

Como fazer uma jornada para nuvem sem (ou quase) turbulência?

Tales Casagrande

23.03.21 10h00





Este trabalho está licenciado sob uma Licença Creative Commons Atribuição-CompartilhaIgual 4.0 Internacional. Para ver uma cópia desta licença, visite http://creativecommons.org/licenses/by-sa/4.0/.





Índice

[1. Considerações iniciais e pré-requisitos 4](#_Toc66863107)

[Recursos usados 4](#_Toc66863108)

[Tópicos não cobertos 4](#_Toc66863109)

[2. Considerações iniciais e pré-requisitos 4](#_Toc66863110)

[2.1. Oracle Cloud 4](#_Toc66863111)

[2.2. Trend Micro Cloud One 4](#_Toc66863112)

[3. Provisionando os recursos 5](#_Toc66863113)

[3.1. Criação de uma instancia VM 5](#_Toc66863114)

[4. Protegendo o Servidor 8](#_Toc66863115)

[5. Protegendo uma aplicação vulnerável em Python 11](#_Toc66863116)

[5.1. Preparando o ambiente 12](#_Toc66863117)

[5.2 Realizando um ataque 13](#_Toc66863118)

[5.3 Visualizando os eventos 14](#_Toc66863119)

[6. Concluindo 14](#_Toc66863120)

# Considerações iniciais e pré-requisitos

## Recursos usados

OCI (all free tier)

* Compute VM 2.1 Shape

Trend Micro Cloud One

* Cloud One Workload Security
* Cloud One Application Security

Local

* Gerador de chaves SSH – usado: PuttyGen
* SSH Terminal Client – usado: Putty / MobaXterm

## Tópicos não cobertos

Instalação dos softwares na máquina host

* Gerador de chaves SSH – usado: PuttyGen
* SSH Terminal Client – Putty / MobaXterm
* Instalação do Git
* Instalação do Docker

# Considerações iniciais e pré-requisitos

## Oracle Cloud

Para esse workshop será necessário criar uma conta na Oracle Cloud é gratuito será a onde iremos prover toda a infraestrutura para o workshop.

Link <https://cloud.oracle.com>

## 2.2. Trend Micro Cloud One

O Trend Micro Cloud One é uma plataforma para times de infraestrutura, segurança e desenvolvimento que com 6 módulos ajuda os times a moverem seu workloads para nuvem ou proteger aplicações que já nasceram em nuvem ou até mesmo ajudar na excelência operacional da nuvem.

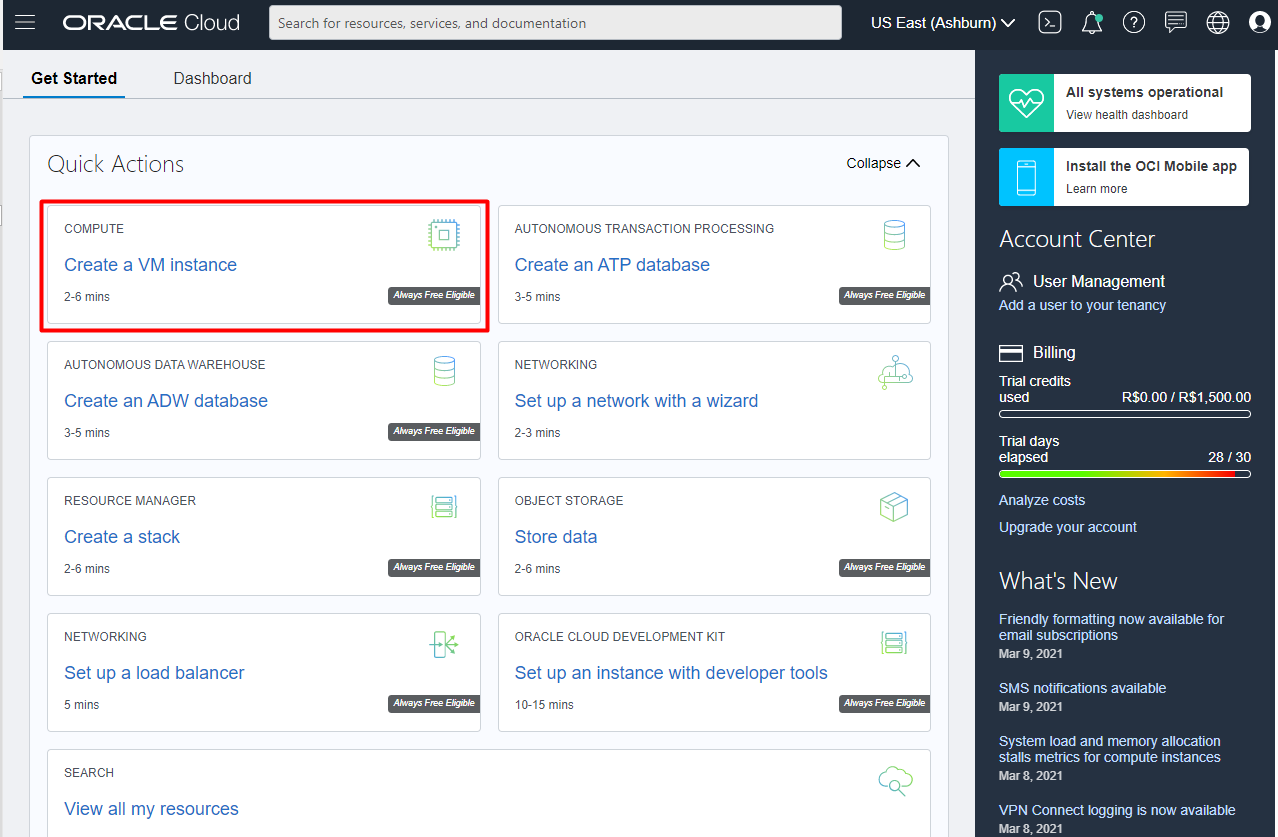
Para esse workshop será necessário criar uma conta no Cloud One, a criação da conta é gratuita e iremos utilizar para realizar a proteção dos workloads e aplicações.

