

Aplicações e Serviços de Computação em Nuvem

Trabalho Prático

2022/2023

Informações Gerais

- Cada grupo é constituído por **5 elementos**. A inscrição é feita via a plataforma *blackboard*.
- O trabalho prático é concluído com a entrega de um relatório via a plataforma *blackboard*.
- **Todos os scripts e ficheiros de configuração** usados na realização do trabalho têm de ser disponibilizados através de um **repositório *github* público**. Importante: As credenciais da conta de serviço GCP (ficheiro JSON) não podem ser colocadas no repositório público!
- É **obrigatório** que o relatório identifique todos os elementos do grupo na capa. É **também obrigatório** incluir o *url* do repositório *github* na capa do relatório.
- O **prazo limite para a entrega do trabalho é 14 de Janeiro de 2023**. Após esta data **não serão considerados** relatórios ou novos *commits* no repositório *github*.
- A apresentação e discussão do trabalho decorre na semana entre 16 e 20 de Janeiro de 2023.
- Uma *FAQ* do trabalho será mantida e atualizada com dúvidas relevantes [aqui](#).

Enunciado

O trabalho prático pretende exercitar os conhecimentos adquiridos na cadeira de forma a **automatizar o processo de instalação, configuração, monitorização e avaliação da aplicação Ghost** (<https://ghost.org>).

Tarefas

Tarefa Base – Instalação e Configuração Automática da Aplicação (12 Valores)

Objetivo: A primeira tarefa consiste em utilizar a ferramenta **Ansible** para automatizar a instalação e configuração da aplicação Ghost no serviço **Google Kubernetes Engine (GKE)** da Google Cloud.

Requisitos (i.e., para além da utilização de Ansible e do serviço Kubernetes Engine):

- Os **diferentes componentes da aplicação Ghost** devem executar, quando possível, em **Pods** distintas.
- A instalação e configuração da aplicação, bem como dos serviços da plataforma Google Cloud, devem ser efetuadas de forma **automática no menor número de passos manuais possíveis**.
- **Deve ser possível parar a aplicação e voltar a executar a mesma** (p.ex., devido a uma tarefa de manutenção programada) **sem que sejam perdidos dados críticos** (p.ex., dados de utilizadores).
- **É obrigatório** respeitar a estrutura e funcionalidades dos **playbooks Ansible** disponibilizados (ver secção de Requisitos de Desenvolvimento).

Resultado: Após instalada e configurada, a aplicação Ghost deve ser **acessível a partir do exterior** através de um *browser*.

O trabalho inclui mais 3 tarefas que tiram proveito da correta execução da tarefa base supramencionada, valorizando a mesma com outros conhecimentos adquiridos na cadeira.

Tarefa 1 – Monitorização (3 Valores)

Objetivo: Esta tarefa consiste na aplicação de ferramentas de monitorização que permitam observar a instalação da aplicação Ghost.

Considerações:

- Para esta tarefa, o grupo deve explorar a ferramenta de monitorização disponibilizada pela plataforma Google Cloud.
- É importante considerar a captura de diferentes métricas de monitorização (p.ex. CPU, RAM, I/O), justificando a sua escolha.
- A visualização e correlação customizada (p.ex., através da criação de novas *dashboards*) destas métricas deve ser considerada e justificada.
- A configuração e customização da ferramenta deve também ser feita de forma automática.

Resultado: Deve ser possível aceder a uma interface de monitorização que apresente as métricas e visualizações escolhidas para a instalação da aplicação Ghost.

Tarefa 2 – Avaliação Experimental (3 Valores)

Objetivo: Nesta tarefa é pretendido que o grupo demonstre o método utilizado para validar experimentalmente a instalação e configuração da aplicação Ghost.

