Documentação Prévia para Documentação de BPC E RDP

Xperience XPXR Aplicação.

A alta disponibilidade e escalabilidade são requisitos desejáveis em qualquer sistema, e apesar de serem vistas com frequência juntas, as soluções que visam prover alta disponibilidade não implicam em escalabilidade e vice-versa. Para fins deste artigo, podemos definir que:

• Alta disponibilidade é a capacidade de um sistema/servidor estar disponível por grandes períodos de tempo;

• Escalabilidade é a capacidade de oferecer serviços com qualidade aceitável mesmo com uma demanda crescente.

Este artigo abordará apenas o primeiro item, uma vez que o segundo requer um conhecimento muito mais específico da aplicação que se deseja atender.

Uma vez que um serviço é colocado à disposição dos usuários, uma pressão, antes inexistente, se manifesta: o serviço deve estar disponível sempre que necessário!

Infelizmente, em qualquer ambiente temos que conviver com eventos que atuam para impedir o objetivo acima de ser satisfeito. Estes eventos são as paralisações dos servidores/serviços.

Para minimizarmos seus impactos temos que entender os tipos de paralisações. Estas podem ser divididas em duas categorias:

• **Não planejadas:**são o resultado de falhas aleatórias (imprevisíveis) dos sistemas, associadas a componentes de hardware ou software. Como exemplo, temos discos rígidos com defeitos físicos, memórias com erros, fontes de alimentação queimadas, entre outros;

• **Planejadas (ou manutenções):**associadas a paralisações agendadas para promover a atualização de hardware ou software. Costumam oferecer menor impacto ao negócio por normalmente serem realizadas em momentos de menor tráfego.

No caso das paralisações não planejadas, você pode ter uma postura preventiva, trocando, por exemplo, componentes antes do tempo médio do aparecimento de problemas. Entretanto, tal atitude não é uma garantia de que o componente não apresentará problemas antes dessa data.

Mas qual o impacto de uma paralisação? Apesar de não ser fácil quantificar o prejuízo de uma paralisação, é possível realizar um simples exercício para se entender o potencial que um problema pode alcançar.

Suponha que você tenha um servidor que demora 5 minutos para completar o processo de reinicialização. Se ele é responsável por atender a 100 requisições por segundo, isso significa que teremos 5 minutos x 60 segundos x 100 requisições por segundo = 30.000 requisições não atendidas!

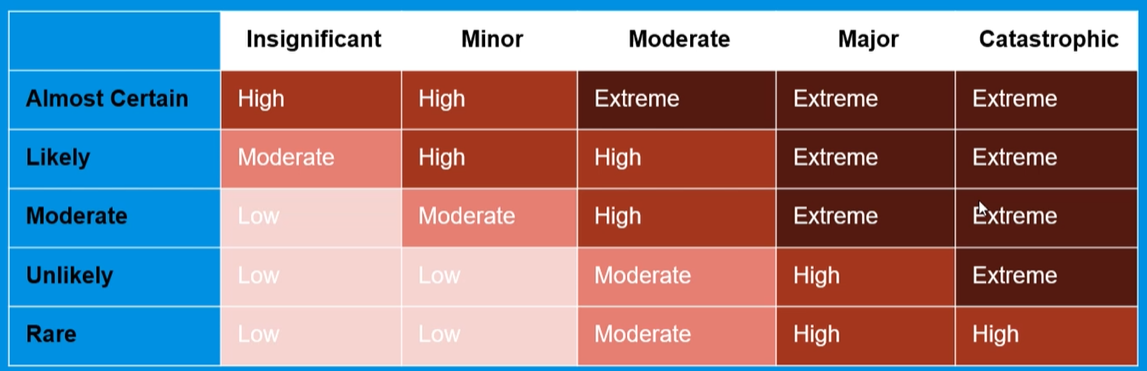
Apesar do desejo de termos uma disponibilidade de 100%, este valor não é realizável na prática. Por conta disso, precisamos definir qual o nível de disponibilidade, conhecido como SLA (Service Level Agreement – que define os níveis aceitáveis para um serviço), antes de desenharmos a nossa solução.