Link <https://cloudone.trendmicro.com/>

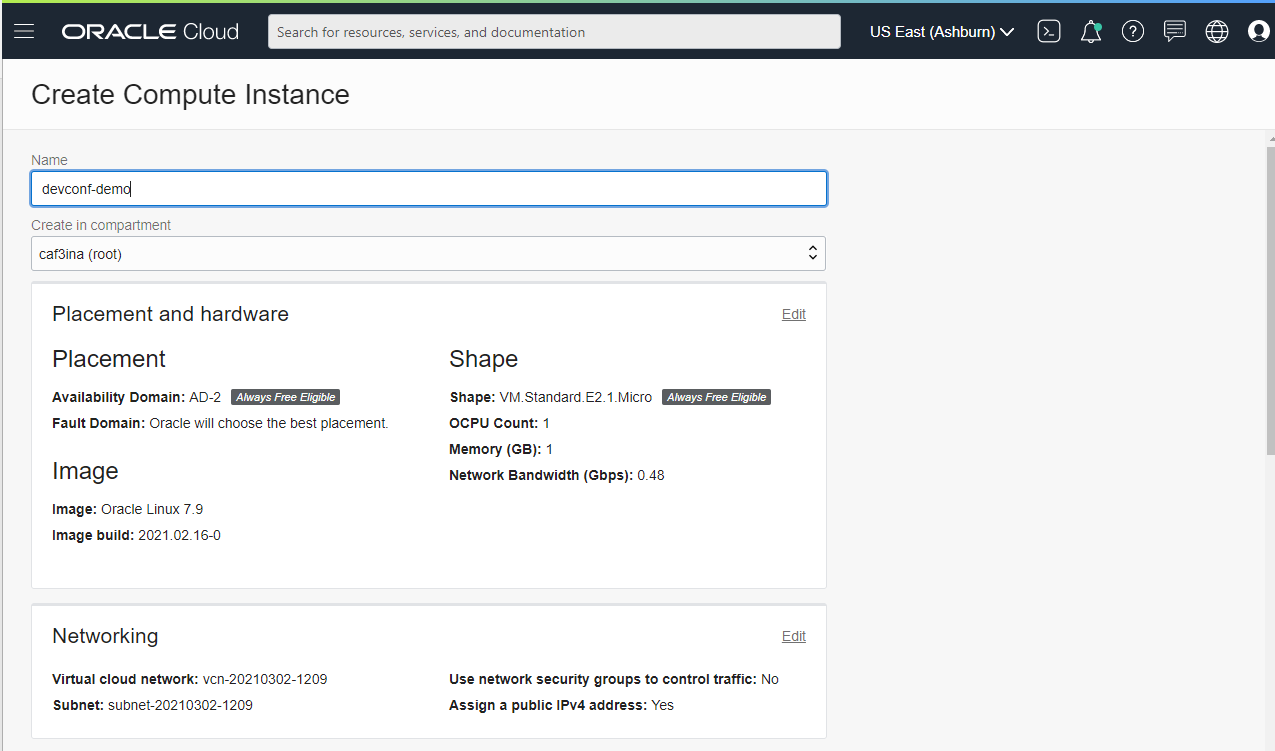
# Provisionando os recursos

## Criação de uma instancia VM

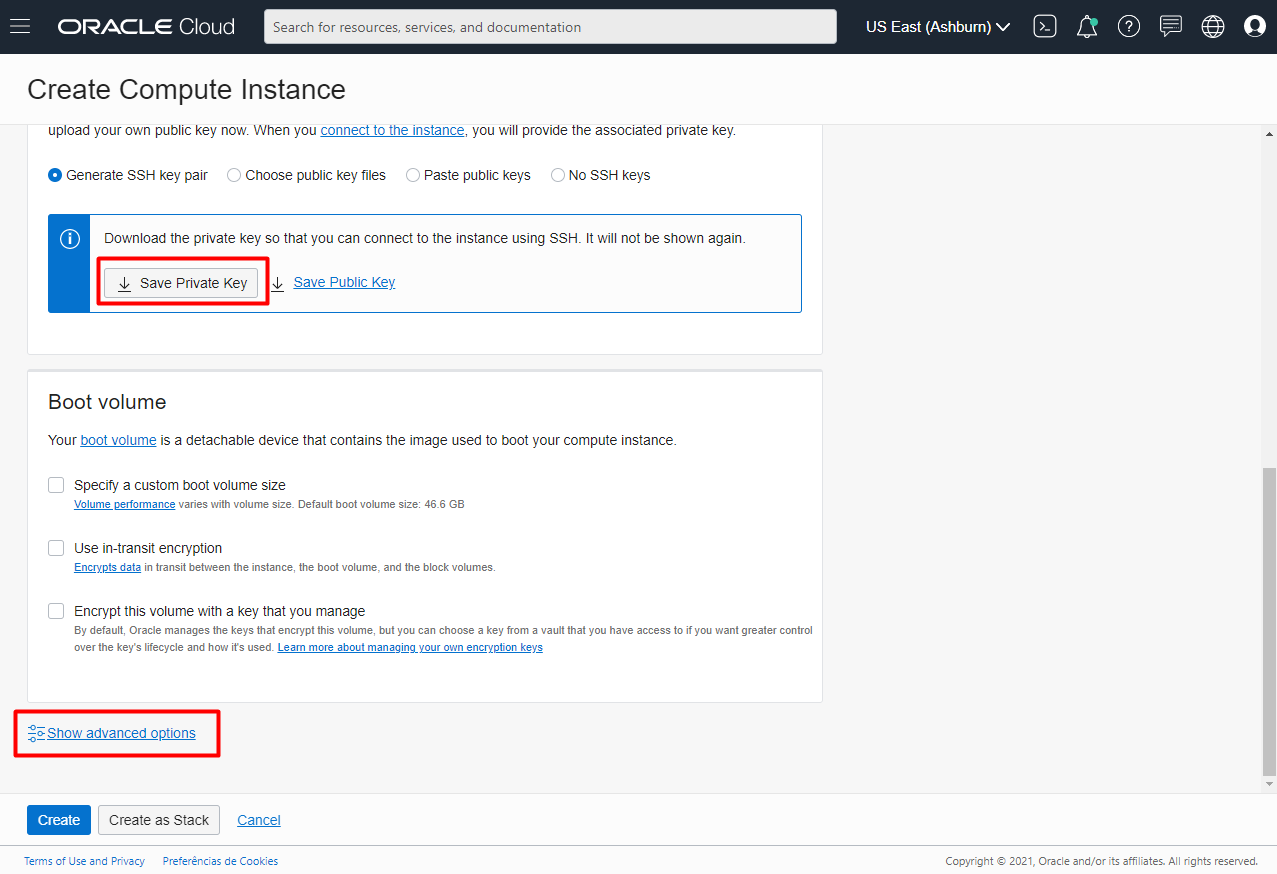
Navegue no menu direito até Compute > Instances também é possível usar a visualização no meu principal como o exemplo abaixo:



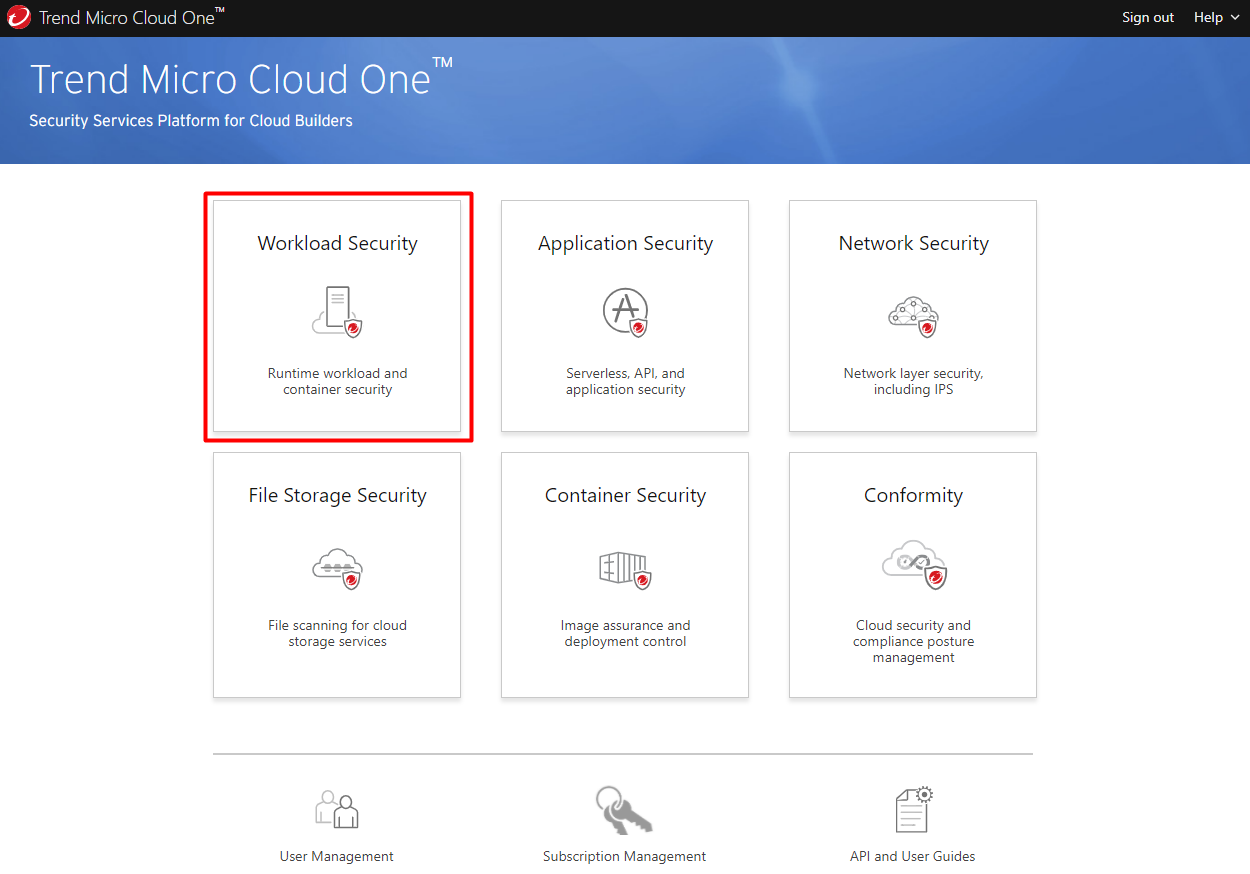
Defina um nome para seu workload, faça o download das chaves privadas para uma conexão futura via ssh.



