

Nome da Palestra

Nome do Palestrante

Data da Palestra <formato dd.mm.yy HhhMI>

<Link do GitHub>

Índice

[1. Considerações iniciais e pré-requisitos 4](#_Toc67297776)

[Recursos usados: 4](#_Toc67297777)

[Tópicos não cobertos: 4](#_Toc67297778)

[2. Provisionamento de recursos 5](#_Toc67297779)

[2.1. Criação de VCN (Virtual Cloud Network) 5](#_Toc67297780)

[2.2. Criação de chave SSH 9](#_Toc67297781)

[2.3. Banco de Dados Oracle Database 21c (máquina virtual) 12](#_Toc67297782)

[2.4. Acessando o banco de dados pela primeira vez 15](#_Toc67297783)

[2.5. Configurando o ambiente 17](#_Toc67297784)

[3. Testando a funcionalidade In-Memory 19](#_Toc67297785)

[4. Finalizando a sessão 25](#_Toc67297786)

[4.1. Desligando banco de dados 26](#_Toc67297787)

[4.2. Destruindo o banco de dados 27](#_Toc67297788)

# Considerações iniciais e pré-requisitos

## Recursos usados:

OCI (all free tier)

* Armazenamento Oracle Object Storage
* Virtual Cloud Network
* Oracle Database Cloud Service (Virtual Machine – 30 dias gratuito)

Local (opcionais)

* Oracle SQL Developer
* Gerador de chaves SSH (sugestão PuttyGen)
* Cliente de SSH (sugestão MobaXterm)

## Tópicos não cobertos:

Instalação dos softwares na máquina host:

* Oracle SQL Developer
* Gerador de chaves SSH
* Cliente de SSH

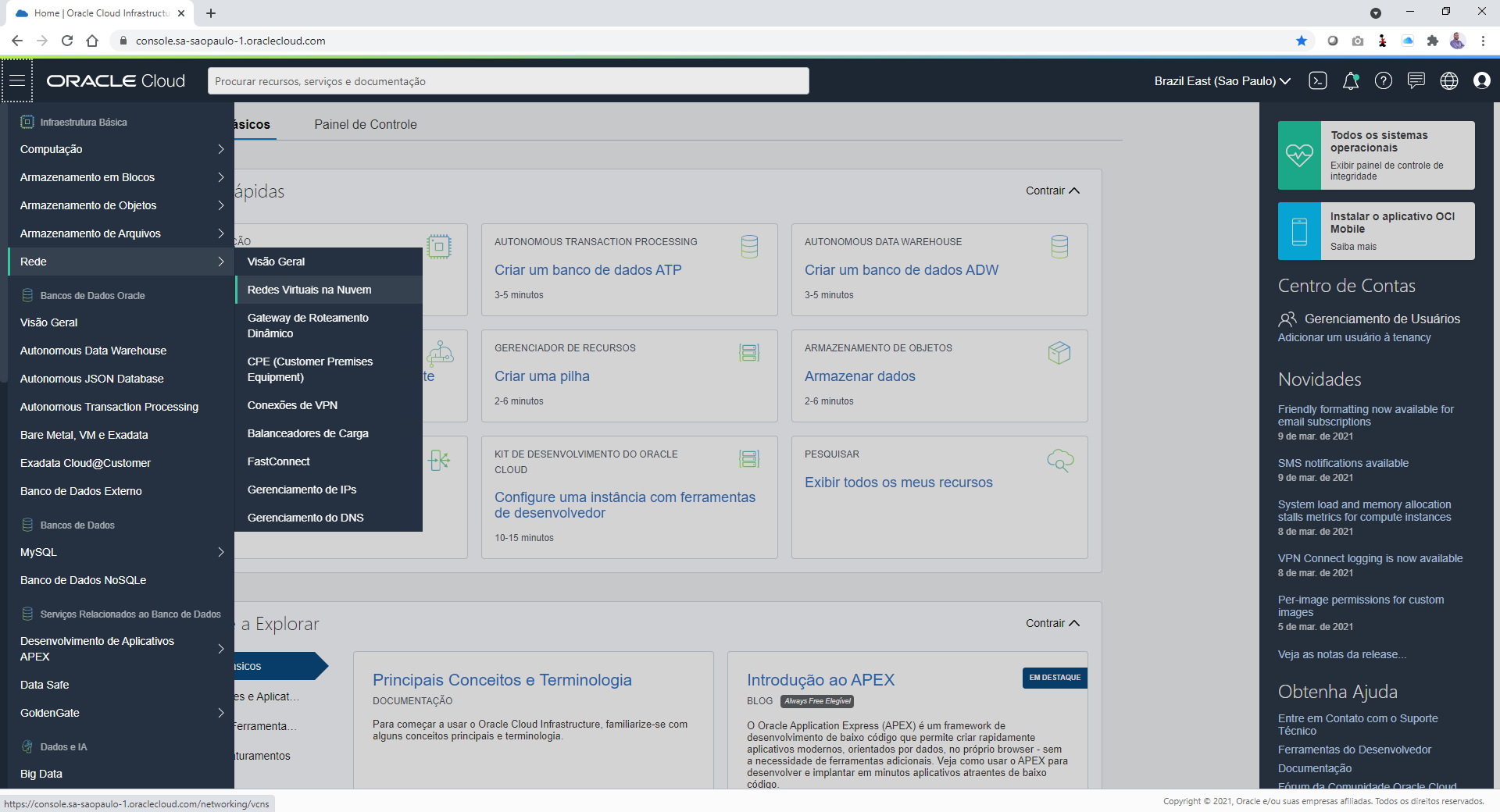
Configuração:

* Conta gratuita no Oracle Cloud

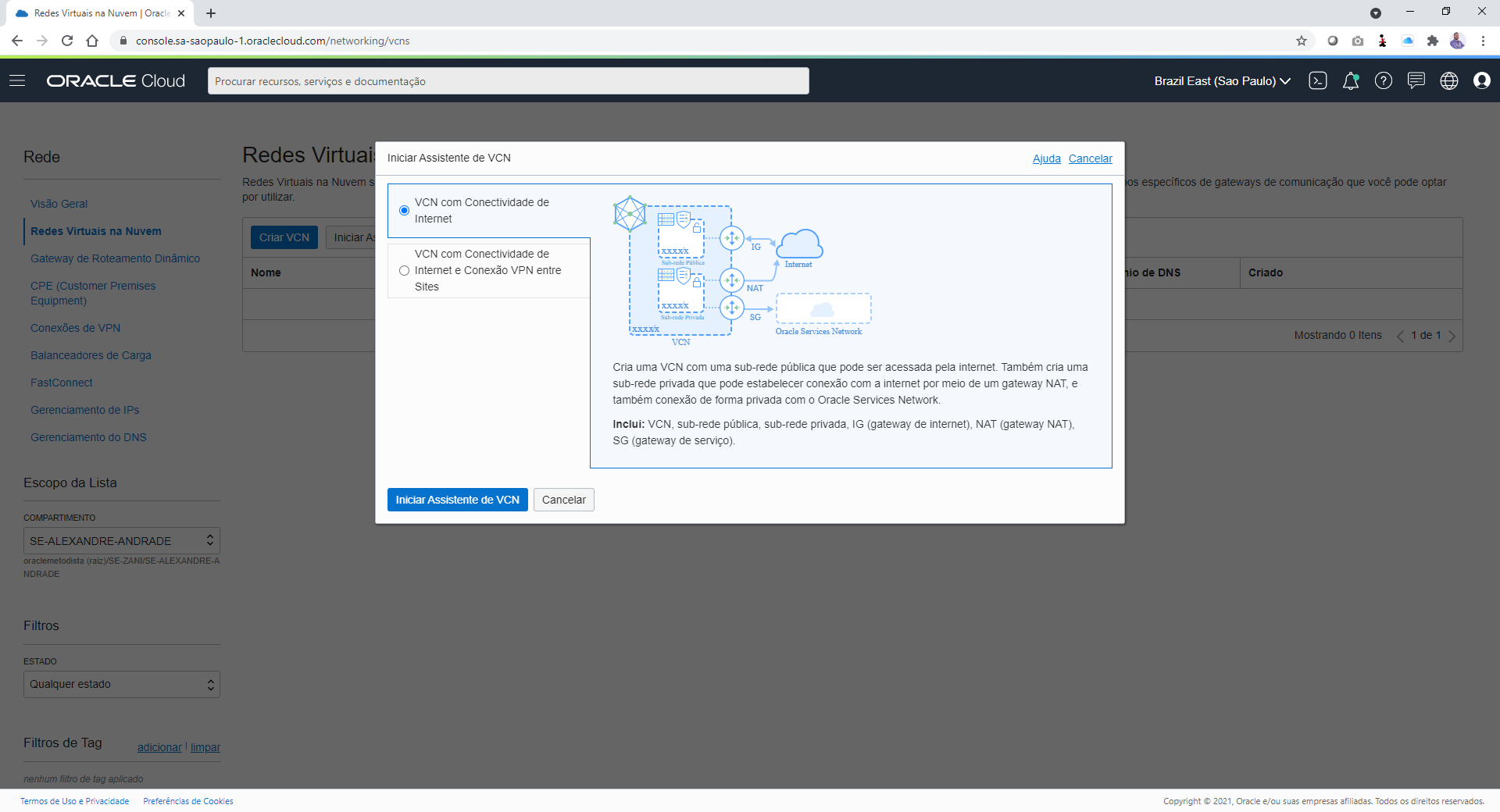
# Provisionamento de recursos

## Criação de VCN (Virtual Cloud Network)

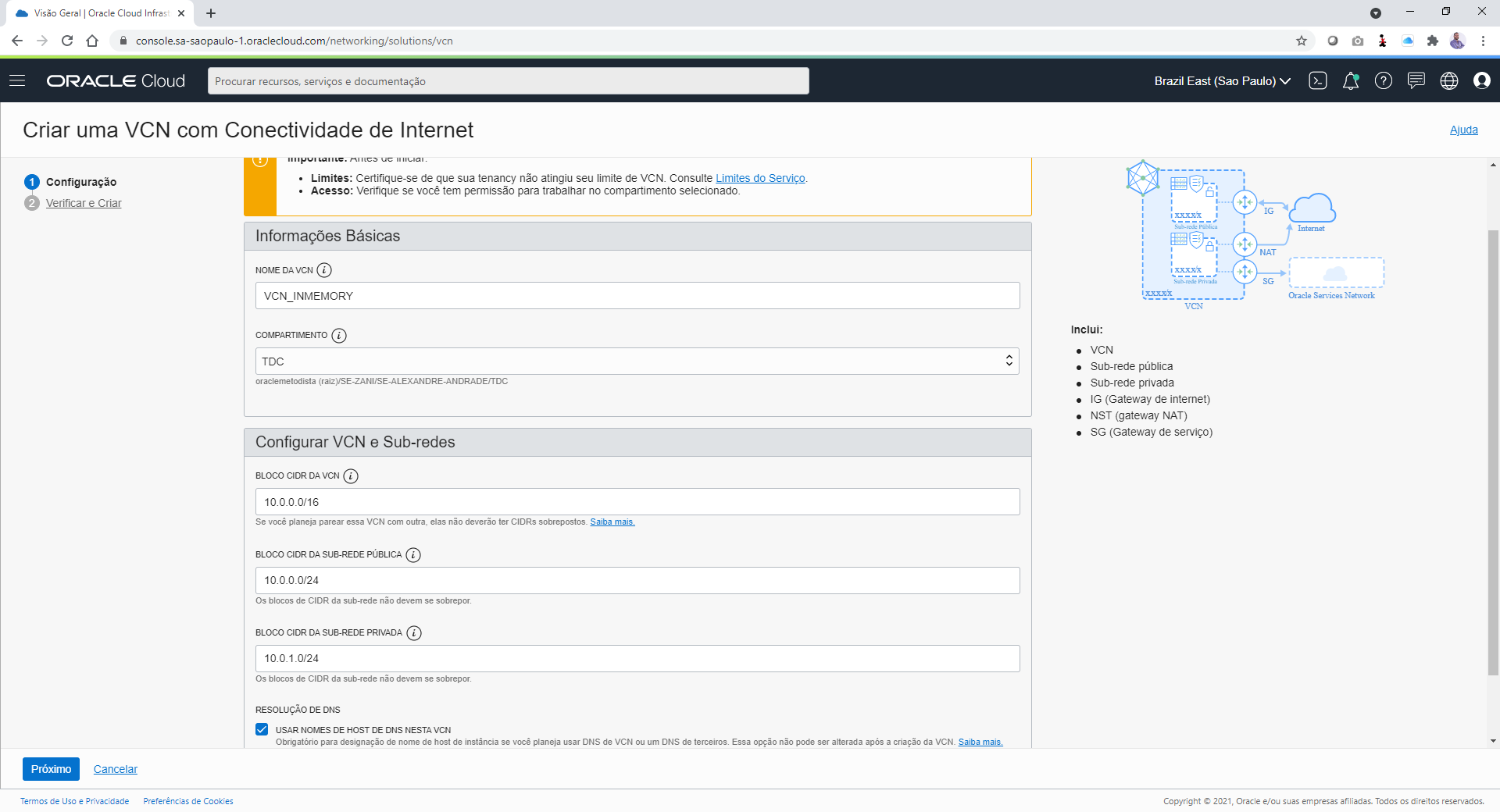
Navegue no Menu e vá até *Infraestrutura Básica -> Rede -> Redes Virtuais na Nuvem*



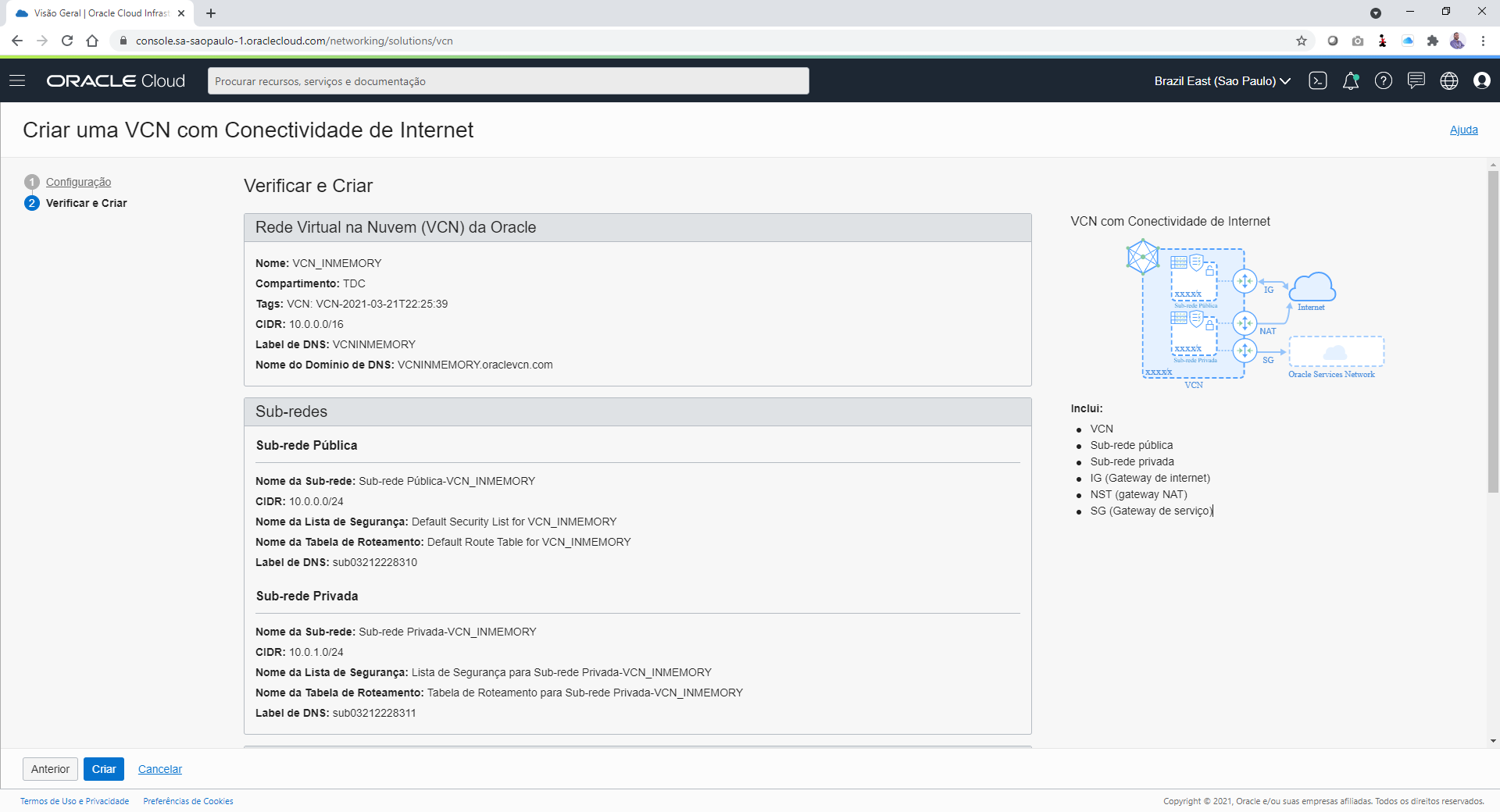
Clique no botão *Iniciar Assistente de VCN*, escolha a opção *VCN com Conectividade de Internet* e *Iniciar Assistente de VCN*.