Considerações:

- Os testes propostos devem avaliar diferentes funcionalidades e/ou componentes da aplicação, tendo também em conta a interação típica dos utilizadores com a mesma.
- A execução dos testes desenvolvidos deve também ser automatizada.
- Caso a Tarefa 1 tenha sido realizada, a avaliação experimental da aplicação pode ser combinada com as métricas de monitorização para extrair observações mais completas sobre os resultados observados.

Resultado: Deve ser possível avaliar a instalação da aplicação Ghost e analisar os resultados obtidos de forma a validar e compreender melhor o seu funcionamento.

Tarefa 3 – Escalabilidade e Resiliência (2 Valores)

Objetivo: Nesta tarefa é pretendido dotar a aplicação de mecanismos de replicação que permitam melhorar o desempenho e/ou resiliência da instalação do Ghost.

Considerações:

- De forma a reduzir a complexidade desta tarefa, a mesma apenas tem de se focar num **único componente** da aplicação (p.ex., servidor aplicacional, base de dados).
- Para o componente escolhido pelo grupo, deve ser possível aumentar ou diminuir o número de réplicas de forma automática e configurável (p.ex., através de um *playbook Ansible*).
- Caso as Tarefas 1 e 2 tenham sido realizadas, as ferramentas de avaliação e monitorização podem ser aproveitadas para avaliar a eficácia do mecanismo de replicação.

Resultado: Deve ser possível aumentar e reduzir o número de réplicas de um dos componentes do ghost de forma automática e configurável.

Requisitos de Desenvolvimento

A solução do trabalho deve seguir a estrutura do código base disponibilizado juntamente com este enunciado. Nomeadamente, **na pasta raiz** do repositório Github público de cada grupo devem constar:

5 playbooks:

- **create-gke-cluster.yml** e **destroy-gke-cluster.yml** - cria e elimina, respetivamente, o *cluster Kubernetes* no serviço GKE.
- **deploy-ghost.yml** - Instala, configura e executa, no *cluster* GKE criado previamente com o Playbook **create-gke-cluster.yml**, todos os componentes da aplicação Ghost.
 - Este *playbook* pode ser corrido quer para uma instalação completamente nova do Ghost, quer para uma instalação prévia para a qual a execução foi terminada mas cujos dados persistentes não foram removidos.
 - Quando é efetuada uma instalação nova, o *playbook* tem de criar automaticamente um administrador de Blog com os seguintes dados:
 - **Nome:** ascn
 - **Palavra-passe:** ascn123
 - **Email:** ascn@example.com
- **undeploy-ghost.yml** - Termina a execução da instalação atual da aplicação Ghost.
 - Apenas quando for especificada a flag (**-e delete_data=true**) é que se deve proceder à remoção completa de todos os dados persistentes da instalação.
- **test-all.yml** - Efetua um conjunto de testes automáticos para validar a execução do Ghost.

A pasta **roles** que inclui os *roles* Ansible necessários para a execução dos 5 *playbooks* anteriores.

- Os *roles* **gke_cluster_create** e **gke_cluster_destroy** necessitam de uma conta de serviço e da ferramenta [Google Cloud CLI](#) para aceder ao GKE (**Nota:** ver *slides GKE* no Blackboard).
- O *role* **test_ghost**, invocado pelo *playbook* **test-all.yml**, testa a disponibilidade da aplicação Ghost, por exemplo, efetuando um pedido *HTTP GET* à página principal do Blog.

Na pasta **inventory**, o ficheiro **gcp.yml** contém variáveis a serem utilizadas pelos *roles* anteriores. As variáveis cujo nome começa com **gcp** dizem respeito a configurações do *cluster* GKE e devem ser atualizadas por cada grupo. Por exemplo:

- A variável **gcp_project** deve identificar o nome do projeto na Google Cloud.
- A variável **gcp_cred_file** deve indicar o caminho para o ficheiro JSON que contém a informação relativa à conta de serviço criada pelo grupo. É **importante** relembrar que este ficheiro tem credenciais privadas e não pode ser colocado no repositório github público!
- Para a correta execução do *playbook* **test-all.yml** é **obrigatório** que, aquando a execução do *playbook* **deploy-ghost.yml**, sejam atualizadas automaticamente as seguintes variáveis:
 - **ghost_ip** que corresponde ao endereço IP público da aplicação Ghost.
 - **ghost_port** que corresponde ao porto público da aplicação.