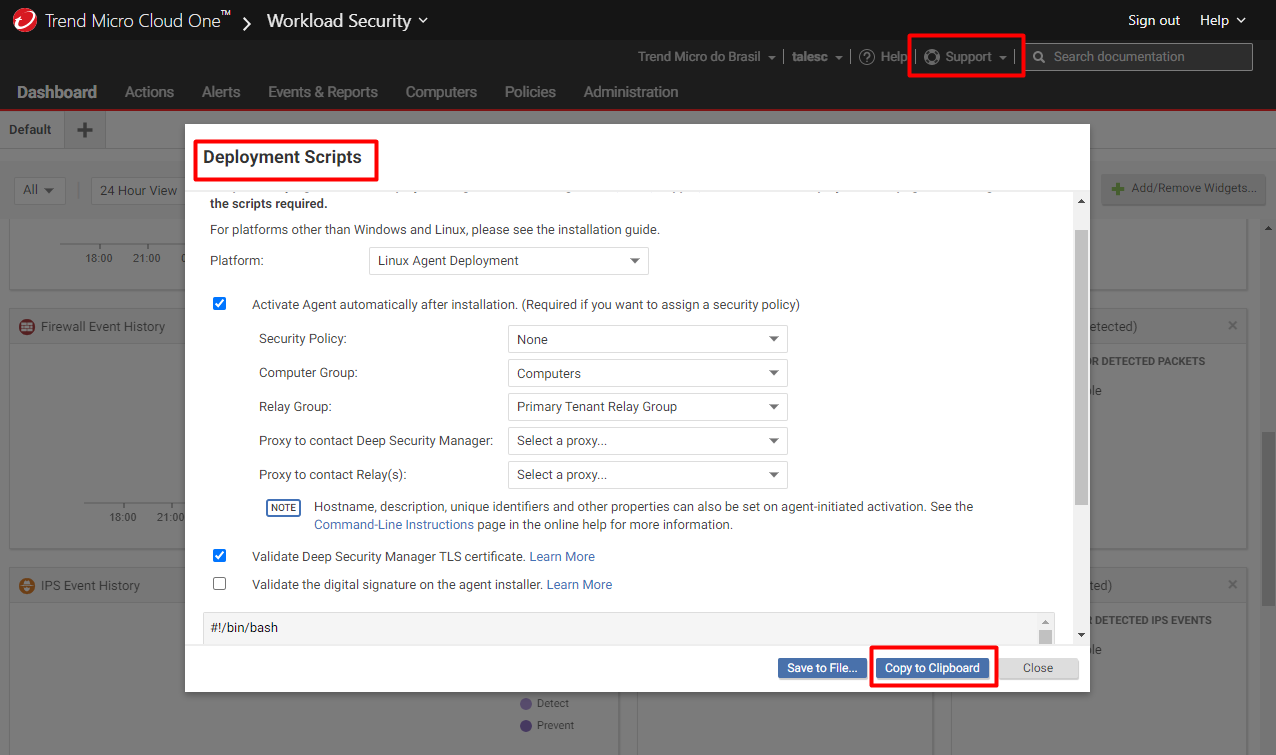
Clique em show advanted options, iremos colocar um script na inicialização do servidor dessa forma realizando a instalação do agente do Cloud One Workload Security.