Usualmente iniciamos estimando o tempo máximo desejável de parada (*downtime*) durante um ano de operação medido em minutos, e encontramos qual a classe de disponibilidade que satisfaz o requisito utilizando a seguinte fórmula:

D = (1 – R) X 365 X 24 X 60x 365 x 24 x 60

Onde D é o tempo máximo de parada em minutos e R é a razão entre o tempo disponível e o indisponível. No mercado é comum encontrarmos os níveis conforme a Tabela.

|  |
| --- |
|  |
| **Disponibilidade (%)** | **Tempo de parada no ano** | |  | |
| **99** | 3,65 dias |  | |  |
| **99,9** | 8,76 horas |  | |  |
| **99,99** | 52,6 minutos |  | |  |
| **99,999** | 5,26 minutos |  | |  |
| **99,9999** | 30 segundos |  |  |  |

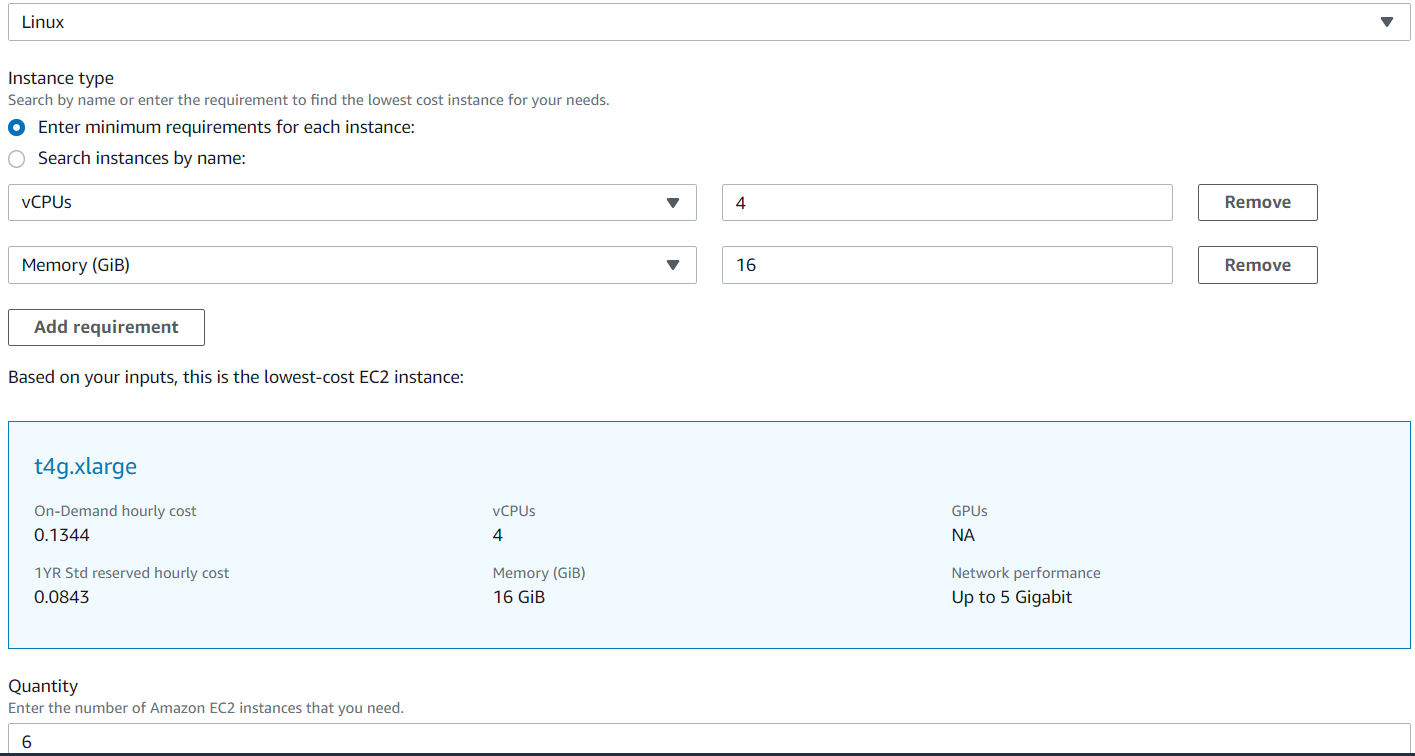


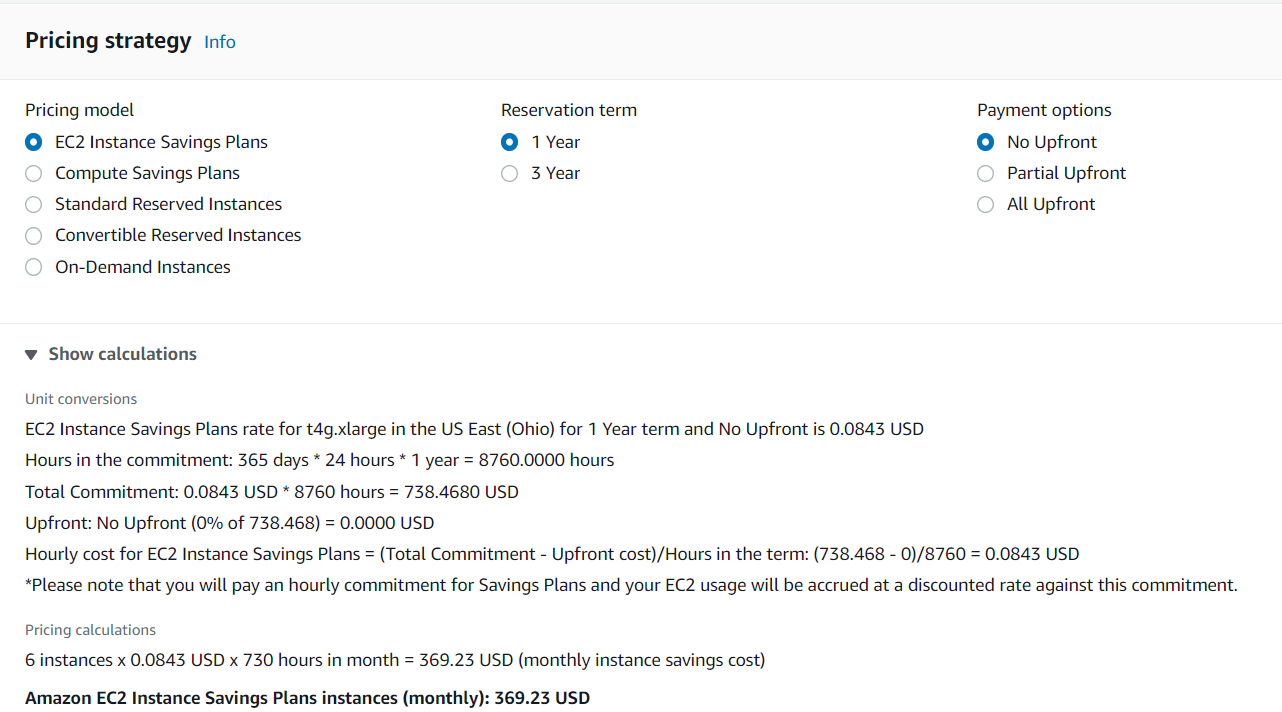
Essa tabela demonstra como devemos preencher a tabela de riscos para ameaças em casos de incidentes.

Precisamos reunir os dados para poder fazer a classificação.

Esse cálculo abaixo são de 2 tipos de montagem de máquinas para dois tipos diferentes de abordagens técnicas para solução que procuramos, uma contém a quantidade de 6 máquinas, já a outra possui a quantidade de 2.

