instance where it runs.

All temporary dynamic content on the web server should be assessed, to make sincere storage provisioning decisions for the EC2 instance where the web server will run.

#### *Network Segmentation*

As there would be no need to split the network in a DMZ or Private Zone as we move the platform to the AWS cloud, we only take care of creating the appropriate security groups, with their corresponding firewall rules for every service. We propose the client to keep the current private zone address space of 172.16.55.0/26, also, considering role separation, we suggest to implement another subnet for all their database infrastructure, which is expected to grow over time. The proposal stands upon using 172.16.105.0/26 for this subnet.

So, based on these premises, we propose to create one AWS VPC with a CIDR of 172.16.0.0/16 and, their corresponding subnets.

## CloudAcademy 2.0 – Exercise 1

### *Other considerations*

Based on the corresponding web server configuration we propose creating a simple EC2 instance capable of running both services currently present on the on-premises hardware, so, we decide upon provisioning an m4.xlarge EC2 instance, doubling the available RAM currently on premises, and, working with the assumption that the server would reduce its load due to the fact that we are moving the static content to an S3 Bucket, thus, giving room to face the frequent load peaks without problem.

### *Pros / Cons*

#### Pros:

- The solution is in the cloud, so the client will be able to grow as demand grows, and will not be constrained by hardware provisioning.
- Full control of resources as the solution is not physically constrained, and can be accessed and supported easily.
- Freedom to change the architecture and expand upon existing cloud services to improve.
- The cloud offers better availability of the service (as it operates over multiple availability regions and can withstand outages that a single point availability within a region cannot support)
- Maintenance and backups can be done without worrying about availability of resources to store and transfer information.

#### Cons:

- Higher costs of operation.
- Better network security assessment is needed when dealing with a cloud infrastructure, as it could quickly increment network load and costs.