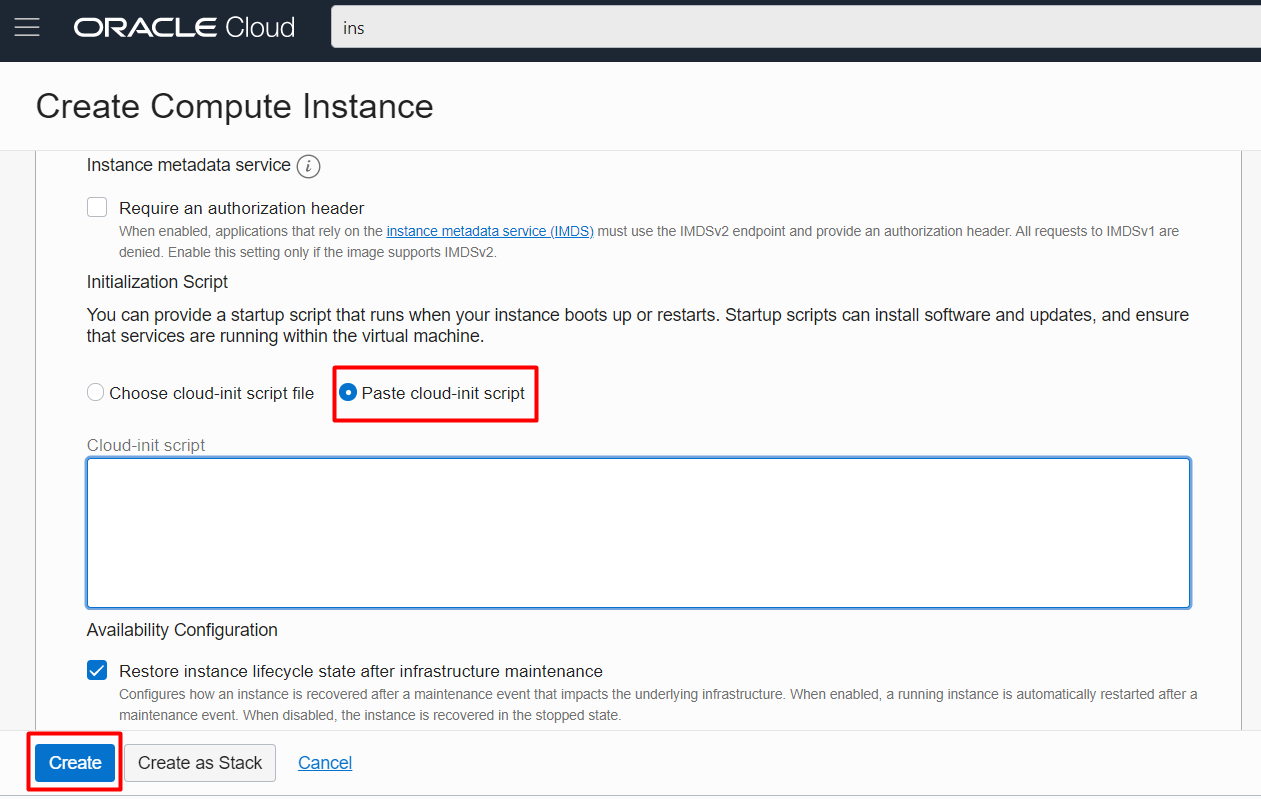
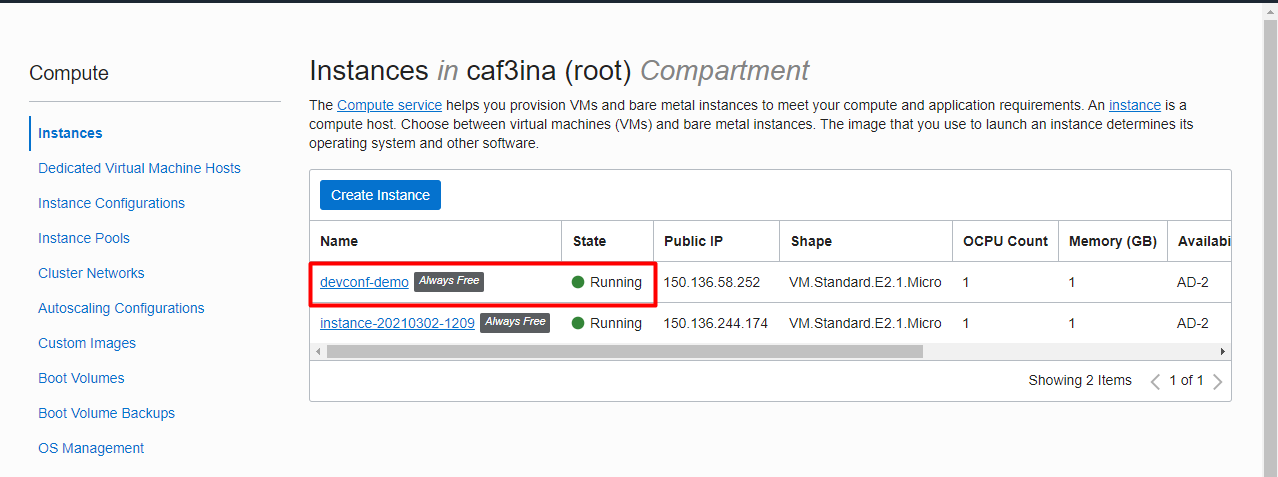
Acessar a plataforma Cloud One (a conta deve ser criada previamente) e escolher a opção Workload Security.



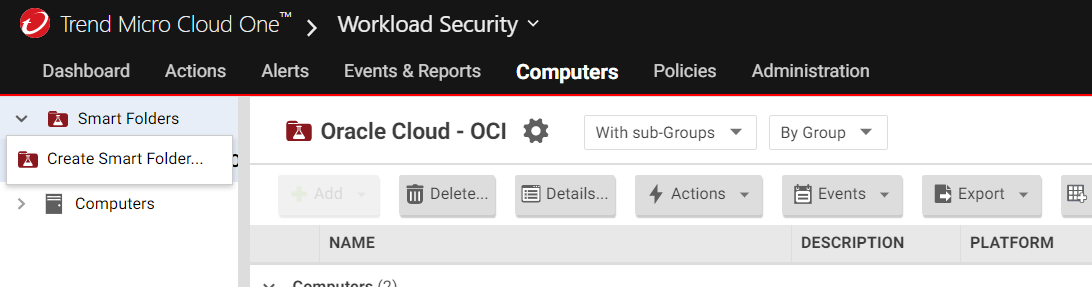
No meu Support selecione a opção Deployment Scripts, selecione a Plataforma Linux Agent Deployment e clique em Copy to Clipboard, iremos copiar o script de instalação e colocar na inicialização do Servidor.