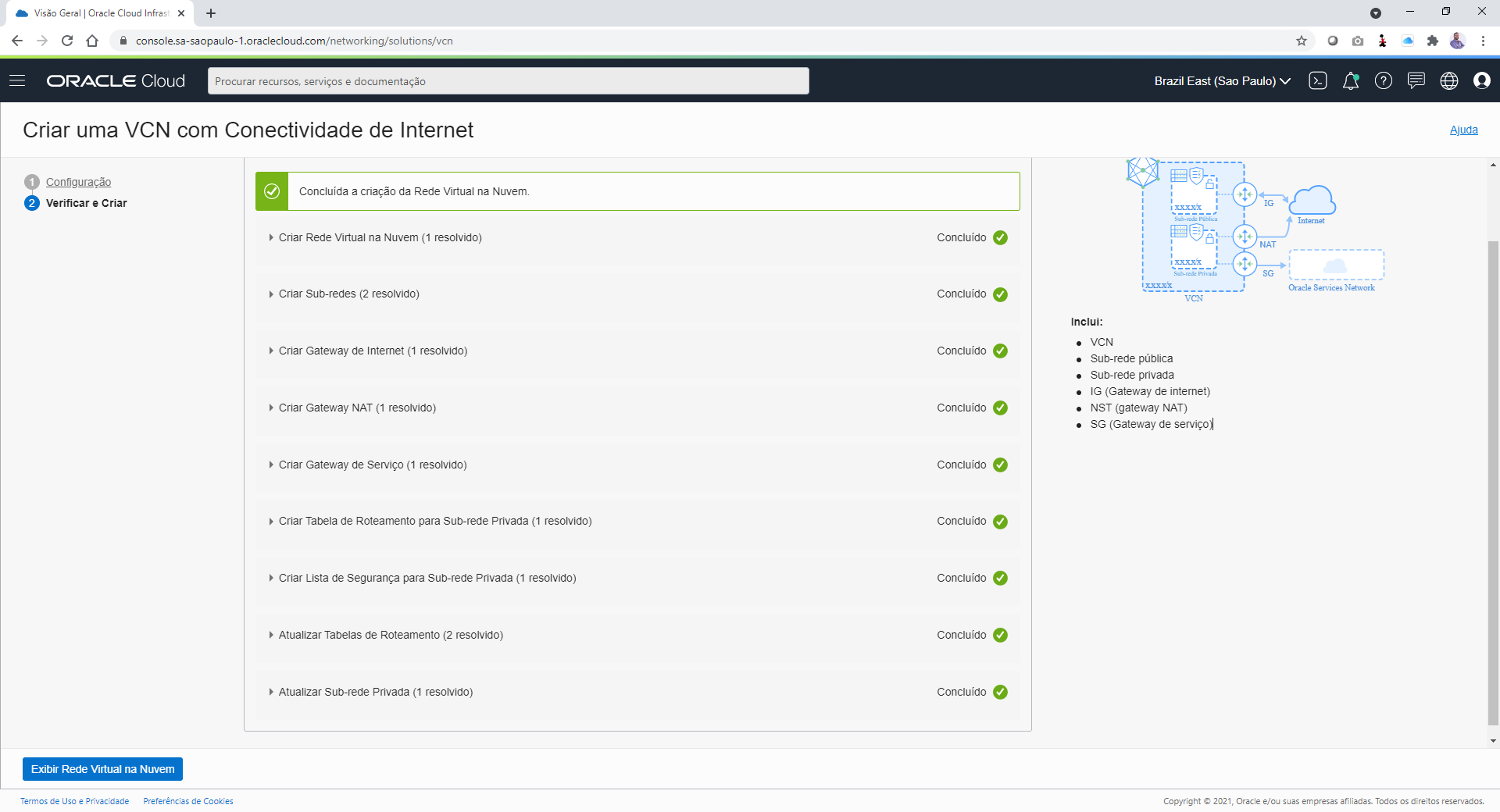
Dê um nome para a sua VCN, escolha o compartimento para criação e os blocos CIDR. Se preferir, mantenha as sugestões. Clique em *Próximo*.



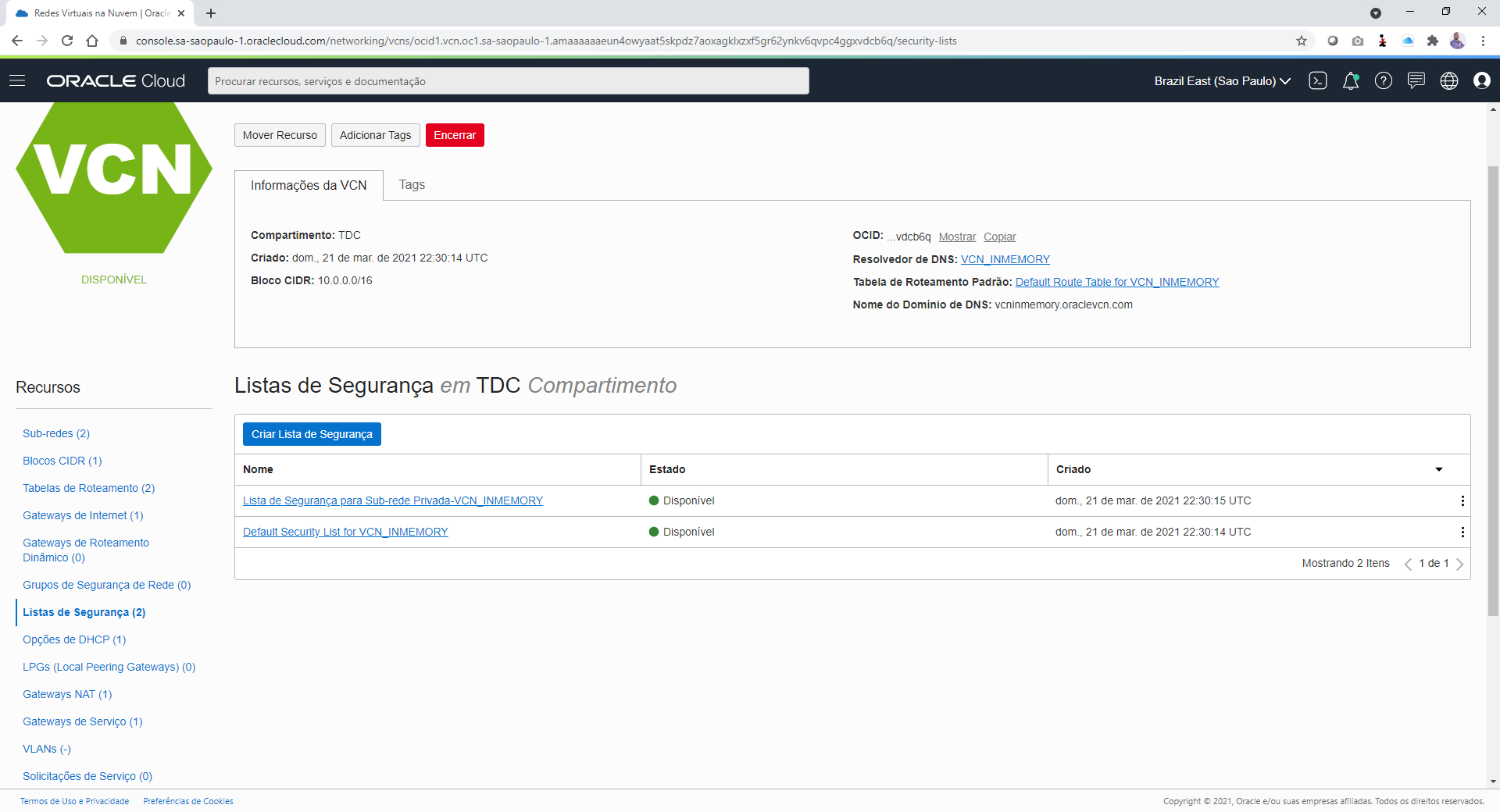
Confira as configurações e clique em *Criar*.



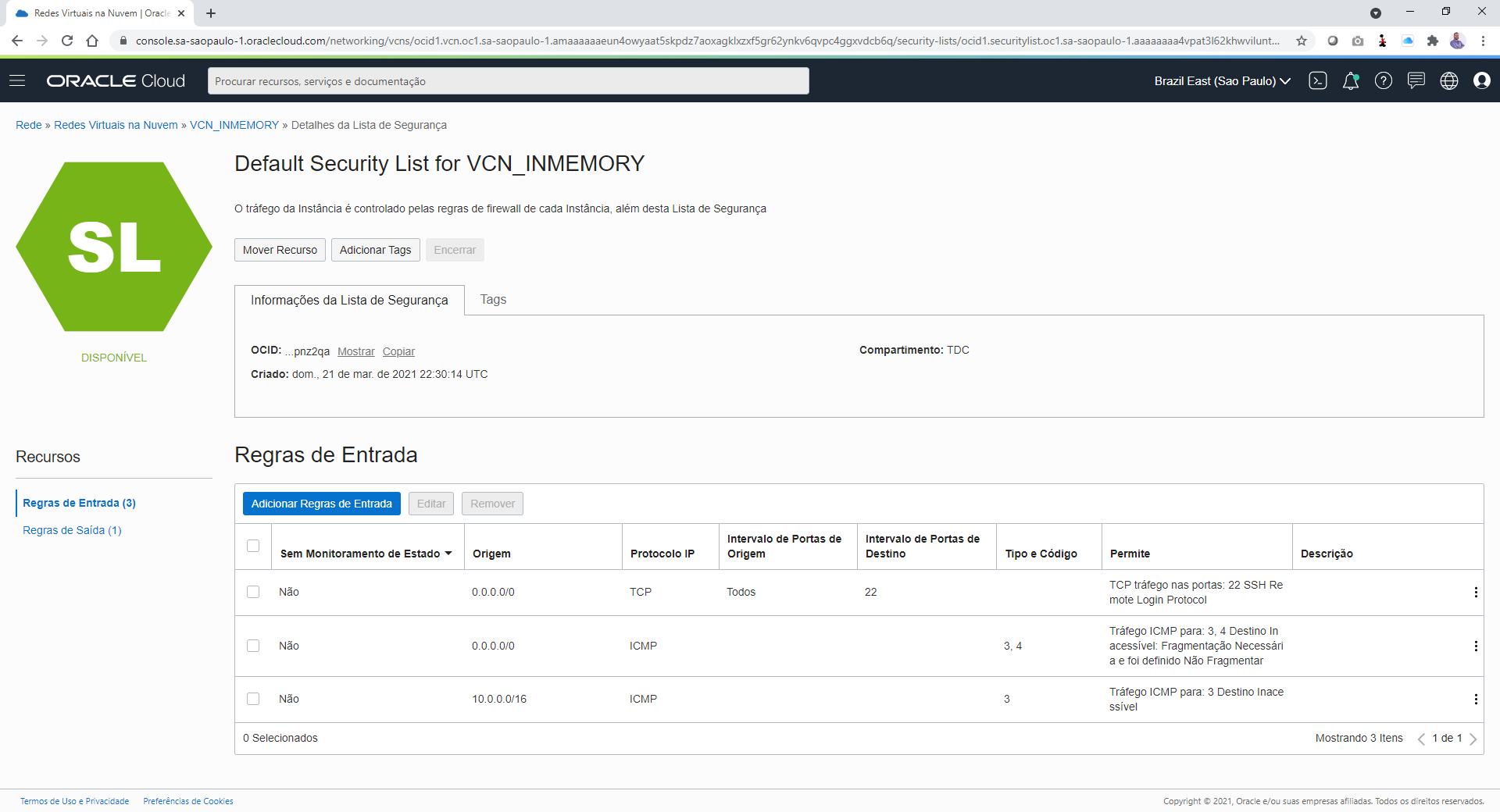
Em alguns segundos a sua rede estará provisionada. Clique no botão *Exibir Rede Virtual na Nuvem*.

**

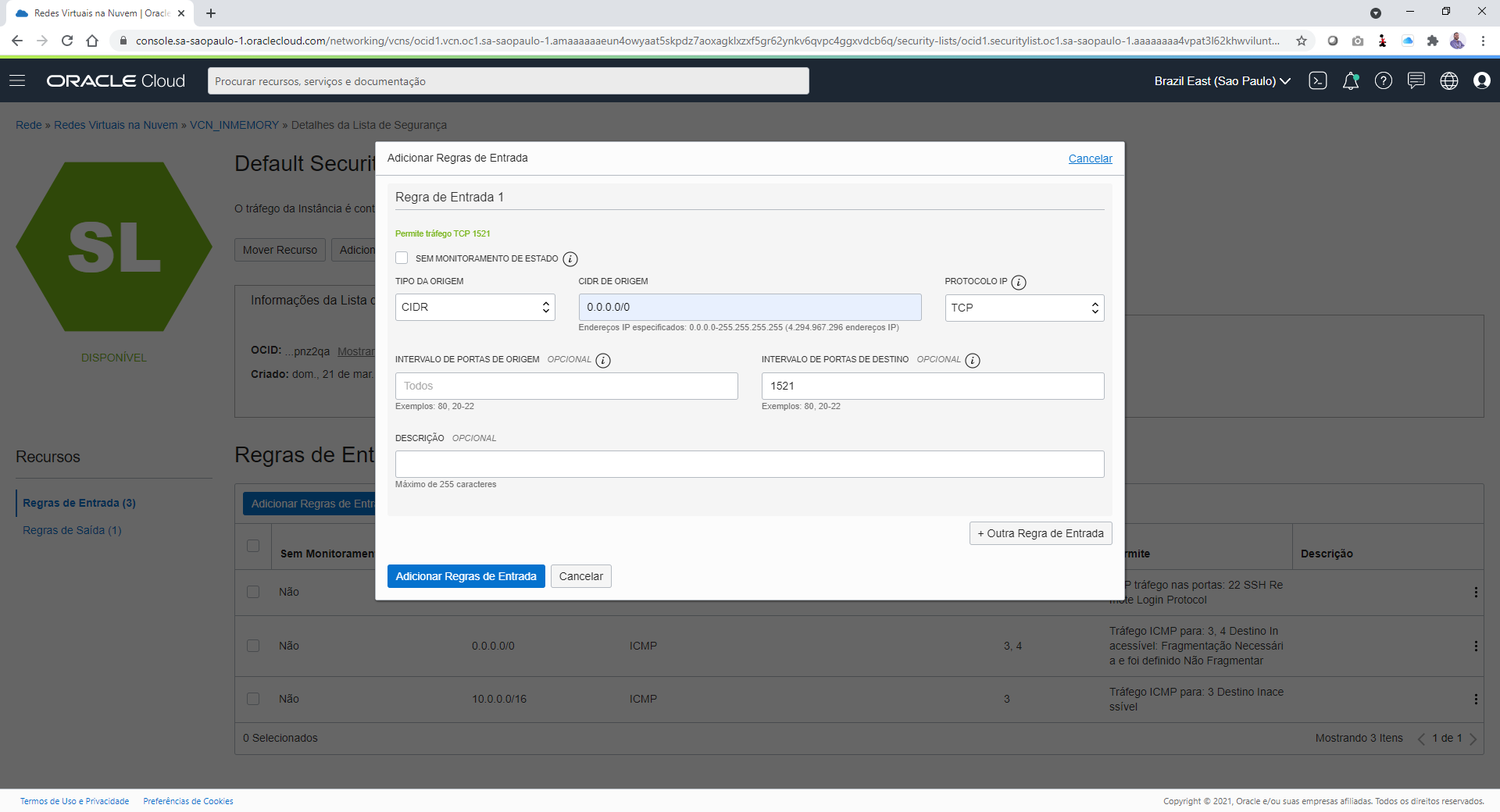
Para que seja possível acessar o banco de dados externamente, como por exemplo através do SQL Developer instalado localmente em sua máquina, será necessário abrir a porta do Listener (1521). Na lista de recursos à esquerda escolha *Lista de Segurança*.