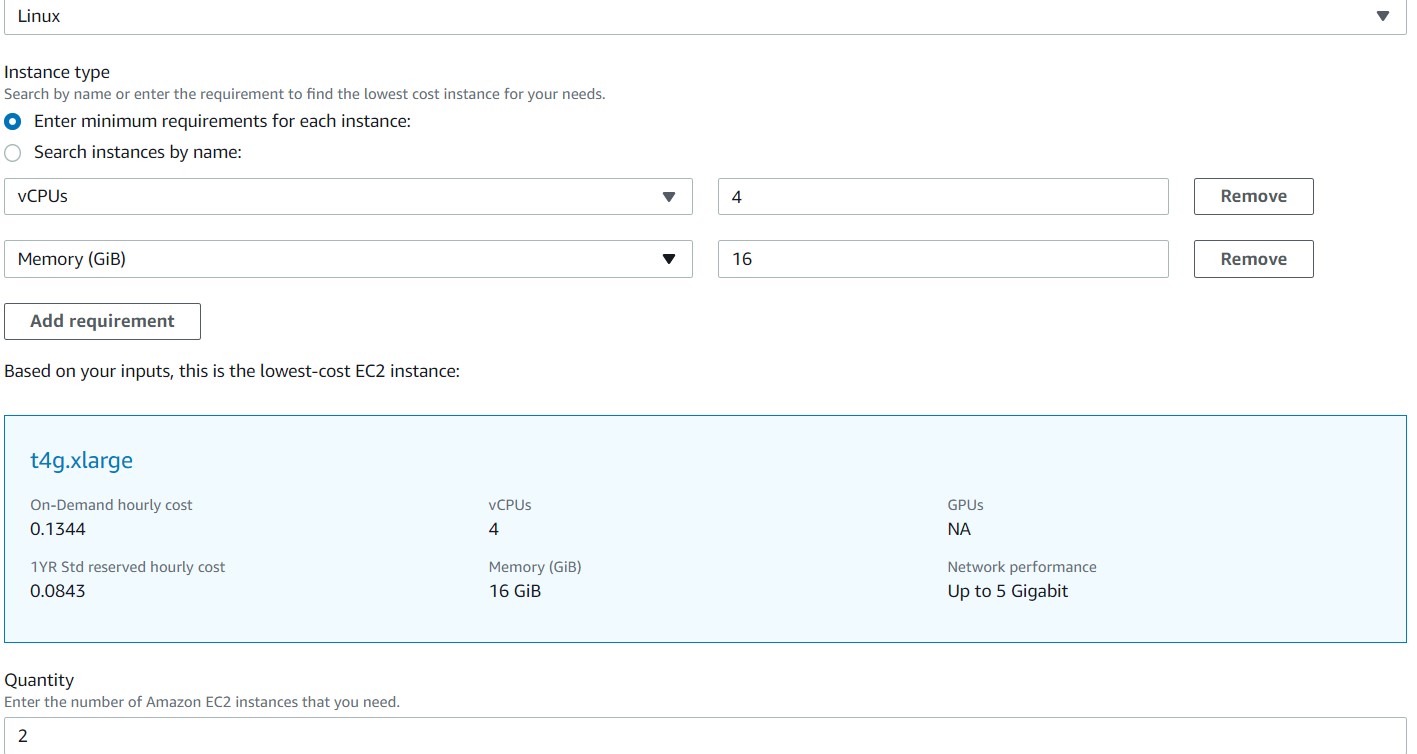
Cálculo de instâncias EC2 para montagem de Load Balance para RDP