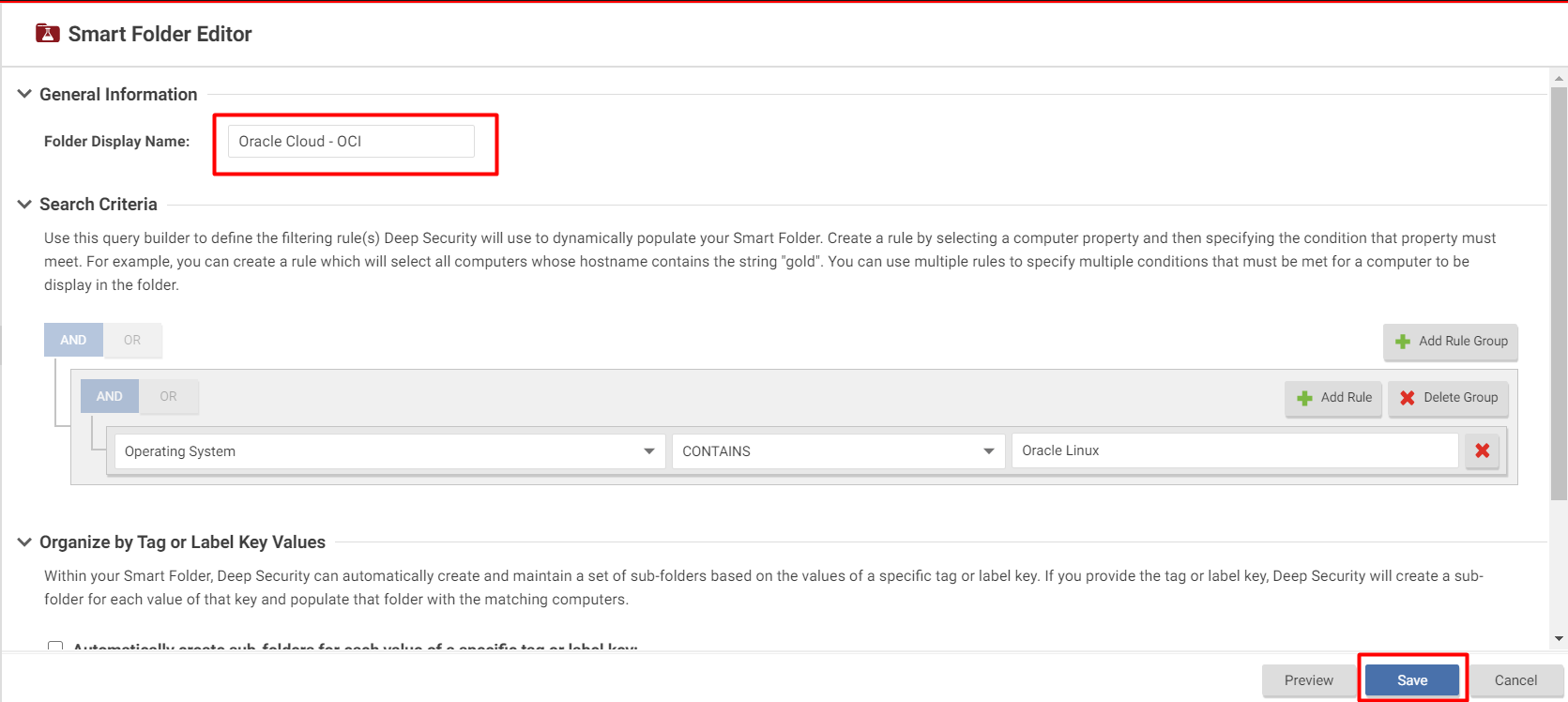
Novamente na console do OCI, selecione a opção Paste cloud-init script e cole o script gerado a partir da console do Cloud One. Após isso selecione create e aguarde alguns minutos até a inicialização do Servidor.

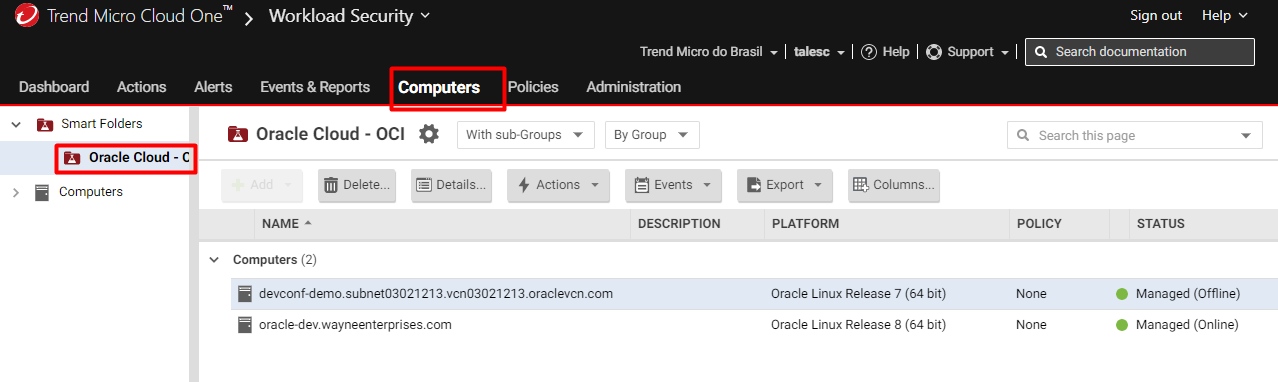
Enquanto o Servidor é inicilizado iremos criar uma Smart Folder para Organizarmos nosso ambiente. Acesse novamente a console do Cloud One e navegue no meu conforme abaixo no menu Computers > Smart Folde > Create Smart Folder.



Define um nome para a Smart Folder no Search Criteria utilize o exemplo da imagem abaixo