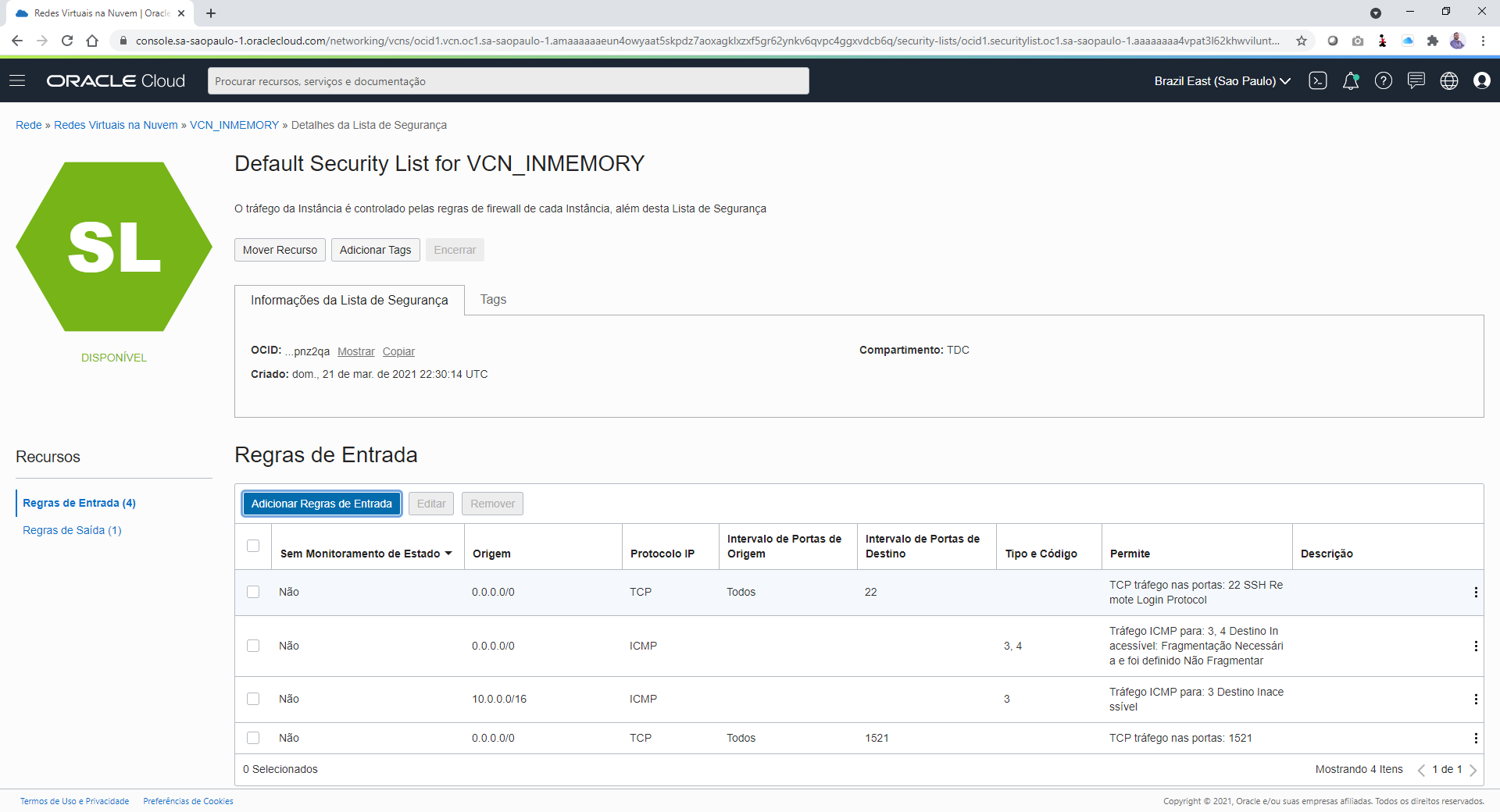
Vamos editar a SL (*Security List)* pública, clique em *Default Security List for <nome\_da\_vcn>*.



Clique no botão *Adicionar Regras de Entrada*. Abra para todos os endereços IP (*CIDR DE ORIGEM: 0.0.0.0/0*), mantenha o protocolo *TCP* e adicione a porta de destino 1521. Clique em *Adicionar Regas de Entrada*.



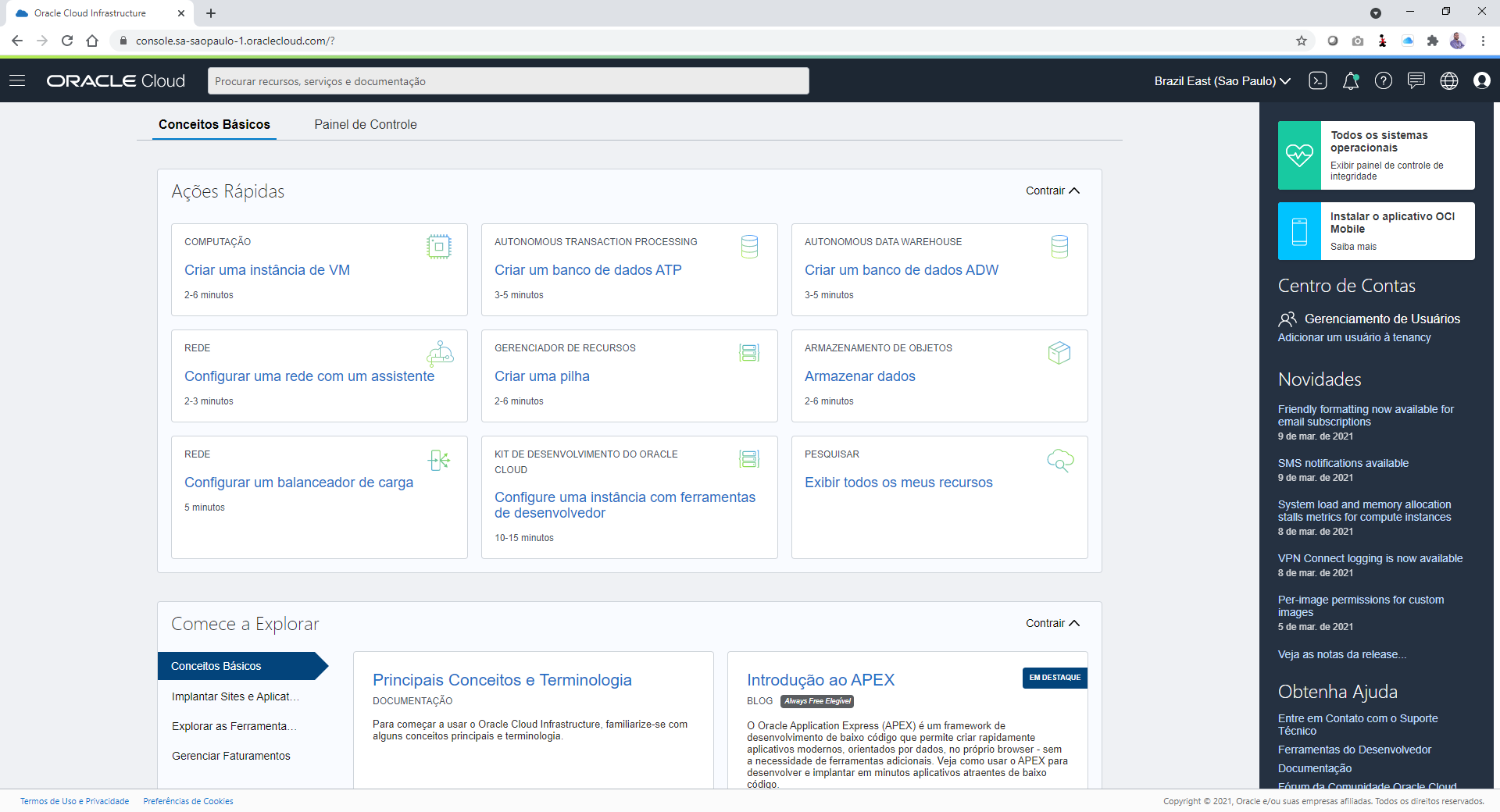
A nova regra de firewall é adicionada à SL.