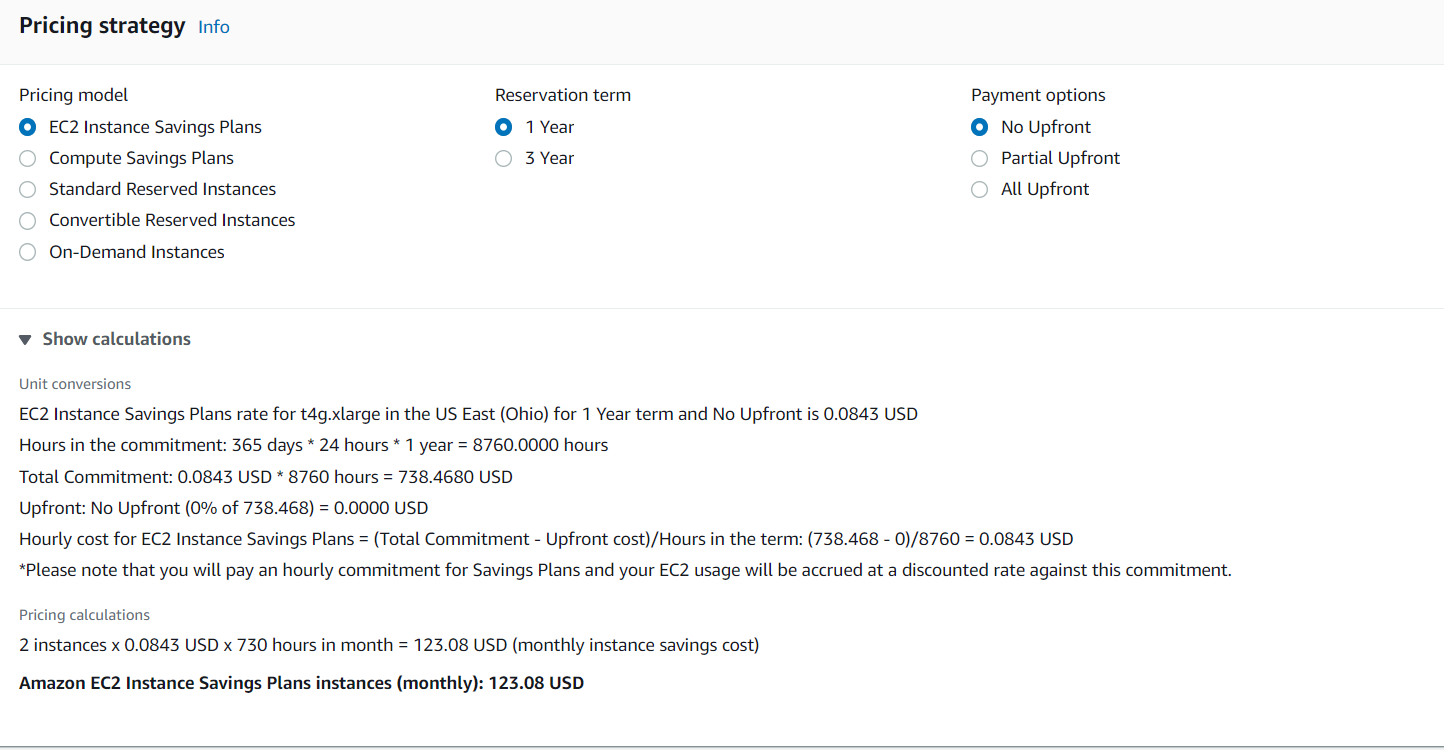




Isso conta com 6 instâncias EC2, sendo 2 para a aplicação de Front-End, 2 para Back-end, 1 File Server e 1 para Load Balance.

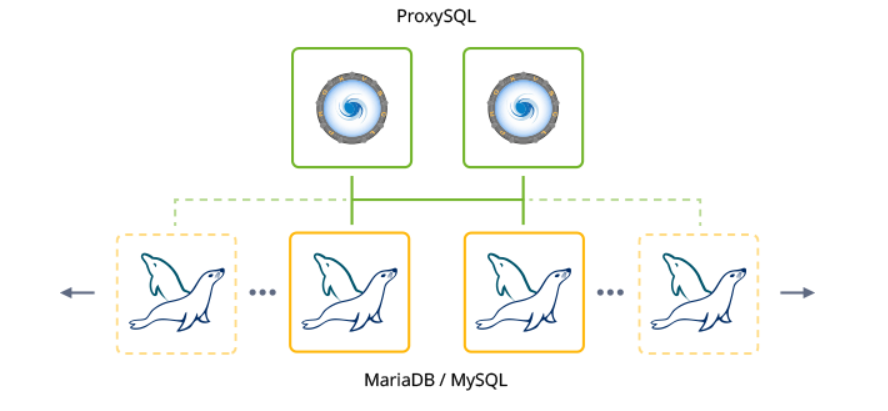
Isso conta com a replicação do servidor para caso de uma aplicação estiver down, estarmos com outra instância em funcionamento para operar com alta disponibilidade.





Nesse caso encontramos a possibilidade de montar 2 instâncias fazendo o load balance de aplicações dentro da mesma, separando somente pela parte de front-end e back-end.

O custo diminui substancialmente, porém garante uma menor alta-disponibilidade, sendo que teremos a aplicação em load balance dentro do servidor, então isso garante indisponibilidade de uma aplicação irá entrar em funcionamento a outra, porém em caso de queda do servidor, não terá alta-disponibilidade.



A solução implementada fornece um conjunto de benefícios:

* alta disponibilidade com opções de replicação pré-configuradas – Master-Slave, Master-Master, Galera, Replicação de Grupo Simples e Multi-Principal
* escalabilidade e descoberta automática – novos nós, adicionados durante o dimensionamento horizontal , são conectados ao cluster com todos os ajustes necessários sendo aplicados automaticamente
* Balanceamento de carga eficiente – cada cluster é complementado com dois nós ProxySQL para balanceamento de carga com divisão automática de solicitações de leitura / gravação
* failover automatizado – os nós do banco de dados que estão temporariamente indisponíveis ou possuem alta latência são automaticamente excluídos do cluster e adicionados novamente quando a conexão é restaurada

Todos esses benefícios podem ser alcançados com apenas alguns cliques em um assistente de topologia. Explore as etapas abaixo para ativar o armazenamento em cluster automático para seus bancos de dados MariaDB e MySQL no Jelastic PaaS.

Para a questão de custos dentro da AWS para esse tipo de funcionamento, seria de $0,017 por hora de cada instância, fazendo os cálculos de (4X24X30)X0,017 = $48,96

Já em questão dos snapshots (Backups em nuvem) seriam pagos pelo valor de $0,010 por GB sendo assim se tivermos que fazer o backup de 100GB teríamos o valor de 100X0,010 = $1,00. Exportações subsequentes dos dados do mesmo snapshot não são incrementais.