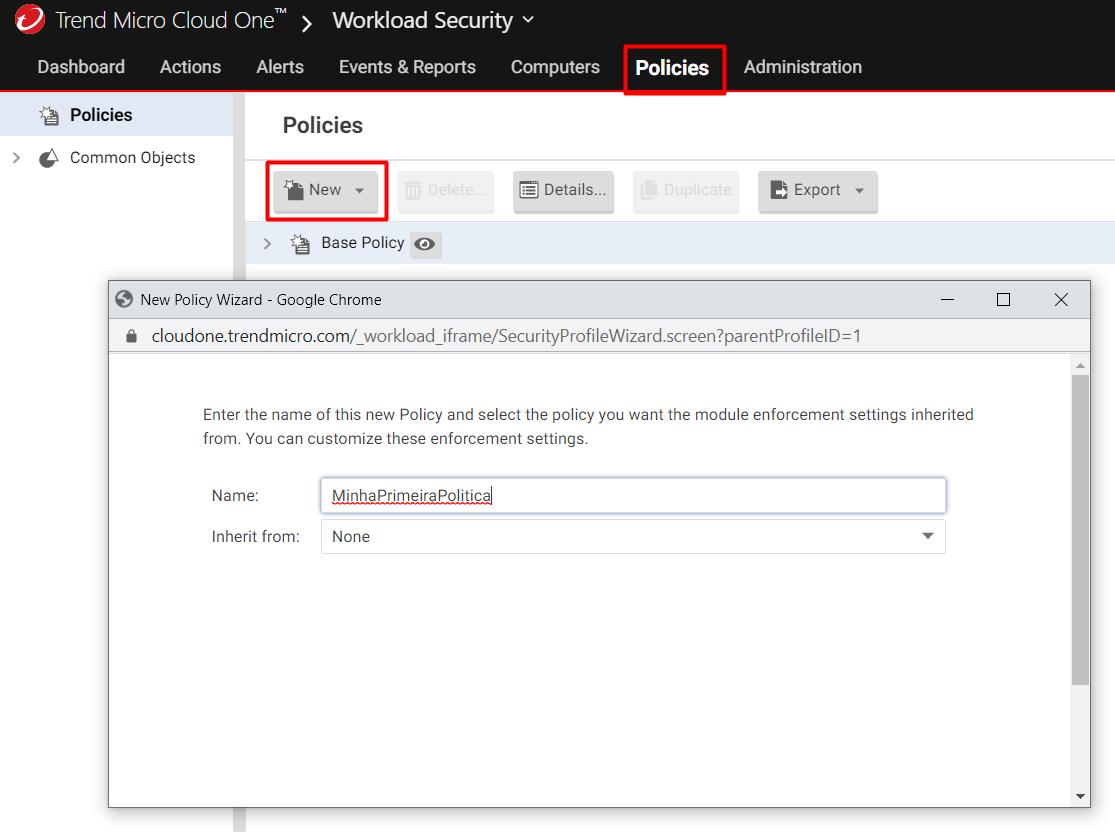
Com o Servidor já inicializado é possível visualizarmos no Cloud One, nesse exemplo tenho dois Servidores um on-premise e outro criado anteriormente.



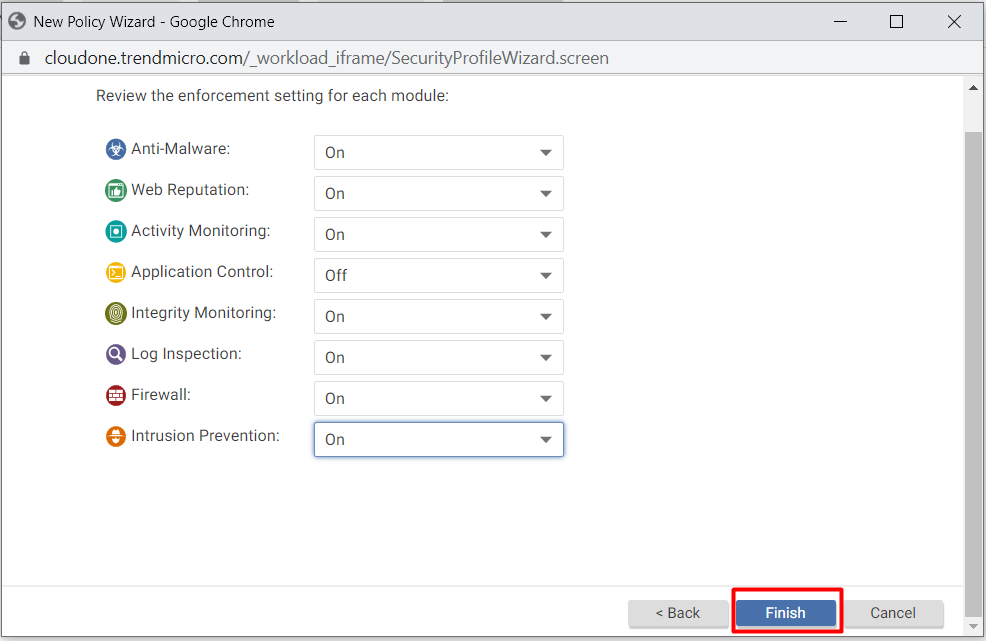
# Protegendo o Servidor

É possível automatizar o processo para que sempre que um novo servidor for adicionado na console do Cloud One ele já inicie com proteção, basta selecionarmos a política no Deployment Scritps.