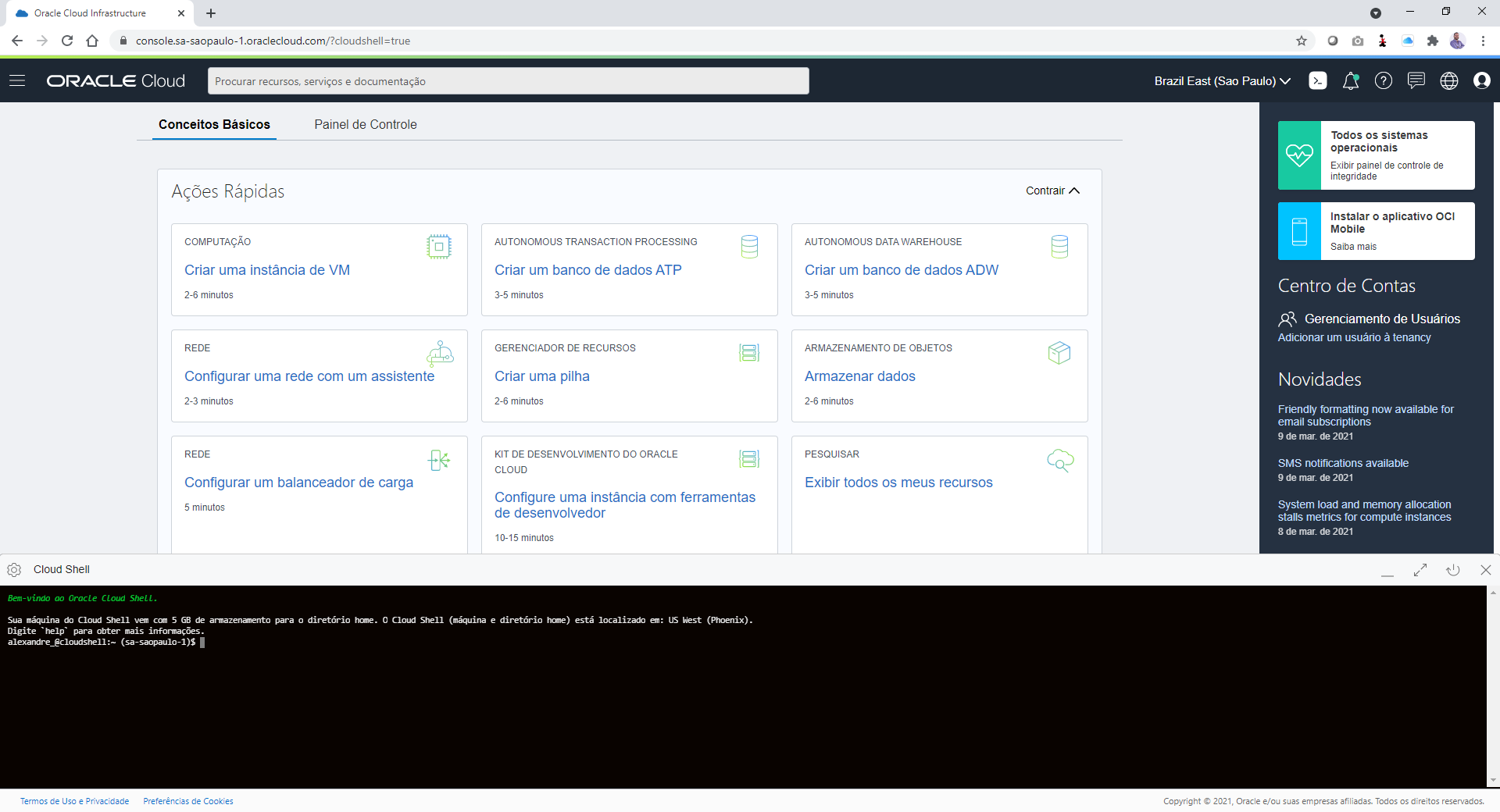


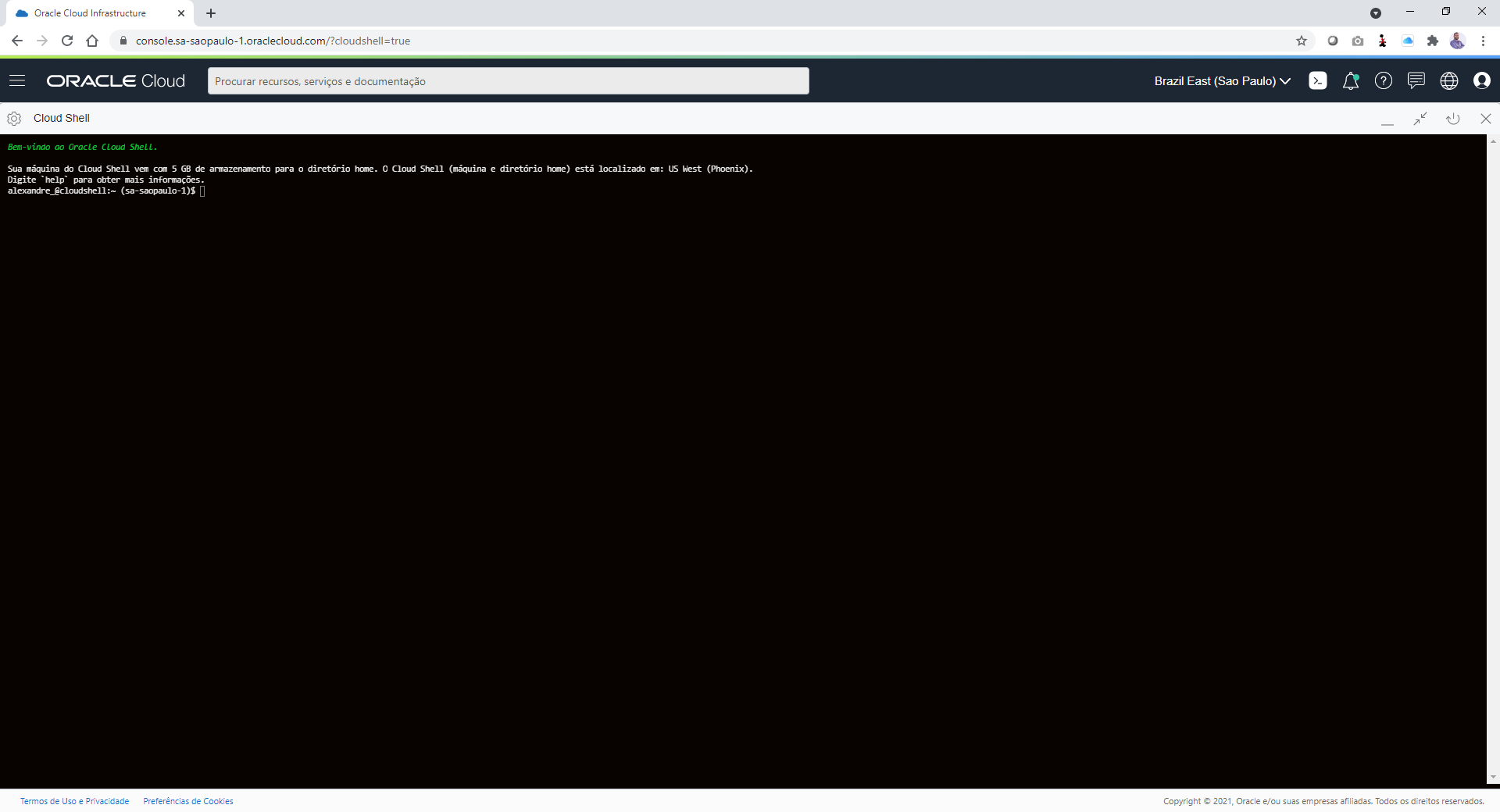
## Criação de chave SSH

Para criação e posterior acesso ao servidor de banco de dados será necessária um par de chaves SSH. Você pode usar para a criação do banco um par que você já use hoje ou criar uma nova através do PuttyGen, MobaXTerm ou ferramenta de sua preferência. Abaixo você pode seguir os passos para criação da chave usando o console do Oracle Cloud.

Clique no botão do *Cloud Shell* no canto superior direito do console do OCI.







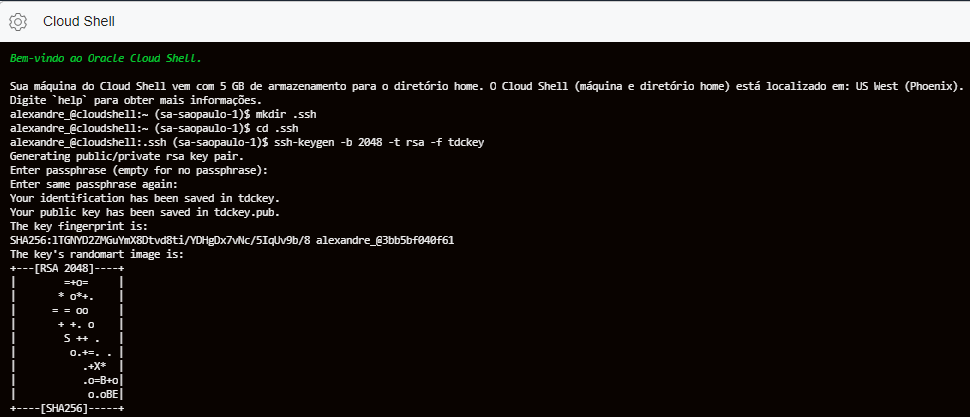
Execute os comandos abaixo para criar a sua chave. Escolha um nome fácil para lembrar depois.

$ mkdir .ssh

$ cd .ssh

$ ssh-keygen -b 2048 -t rsa -f <<nome\_da\_chave>>

**Obs.:** durante a criação será solicitado uma senha. Ela é opcional e você pode prosseguir teclando [Enter].



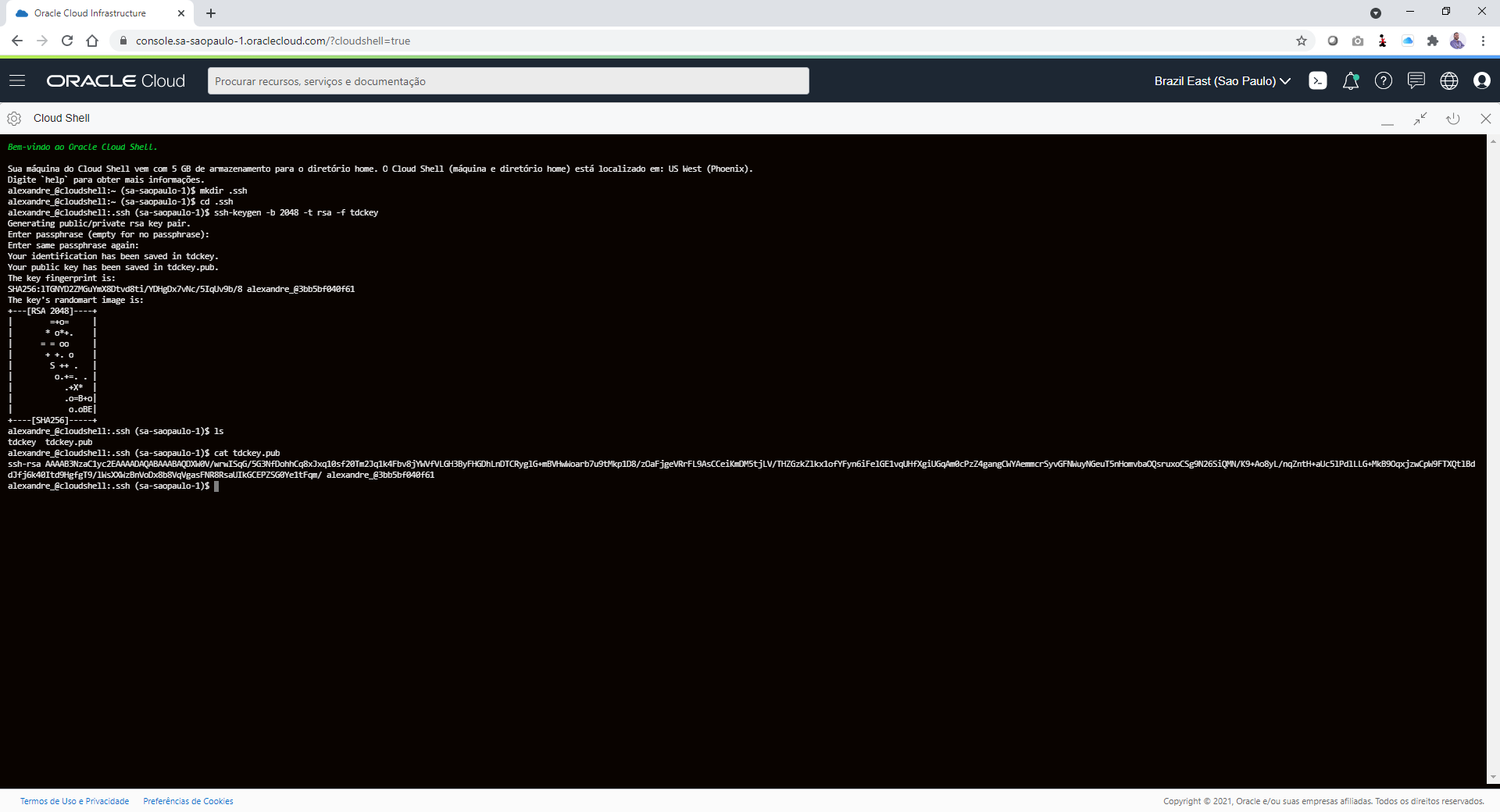
Execute o comando *ls* para verificar que as duas chaves foram criadas

$ ls

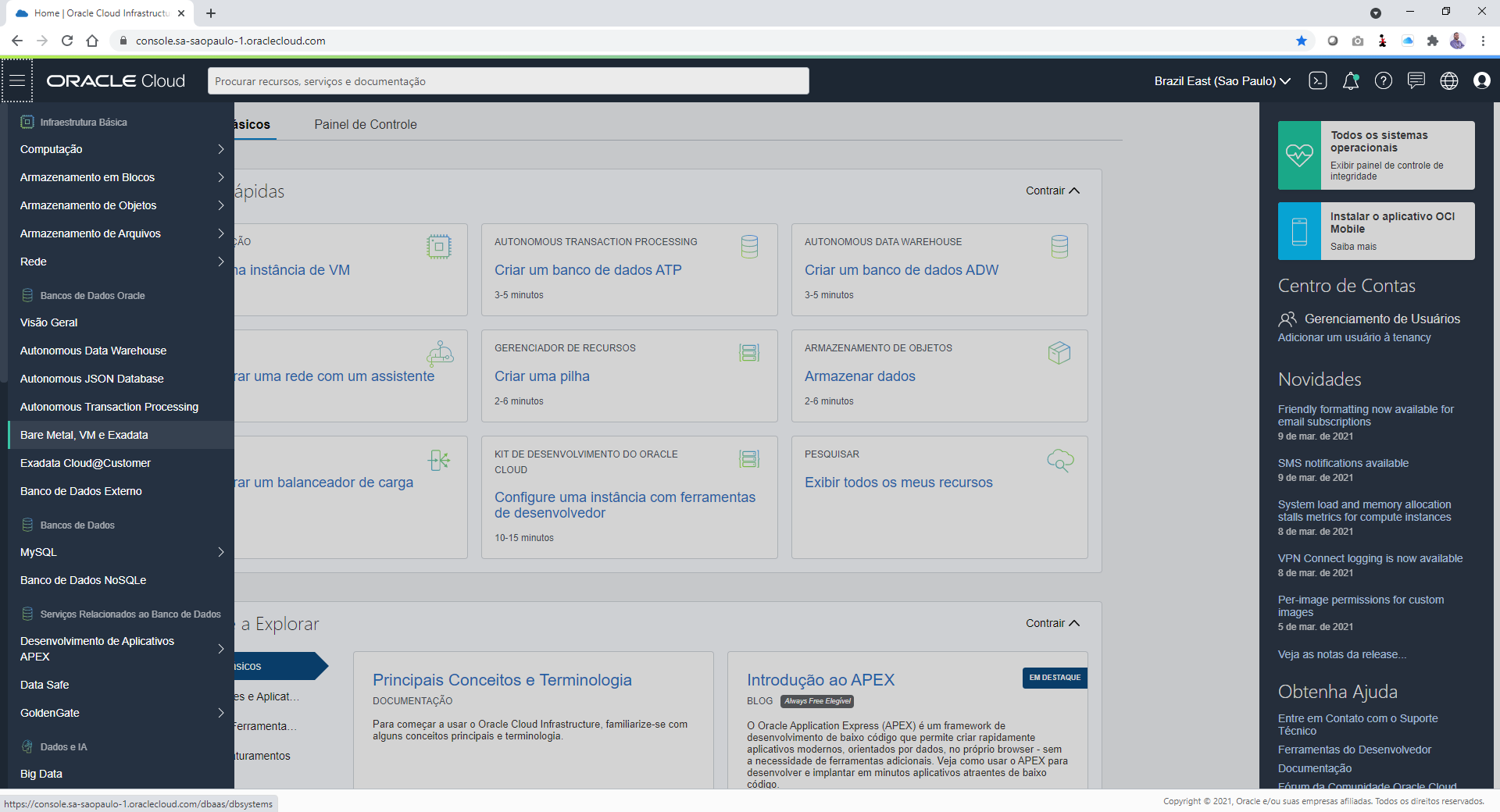


Você pode verificar o conteúdo de suas chaves através do comando *cat*. Copie e cole o conteúdo da chave pública, ela será necessária durante a criação do banco de dados.