NOTAS:

- A estrutura do código base (ficheiros e pastas) tem de ser respeitada e colocada diretamente na pasta raiz do repositório *github*.
- O código dos *playbooks* de criação e destruição do cluster GKE, assim como os respectivos *roles*, são fornecidos pela equipa docente e não devem ser alterados. **No entanto**, as variáveis que dizem respeito ao projeto e contas de serviço devem ser atualizadas.
- O *playbook* **test-all.yml**, e respectivo *role*, permitem validar a correta implementação dos restantes *playbooks*, assim como a alteração correta das variáveis do inventário. Assim sendo, o código existente não deve ser alterado. **No entanto**, o *playbook* e *role* podem ser estendidos para incluir testes adicionais (Tarefa 2 do enunciado).
- A criação de outros *playbooks*, *roles* e variáveis, com vista a responder às tarefas do enunciado, fica ao critério de cada grupo.

Requisitos do Relatório

O relatório apresentado deve conter a seguinte informação. De notar que a ordem pela qual cada um dos pontos em baixo é apresentada no relatório fica ao critério do grupo.

- Uma descrição breve da arquitetura e principais componentes da aplicação Ghost.
- Identificação das ferramentas e abordagem utilizadas para a instalação e configuração automática da aplicação.
- Apresentação e análise crítica da abordagem escolhida para o mecanismo de replicação.
- Ferramentas de monitorização, métricas e visualização escolhidas, justificando a sua escolha.
- Ferramentas de avaliação e testes desenvolvidos, justificando a sua escolha.
- Apresentação e análise dos resultados da avaliação experimental.
- Reflexão final sobre o trabalho apontando os principais pontos fortes e os pontos a melhorar.

O relatório não necessita de incluir os scripts de aprovisionamento (estes são disponibilizados no repositório *github*) e deve focar-se nas principais decisões tomadas para resolver os pontos especificados neste enunciado. Embora não exista um limite máximo de páginas, um relatório de 10 páginas deverá ser suficiente para responder ao enunciado.

Mais ainda, cada grupo deverá estar preparado para fazer uma breve apresentação do relatório que contempla uma demonstração de 5 minutos. Esta demonstração deverá ser realizada utilizando a Google Cloud Platform e, por uma questão de otimização de tempo, cada grupo deve garantir que o cluster de *Kubernetes* GKE é criado e encontra-se utilizável antes da apresentação.

Créditos Google Cloud

Cada grupo terá um **limite de 200\$** de crédito no GCP. As instruções para aceder ao mesmo serão transmitidas posteriormente via a plataforma *blackboard*. Como regras importantes:

- A gestão do crédito é da responsabilidade de cada grupo. O mesmo será atribuído pela equipa docente de forma faseada (i.e., em cupões de 50\$) e a pedido do grupo.
- O crédito deve ser suficiente para realizar as diversas tarefas do trabalho, bem como para a **demonstração final** do trabalho prático.
- **De forma a fazer uma gestão eficiente do crédito é importante:**
 - Desligar todos os serviços GCP (p.ex., Containers, *Kubernetes*, VMs, armazenamento, etc) quando estes não estiverem a ser usados.
 - Ter em atenção o custo dos recursos a serem requisitados. Por exemplo, a capacidade de processamento, memória e armazenamento de uma instância (VM, container, ...) tem um impacto direto no seu preço.
 - Fazer testes preliminares com instâncias com um menor número de recursos (mais baratas) e aumentar o mesmo quando necessário e, por exemplo, para testes / demonstrações finais.