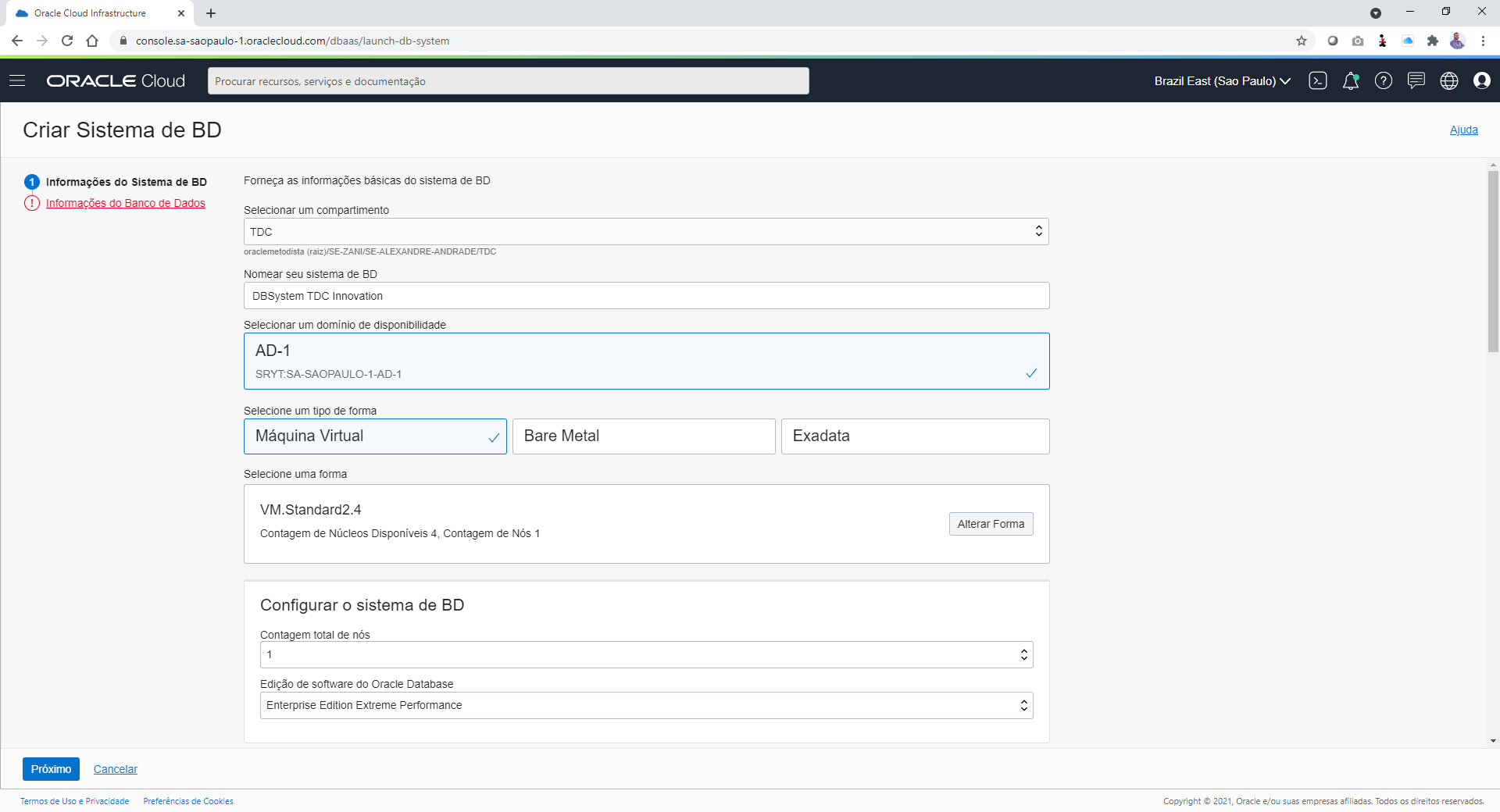
$ cat <<nome\_da\_chave>>.pub



## Banco de Dados Oracle Database 21c (máquina virtual)

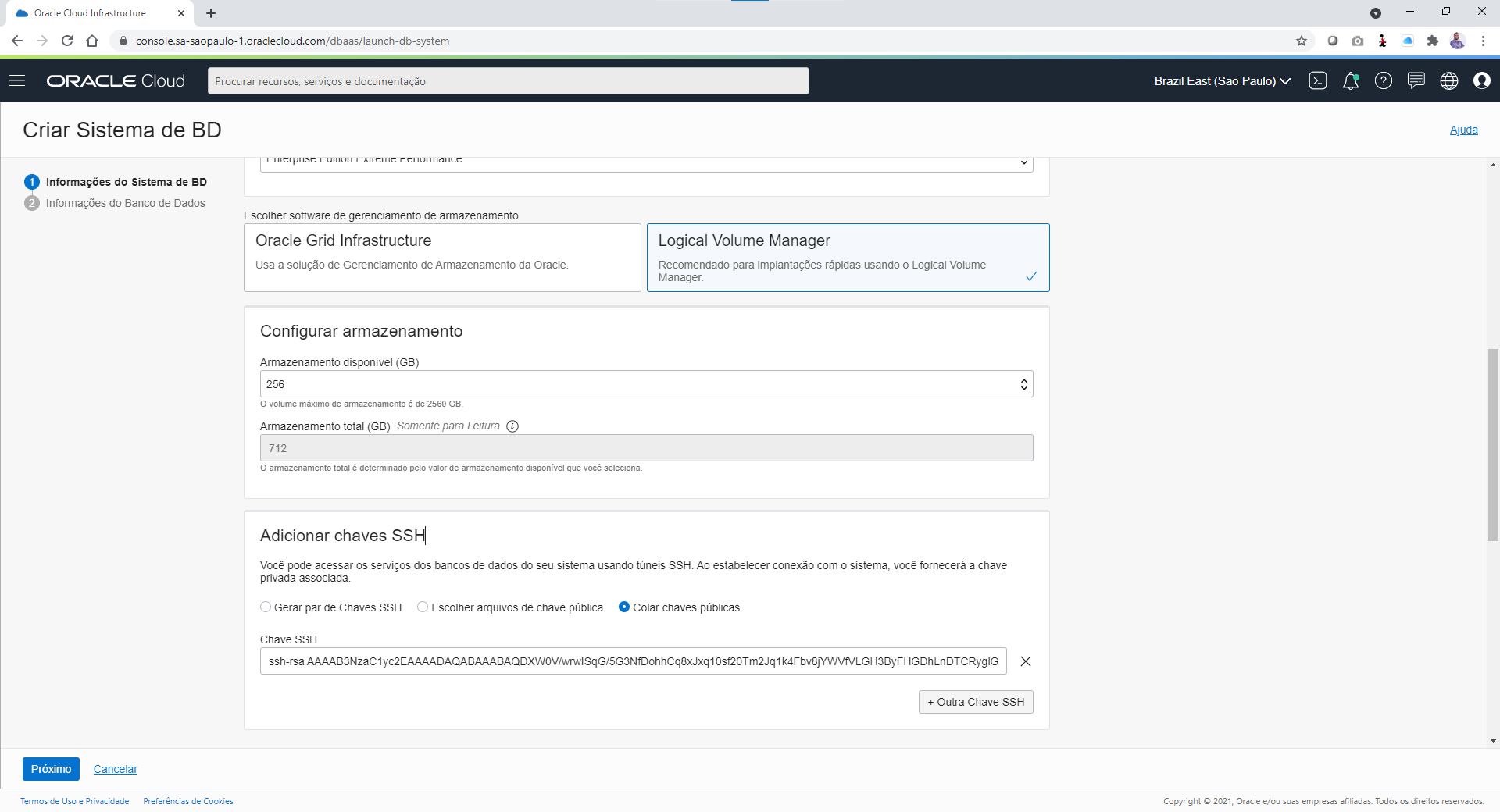
Com a rede pronta e o par de chaves SSH disponíveis, vamos provisionar o banco de dados. Clique no menu, vá para *Banco de Dados Oracle -> Bare Metal, VM e Exadata.*

Escolha o compartimento e dê um nome para o seu sistema. Mantenha o domínio de disponibilidade, o formato *Máquina Virtual* e o shape *VM. Standard2.4*. Na configuração do BD mantenha um único nó e seleciona o software *Enterprise Edition Extreme Performance*.



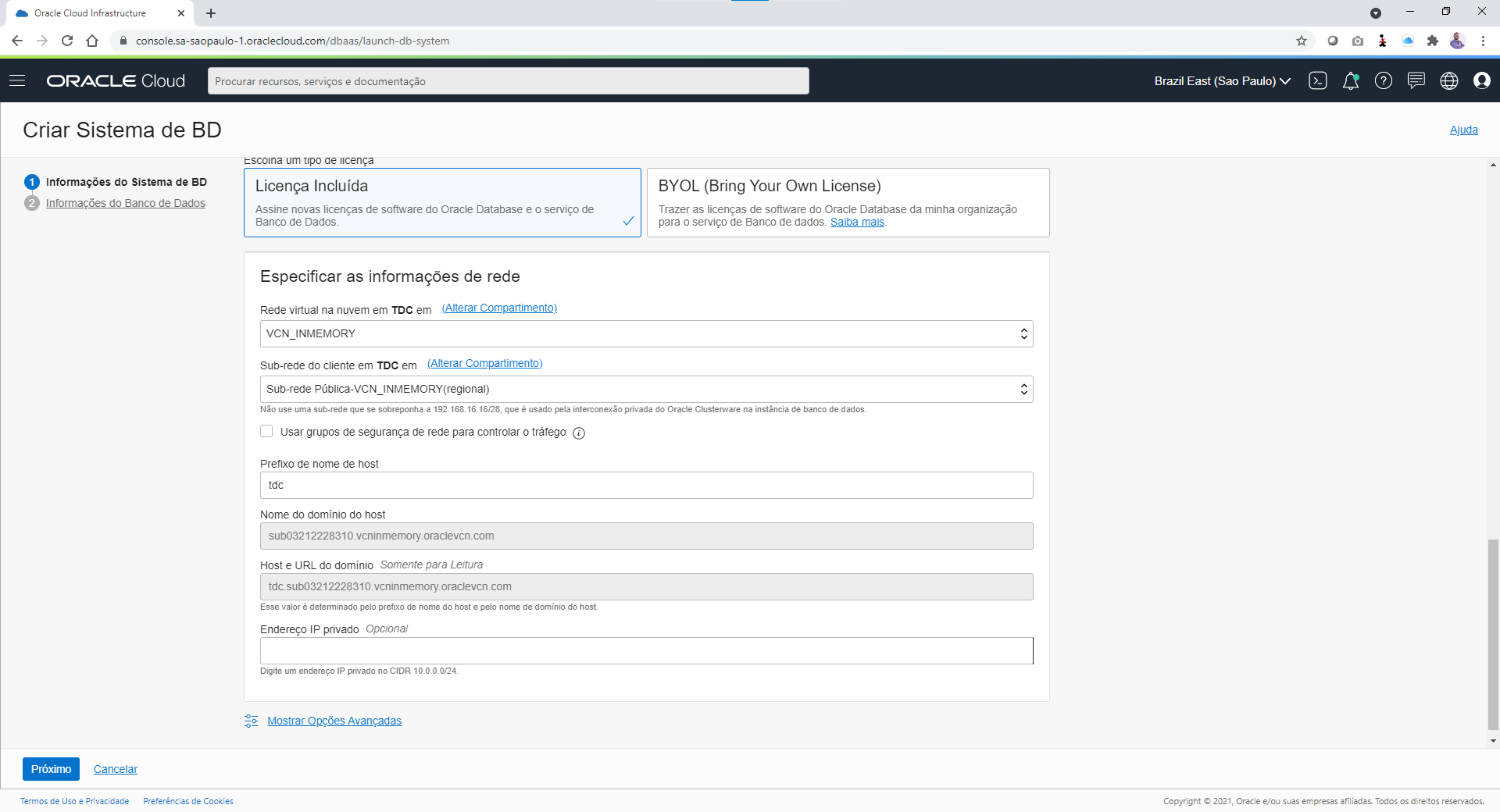
Ainda na primeira tela troque o gerenciamento de armazenamento para *Logical Volume Manager*. Não é necessário alterar as configurações de armazenamento.

Em *Adicionar chaves SSH* escolha a opção *Colar chaves públicas* e cole o valor da chave pública criada no passo anterior.



As últimas opção da tela dizem respeito ao licenciamento (escolha a opção Licença Incluída) e especifique as informações de rede, escolhendo a rede e sub-rede pública criadas anteriormente.

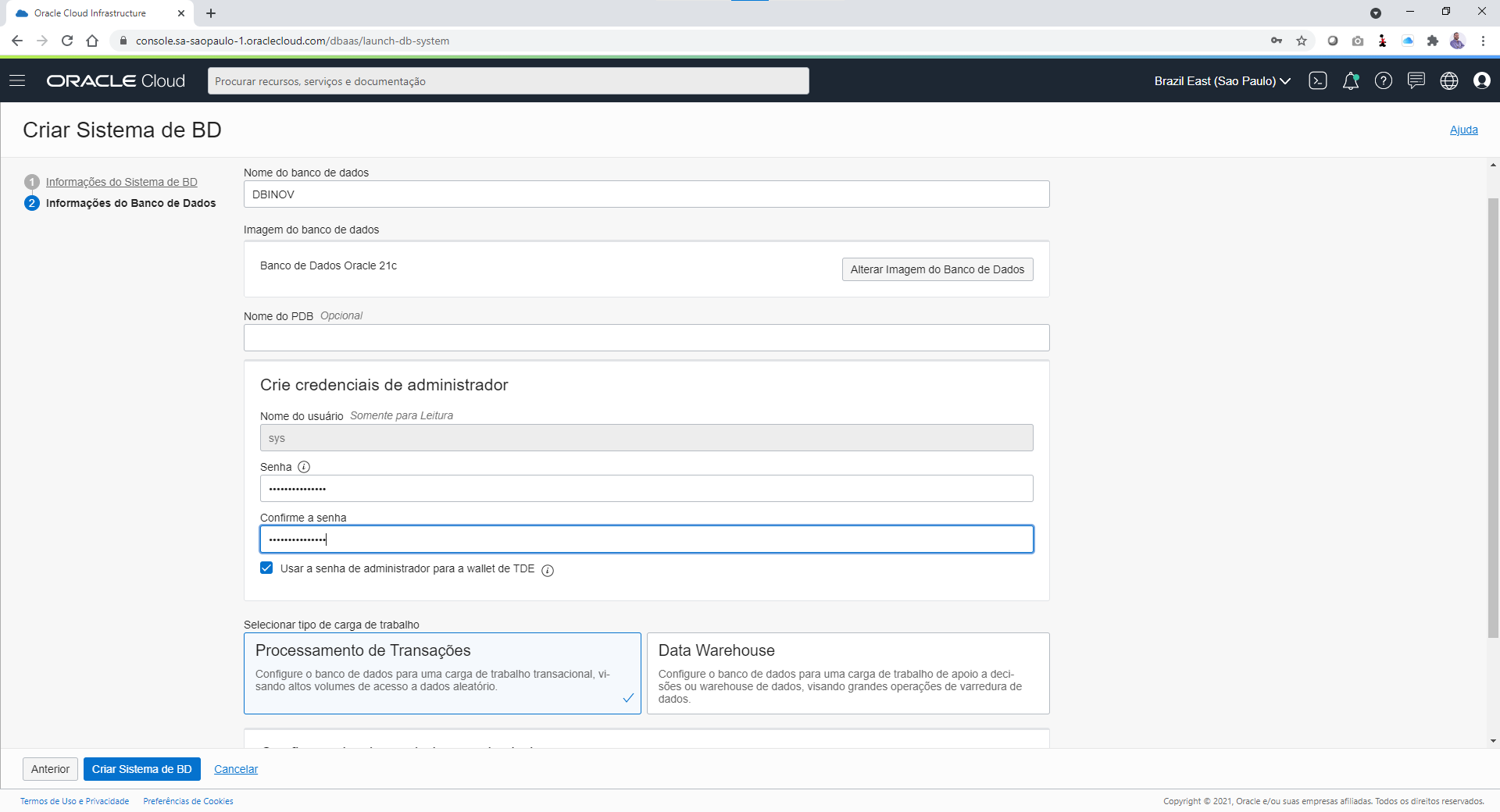
Adicione um prefixo de nome de host. Clique em *Próximo*.



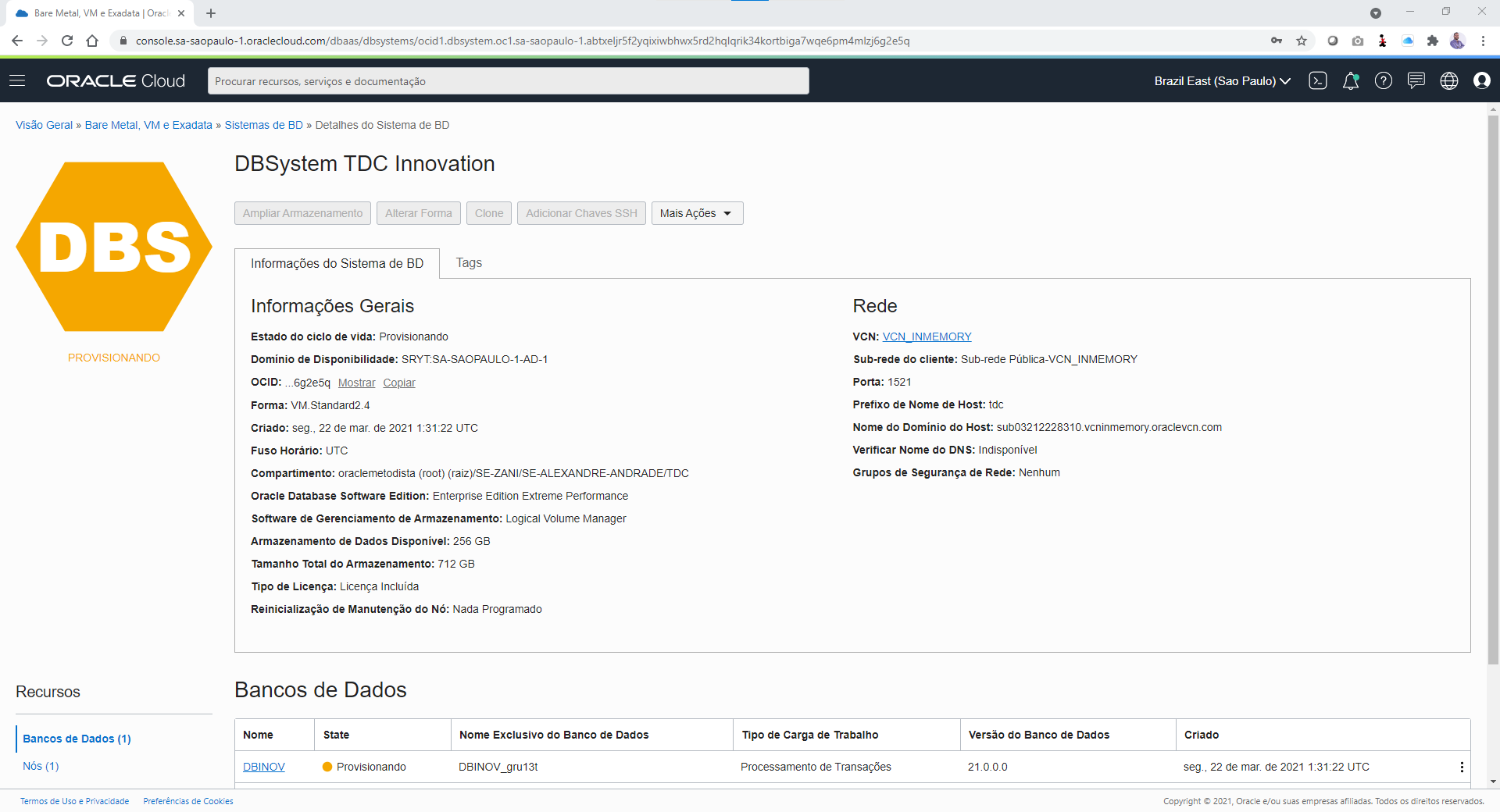
Na próxima janela você escolherá o nome para o banco de dados, a imagem e a senha do usuário sys:

* O nome do banco deve ter até 8 caracteres
* Clique no botão *Alterar Imagem do Banco de Dados* e escolha a versão 21c. Clique em *Selecionar*.
* Escolha a senha observando as restrições que se aplicam na criação. Guarde-a com cuidado.
* Para o tipo de carga selecione *Processamento de Transações*

Clique em *Criar Sistema de BD*.

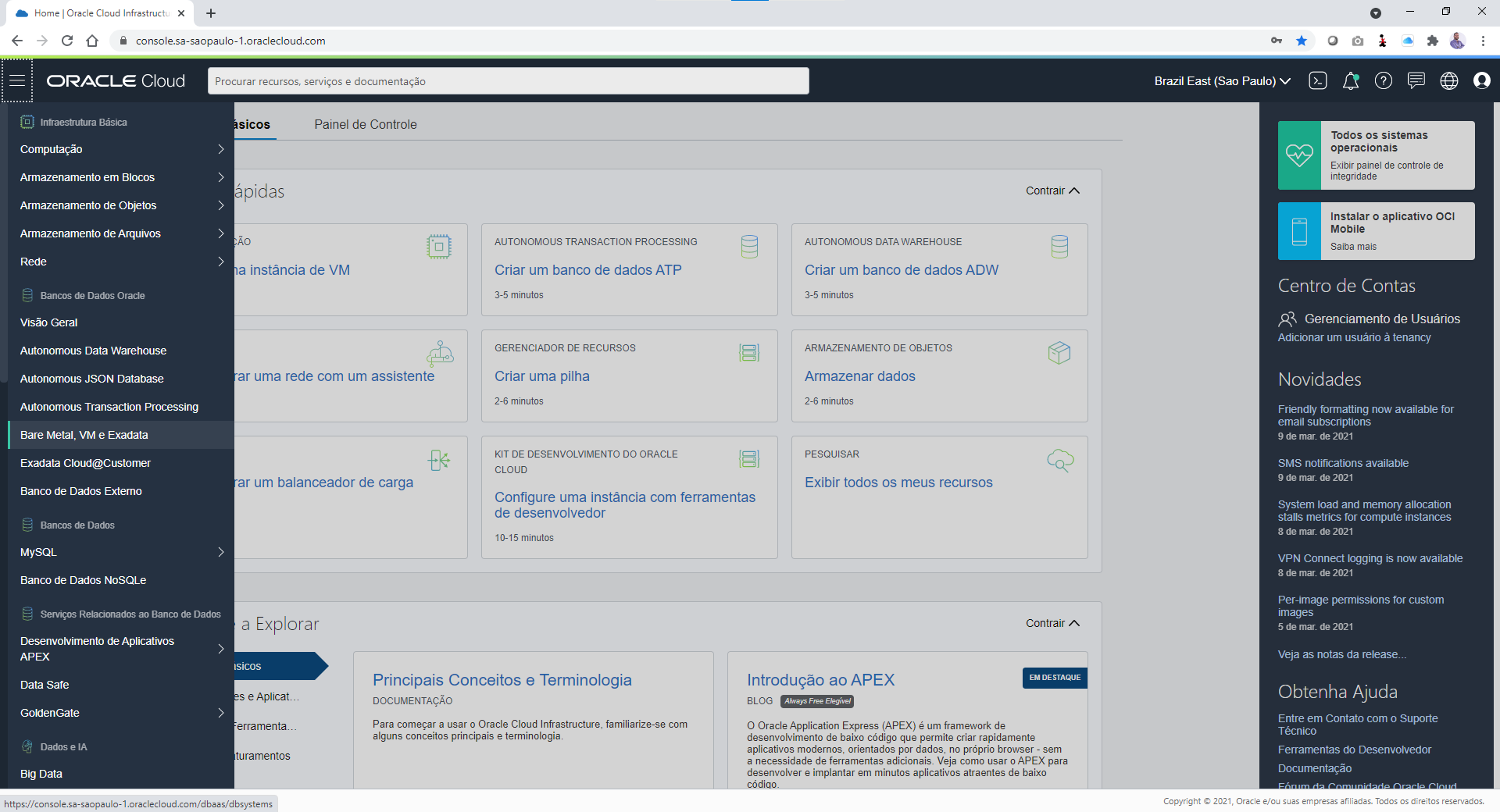


Normalmente o provisionamento leva poucos minutos, mas é normal levar um pouco mais de 1 hora para finalização do processo. Após o provisionamento você poderá acessá-lo via SQL\*Plus através de uma sessão ssh, usando o SQL Developer ou qualquer outro cliente de banco de dados.

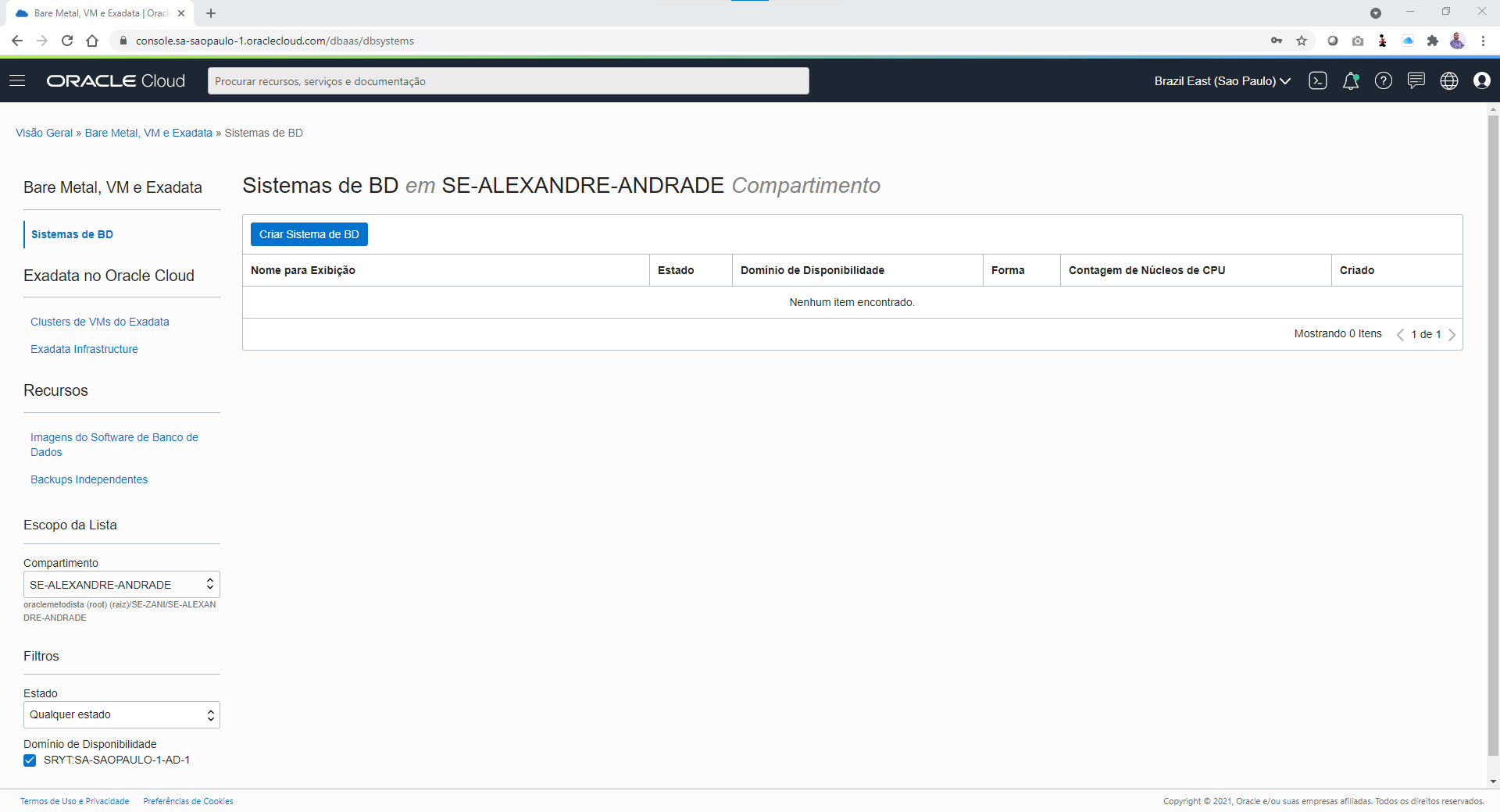


## Acessando o banco de dados pela primeira vez

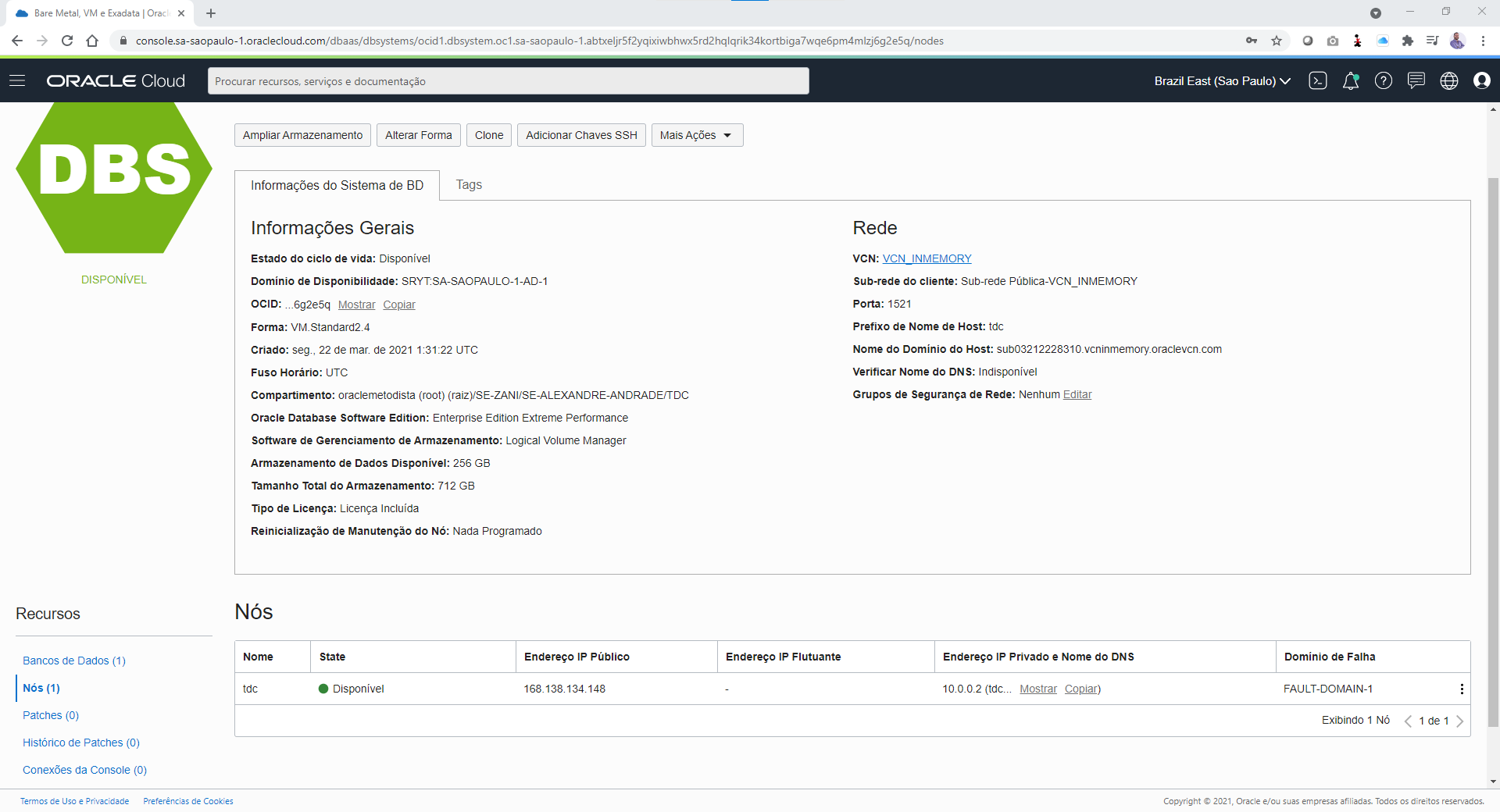
Uma vez finalizado o provisionamento do banco de dados iremos configurar o ambiente carregando as tabelas que serão utilizadas. Para este acesso precisamos do endereço IP do servidor. Vá em *Banco de Dados Oracle -> Bare Metal, VM e Exadata*



Na lista de Sistemas de BD clique no nome que você escolheu anteriormente. Se não aparecer o seu banco de dados, confirme que o compartimento correto foi selecionado.



Na janela que se abre clique em *Nós* (na lista *Recursos*, abaixo do status do banco de dados) e copie o endereço IP do servidor.



Abra novamente o *Cloud Shell* (botão à direita na barra superior do OCI) onde iremos acessar o ambiente recém-criado via SSH. Se preferir, pode usar o seu cliente de preferência localmente, lembrando que precisará ter em sua máquina as chaves públicas e privadas criadas anteriormente.

$ cd .ssh

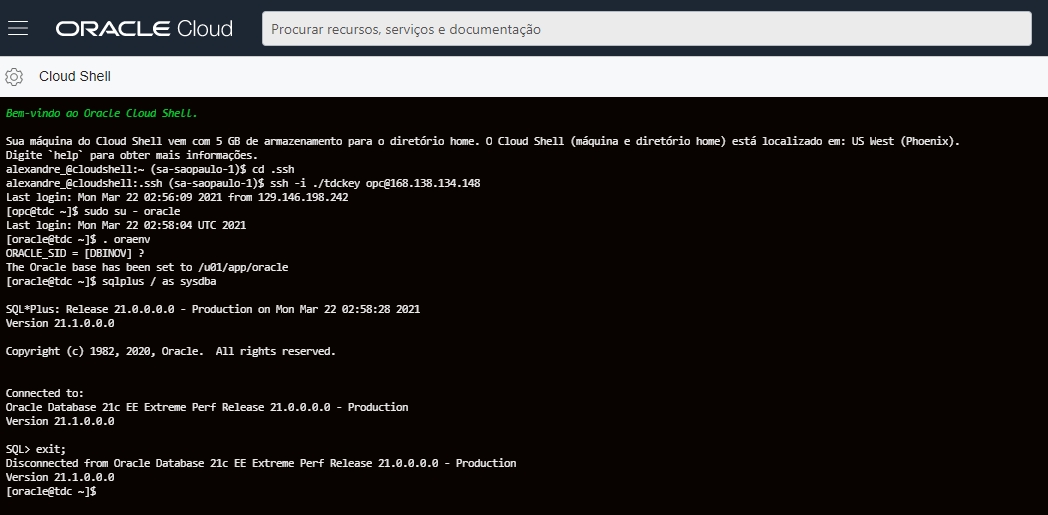
$ ssh -i ./ <<nome\_da\_chave>> opc@<<seu ip>>

$ sudo su - oracle

$ sqlplus / as sysdba

**Obs.:** No primeiro acesso será necessário confirmar a autenticidade do servidor, entre com o valor *yes* quando solicitado.

Siga os passos acima para confirmar que a conectividade ao banco de dados está funcionando conforme esperado. Na sequência pode sair do SQL\*Plus com o comando *exit;*.



## Configurando o ambiente

Agora você copiará os scripts de banco de dados para criação e carga das tabelas que utilizaremos em nossos exemplos

$ mkdir tdc

$ cd tdc

$ wget https://objectstorage.sa-saopaulo-1.oraclecloud.com/n/groyzeiavcrz/b/Labs\_ABD/o/Loader.zip

$ unzip Loader.zip

$ cd Loader

$ chmod 775 setup.sh

$ nohup ./setup.sh &

Os scripts serão executados em background, deve demorar uns 10 a 20 minutos. Pode ir pegar um café 😊.

Acesse o SQL\*Plus para conferir se a carga terminou. Se a sua sessão tiver caído desde a etapa anterior, execute os 4 passos abaixo, caso contrário, apenas o último comando.

$ cd .ssh

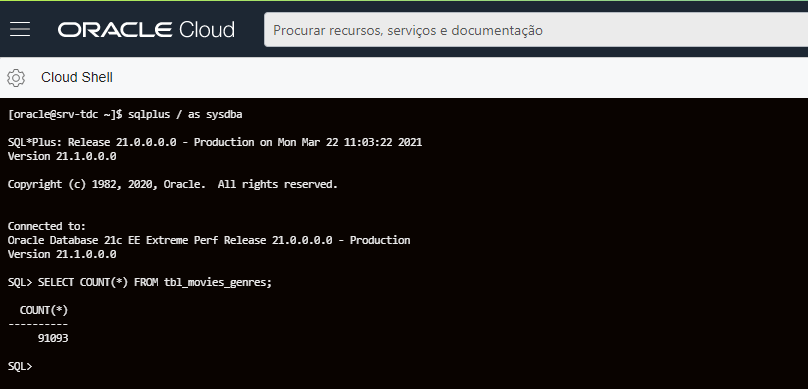
$ ssh -i ./ <<nome\_da\_chave>> opc@<<seu ip>>

$ sudo su - oracle

$ sqlplus c##tdc/TDC#inov\_2021

Uma boa maneira de verificar se a carga acabou é validando a tabela TBL\_MOVIES\_GENRES, pois é a última a ser carregada.

SELECT COUNT(\*) FROM tbl\_movies\_genres;



# Testando a funcionalidade In-Memory

A partir de agora estamos prontos para analisar os planos de execução de queries de diferentes tipos e complexidades. A sequência abaixo tem um conjunto de consultas que serão executadas uma primeira vez com o InMemory desabilitado e depois novamente com o InMemory habilitado.

Para efeito de comparação, anote os custos de execução de cada query na primeira rodada e depois anote novamente, ao reexecutar as consultas com o InMemory habilitado.

-- Aparições do personagem Darth Vader

explain plan for

SELECT DISTINCT character\_name, actor\_id

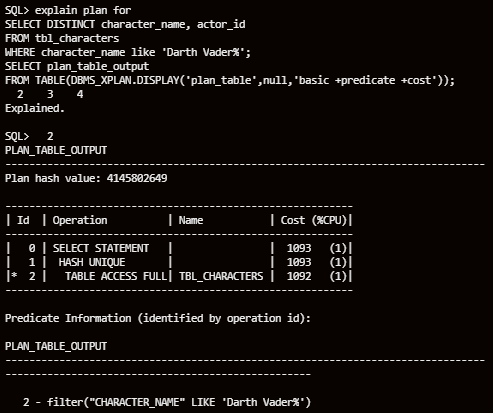
FROM tbl\_characters

WHERE character\_name like 'Darth Vader%';

SELECT plan\_table\_output

FROM TABLE(DBMS\_XPLAN.DISPLAY('plan\_table',null,'basic +predicate +cost'));

Observe que para essa primeira consulta o custo de execução foi de **1093**. Eventualmente pode haver alguma pequena diferença na sua execução



Caso esteja curioso em saber o resultado desta e das demais consultas, você pode executá-la sem o *explain plan*. Recomendo também adicionar ao final a cláusula *ROWNUM <= 10* para evitar que muitas linhas sejam carregadas na tela.

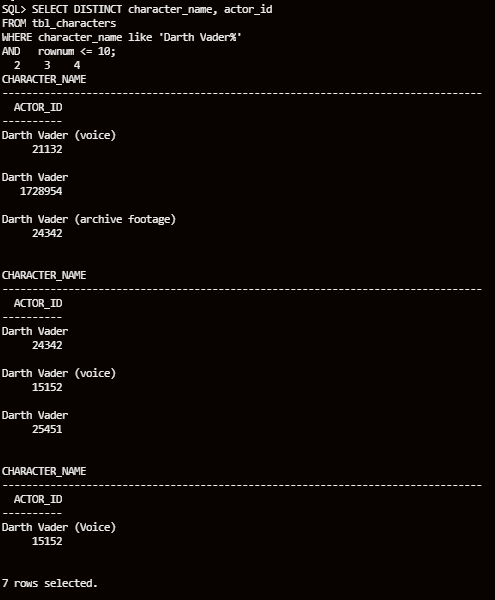
-- Aparições do personagem Darth Vader

SELECT DISTINCT character\_name, actor\_id

FROM tbl\_characters

WHERE character\_name like 'Darth Vader%'

AND rownum <= 10;

****

Execute as consultas abaixo sempre anotando o custo de execução no armazenamento em linha sem o uso do InMemory.

-- Atores que interpretaram Darth Vader

explain plan for

SELECT DISTINCT a.actor\_id,

a.actor\_name

FROM tbl\_characters c,

tbl\_actors a

WHERE c.actor\_id = a.actor\_id

AND c.character\_name like 'Darth Vader%';

SELECT plan\_table\_output

FROM TABLE(DBMS\_XPLAN.DISPLAY('plan\_table',null,'basic +predicate +cost'));

-- Quantos personagens cada um desses atores interpretou

explain plan for

SELECT a.actor\_id,

a.actor\_name,

count(\*) qtde\_personagens

FROM tbl\_characters c,

tbl\_actors a

WHERE c.actor\_id = a.actor\_id

AND a.actor\_id in (15152, 21132, 25451, 1728954, 1728453, 24342)

GROUP BY a.actor\_id, a.actor\_name

ORDER BY qtde\_personagens DESC;

SELECT plan\_table\_output

FROM TABLE(DBMS\_XPLAN.DISPLAY('plan\_table',null,'basic +predicate +cost'));

Perceba que agora iniciaremos queries com características analíticas, consultando diversas tabelas, com maior número de linhas e agregações. Consequentemente o custo associado também será maior.

-- Quais os filmes que o intérprete do Darth Vader que mais fez personagens já fez?

explain plan for

SELECT mm.original\_title,

c.character\_name,

round(to\_number(mm.popularity, '999D9999999999', 'NLS\_NUMERIC\_CHARACTERS = ''.,'''), 2) popularidade,

to\_date(release\_date, 'dd/mm/yy') data\_lancamento

FROM tbl\_characters c,

tbl\_movies\_characters mc,

stg\_movies\_metadata mm

WHERE c.character\_id = mc.character\_id

AND mc.movie\_id = mm.id

AND c.actor\_id = 15152

ORDER BY popularidade DESC;

SELECT plan\_table\_output

FROM TABLE(DBMS\_XPLAN.DISPLAY('plan\_table',null,'basic +predicate +cost'));

A consulta abaixo adiciona ainda mais complexidade, usando a maior tabela do banco, STG\_RATINGS, que tem 26.024.289 registros.

-- Qual a nota média para os filmes interpretados por James Earl Jones

explain plan for

SELECT mm.original\_title,

c.character\_name,

round(to\_number(mm.popularity, '999D9999999999', 'NLS\_NUMERIC\_CHARACTERS = ''.,'''), 2) popularidade,

to\_date(release\_date, 'dd/mm/yy') data\_lancamento,

round(avg(to\_number(r.rating, '999D99', 'NLS\_NUMERIC\_CHARACTERS = ''.,''')), 2) nota\_media

FROM tbl\_characters c,

tbl\_movies\_characters mc,

stg\_movies\_metadata mm,

stg\_ratings r

WHERE c.character\_id = mc.character\_id

AND mc.movie\_id = mm.id

AND c.actor\_id = 15152

AND mm.id = r.movieid

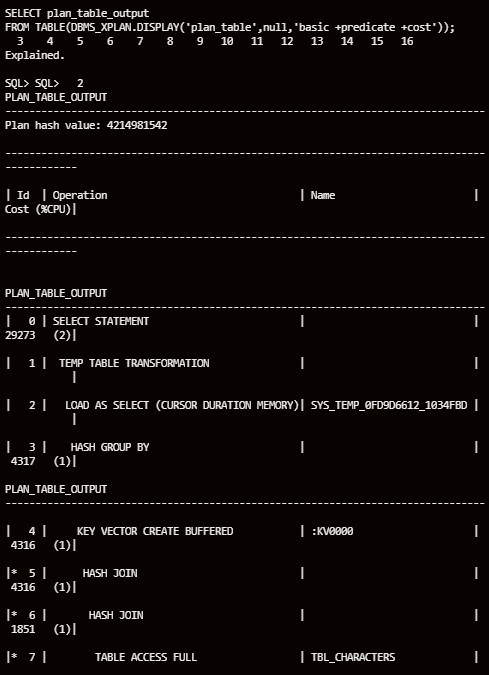
GROUP BY mm.original\_title, c.character\_name, round(to\_number(mm.popularity, '999D9999999999', 'NLS\_NUMERIC\_CHARACTERS = ''.,'''), 2), to\_date(release\_date, 'dd/mm/yy')

ORDER BY popularidade DESC;

SELECT plan\_table\_output

FROM TABLE(DBMS\_XPLAN.DISPLAY('plan\_table',null,'basic +predicate +cost'));

Consequentemente o custo também é bastante alto. Note que também há diversas indicações *TABLE ACCESS FULL*, o que coopera substancialmente para o custo. Tipicamente a solução para minimizar esse custo seria a criação de índices, entretanto quanto mais índices nas tabelas, pior a performance nas operações transacionais.



Agora vamos habilitar a funcionalidade InMemory do banco de dados. Este não é um comando dinâmico, portanto será necessário reiniciar a instância. Vamos configurar a InMemory área para usar 10GB.

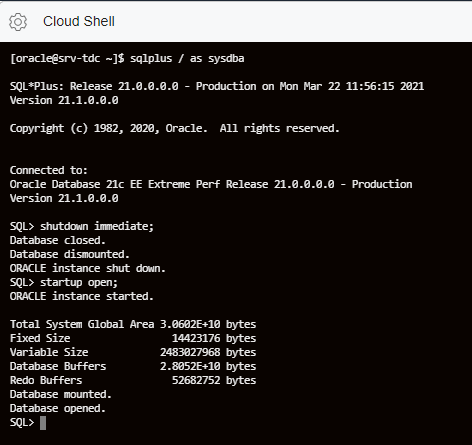
alter system set inmemory\_size=10G scope=spfile;

conn / as sysdba

shutdown immediate;

startup open;

conn c##tdc/TDC#inov\_2021



Vamos colocar as tabelas individualmente em memória com prioridade *HIGH*. Dessa maneira elas começam a ser atribuídas imediatamente à *InMemory Area*. Se mantivermos a opção padrão a tabela só será alocada em memória quando a primeira consulta for realizada.

ALTER TABLE tbl\_characters INMEMORY PRIORITY HIGH;

ALTER TABLE tbl\_actors INMEMORY PRIORITY HIGH;

ALTER TABLE tbl\_movies\_characters INMEMORY PRIORITY HIGH;

ALTER TABLE stg\_movies\_metadata INMEMORY PRIORITY HIGH;

ALTER TABLE stg\_ratings INMEMORY PRIORITY HIGH;

Podemos acompanhar a alocação em memória pela query abaixo. Note que levará alguns segundos até tudo subir. Nessa consulta também é possível observar o tamanho ocupado em disco pela tabela e o quanto ela ocupa em memória.

select owner,

segment\_name,

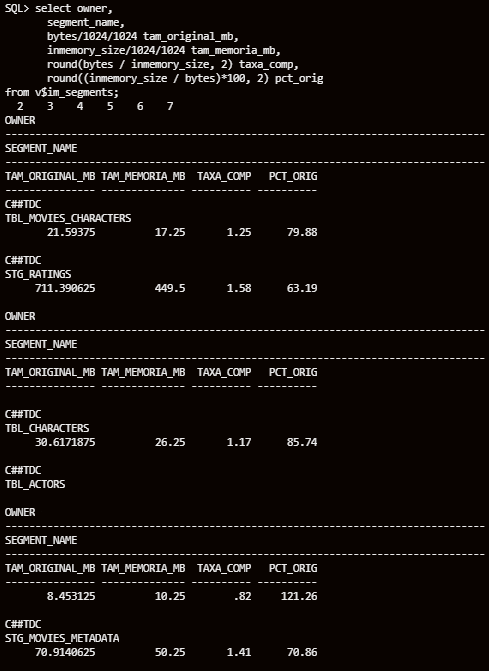
bytes/1024/1024 tam\_original\_mb,

inmemory\_size/1024/1024 tam\_memoria\_mb,

round(bytes / inmemory\_size, 2) taxa\_comp,

round((inmemory\_size / bytes)\*100, 2) pct\_orig

from v$im\_segments;



Agora execute novamente todos as queries acima e anote os novos custos de execução. Você terá algo parecido com o apresentado abaixo:

|  |  |  |  |
| --- | --- | --- | --- |
| **Consulta** | **Custo em linha / disco** | **Custo em colunar / memória** | **Ganho de Performance** |
| 1 | 1093 | 51 | 95,33% |
| 2 | 1399 | 66 | 95,28% |
| 3 | 1408 | 72 | 94,89% |
| 4 | 4317 | 182 | 95,78% |
| 5 | 29273 | 1415 | 95,17% |

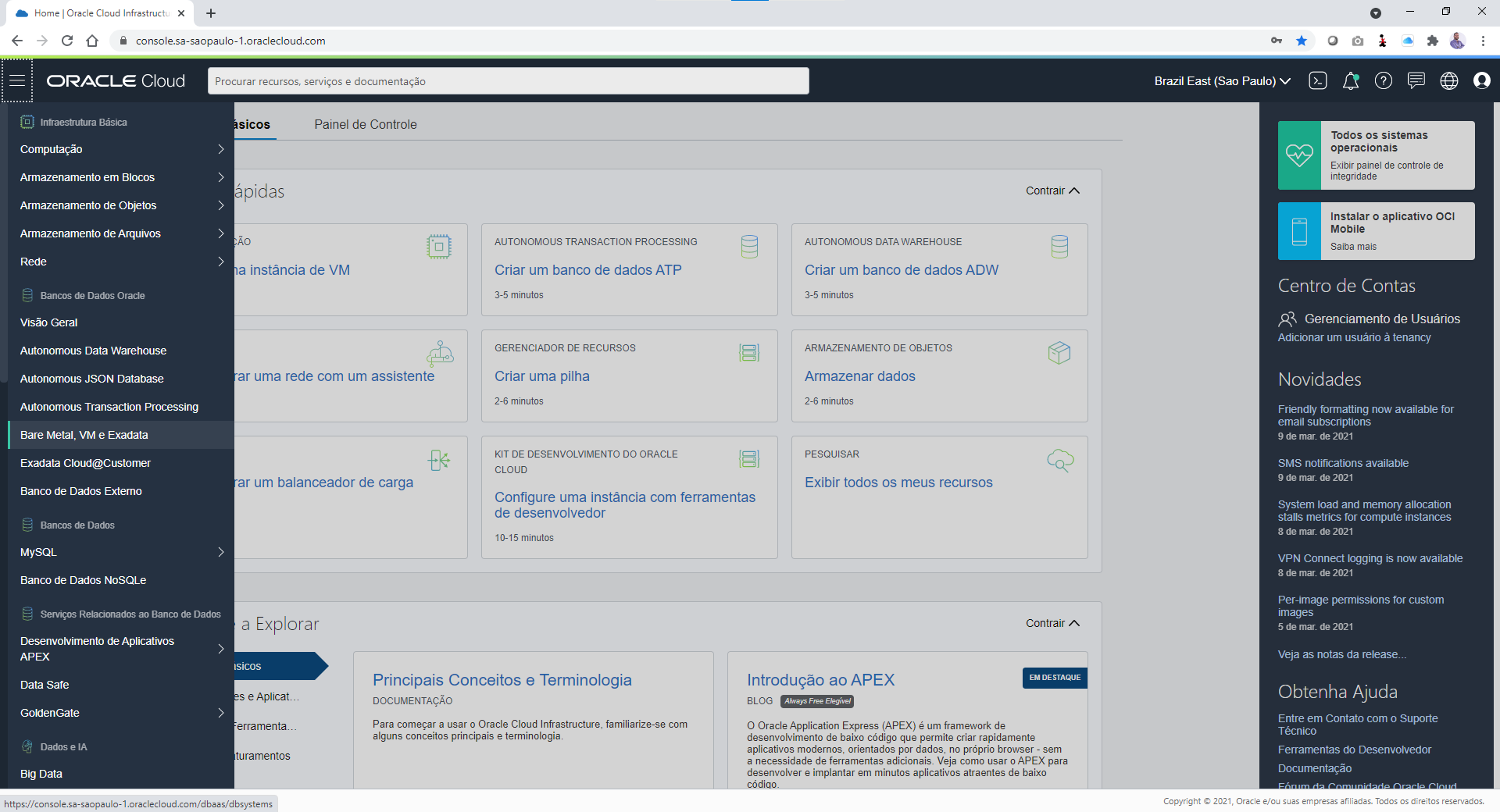
A redução de custos nem sempre será acima de 90%, dependendo do tipo de query o custo pode ser mais próximo, entretanto podemos concluir que o InMemory é uma excelente solução para cargas de trabalho mistas onde um mesmo banco de dados precisa realizar operações transacionais e consultas analíticas para relatórios em tempo real.

# Finalizando a sessão

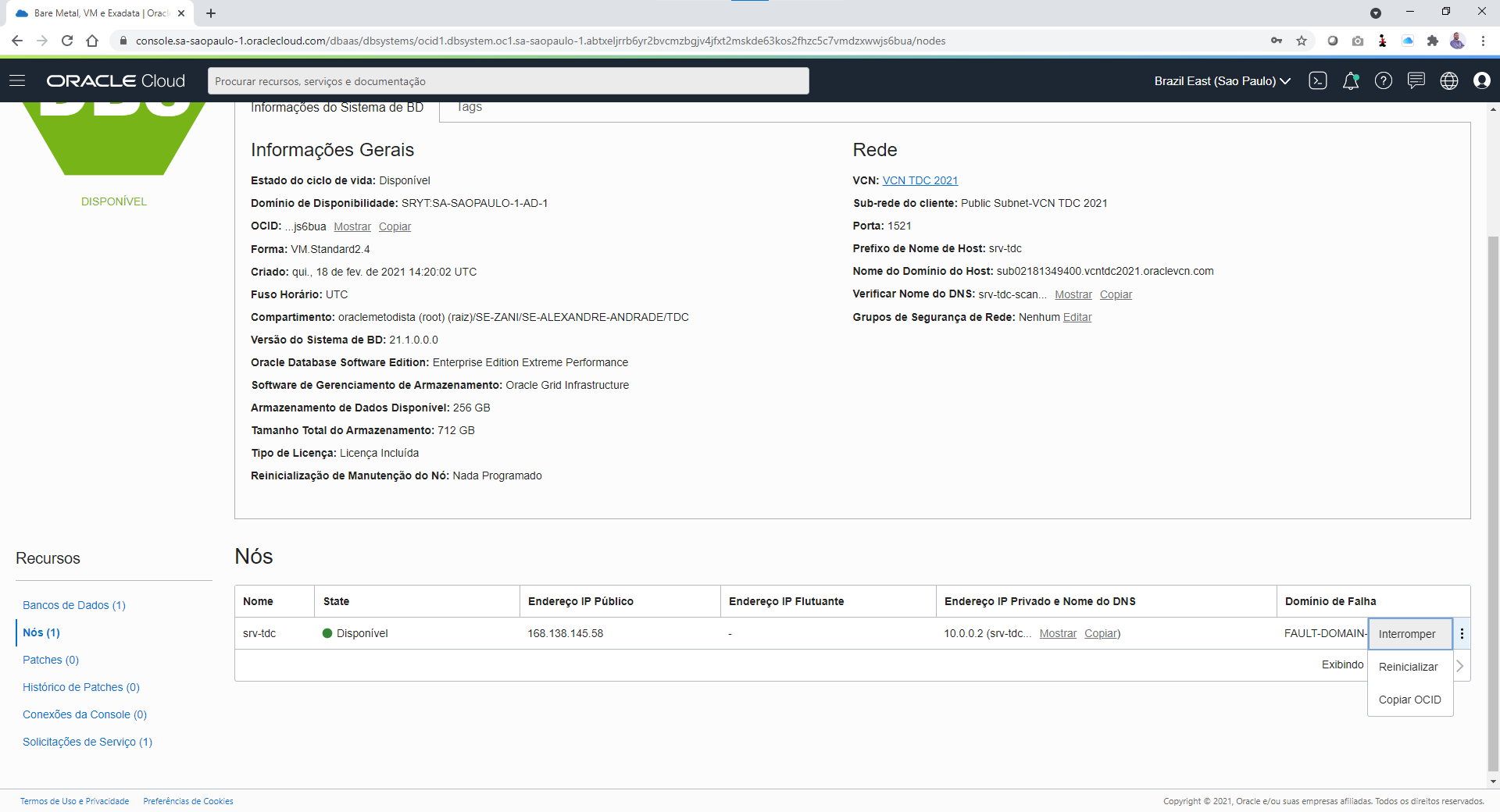
Enquanto o seu banco de dados estiver ativo ele seguirá consumindo recursos de seu *tenant* no *Oracle Cloud*. Portanto você pode desligá-lo, onde será cobrado apenas o uso do armazenamento, ou destruí-lo, cessando por completo a cobrança.

## Desligando banco de dados

Para desligar o banco de dados abra o menu e vá em *Banco de Dados Oracle -> Bare Metal, VM e Exadata*



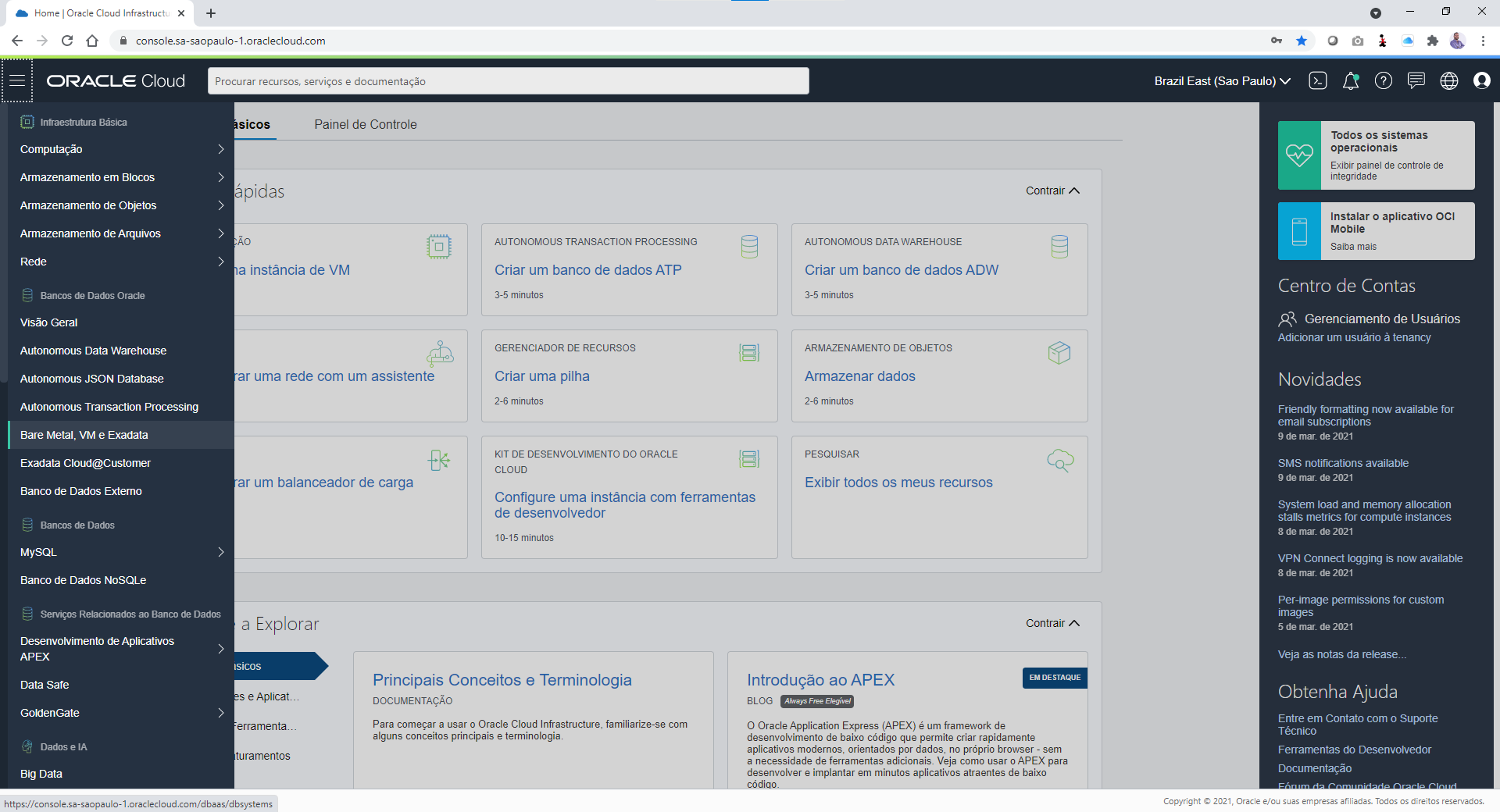
Na lista de Sistemas de BD clique no nome que você escolheu anteriormente. Na lista de recursos que se encontra à esquerda, clique em *Nós*. À direita do nome do servidor clique nos 3 pontos e escolha a opção *Interromper*.



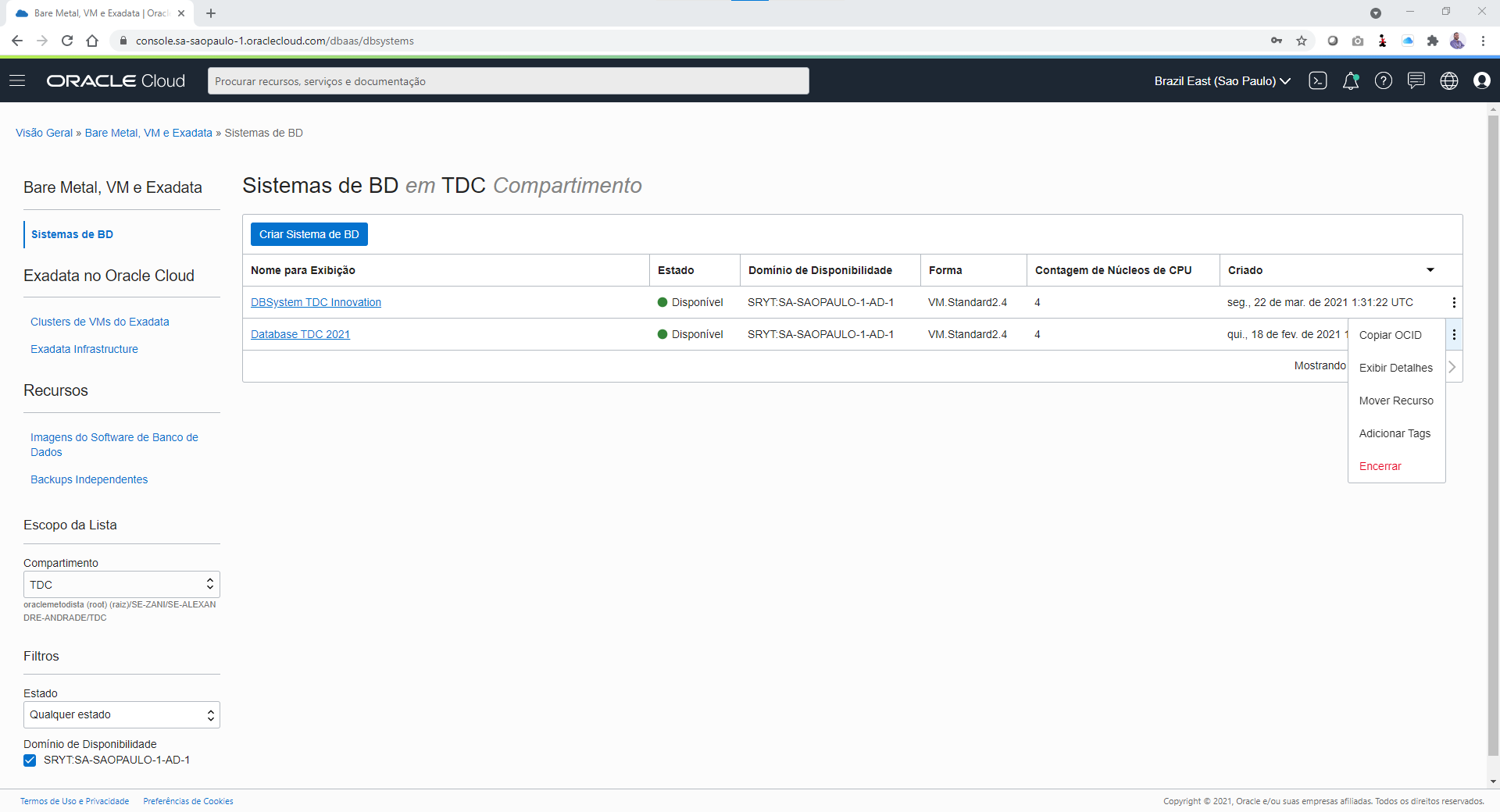
Para usar novamente o seu banco de dados, volte neste mesmo menu e clique em *Iniciar*.

## Destruindo o banco de dados

Para destruir o banco de dados abra o menu e vá em *Banco de Dados Oracle -> Bare Metal, VM e Exatadata*



Na lista de Banco de Dados que surgir clique nos 3 pontos que aparecem à direita do nome de seu sistema. Faça isso com bastante atenção, essa operação não pode ser desfeita e todos os dados serão destruídos.



Confirme a exclusão do banco de dados entrando com o nome do sistema e clique em *Encerrar Sistema de BD*. O banco de dados será encerrado definitivamente e não consumirá mais recursos de seu *tenant*.