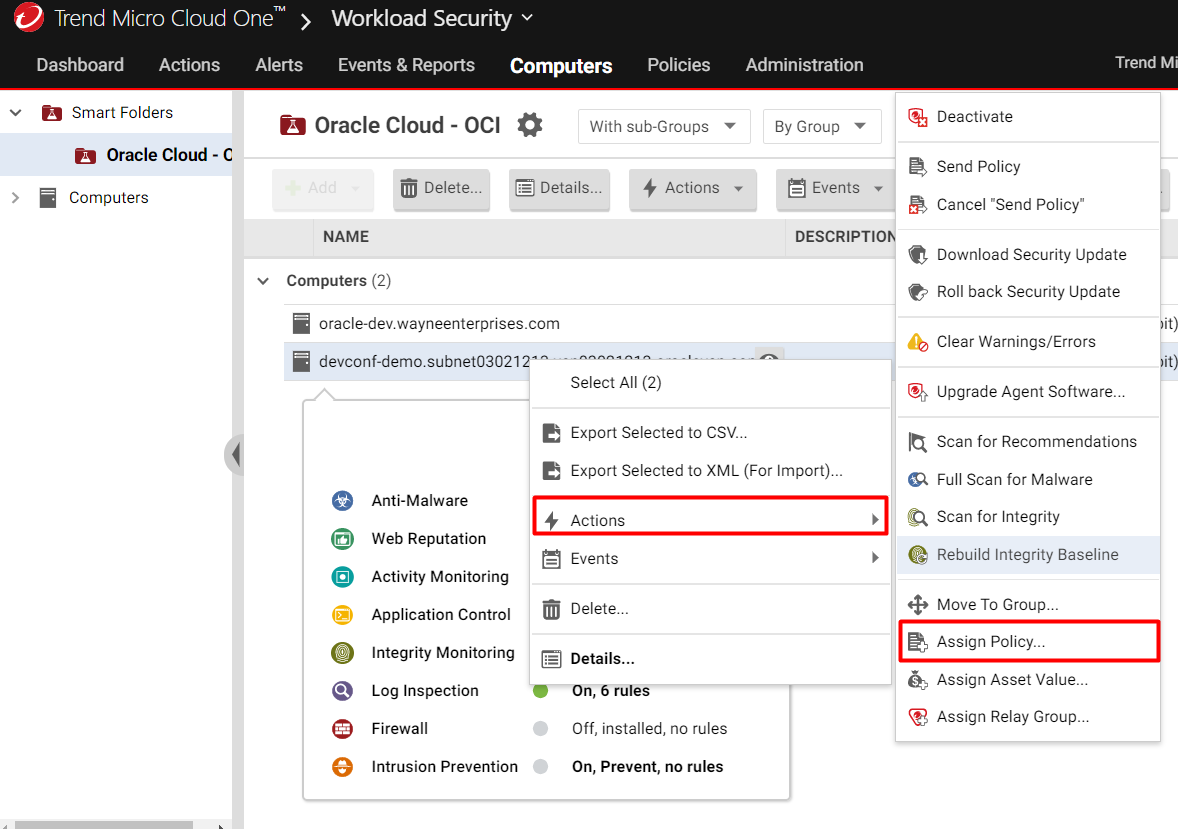
No meu Policies, basta clicar em new, definir um nome e clicar em next.



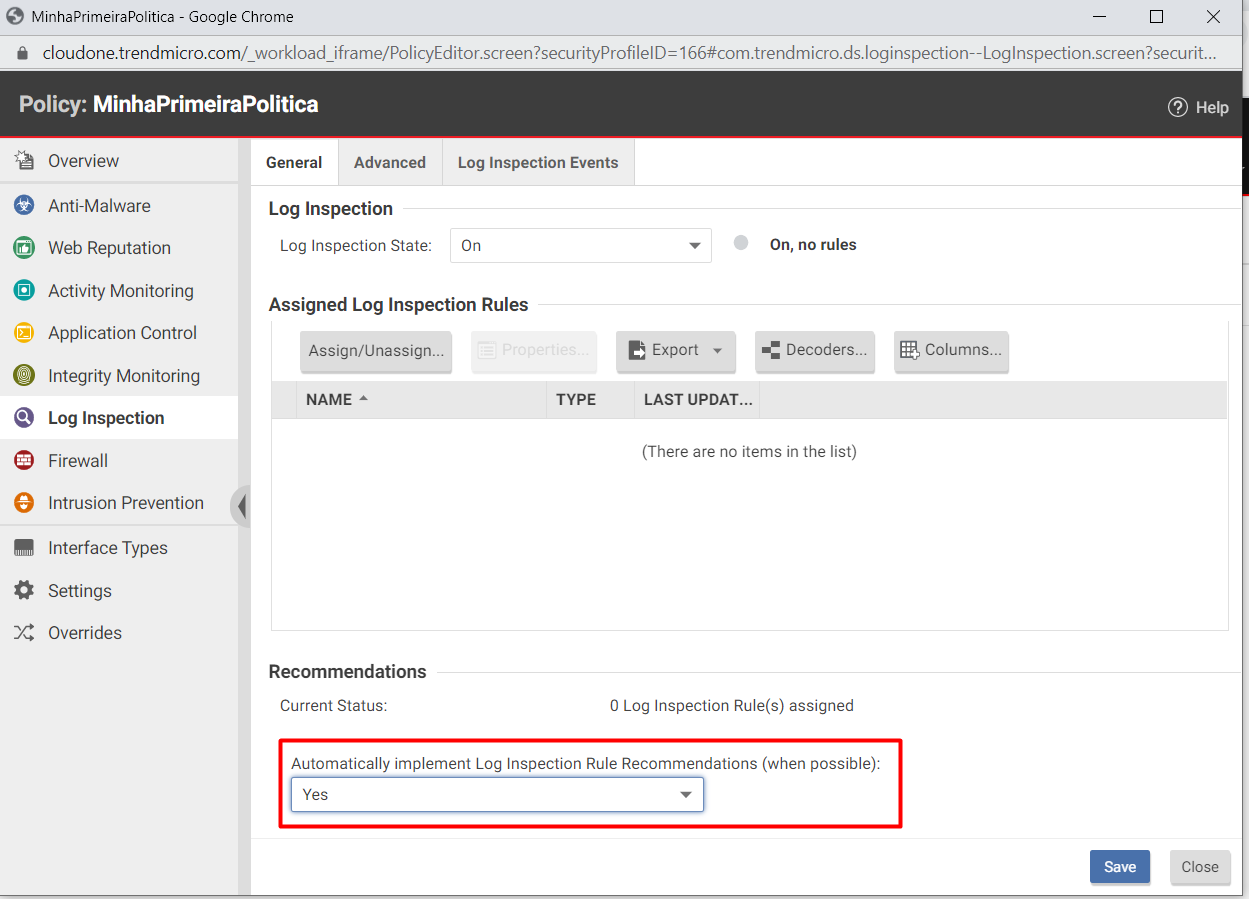
Na opção “Do you want to base this policy on an existing Computer's current configuration? Pode escolher a opção No, não iremos utilizar nenhuma herença.



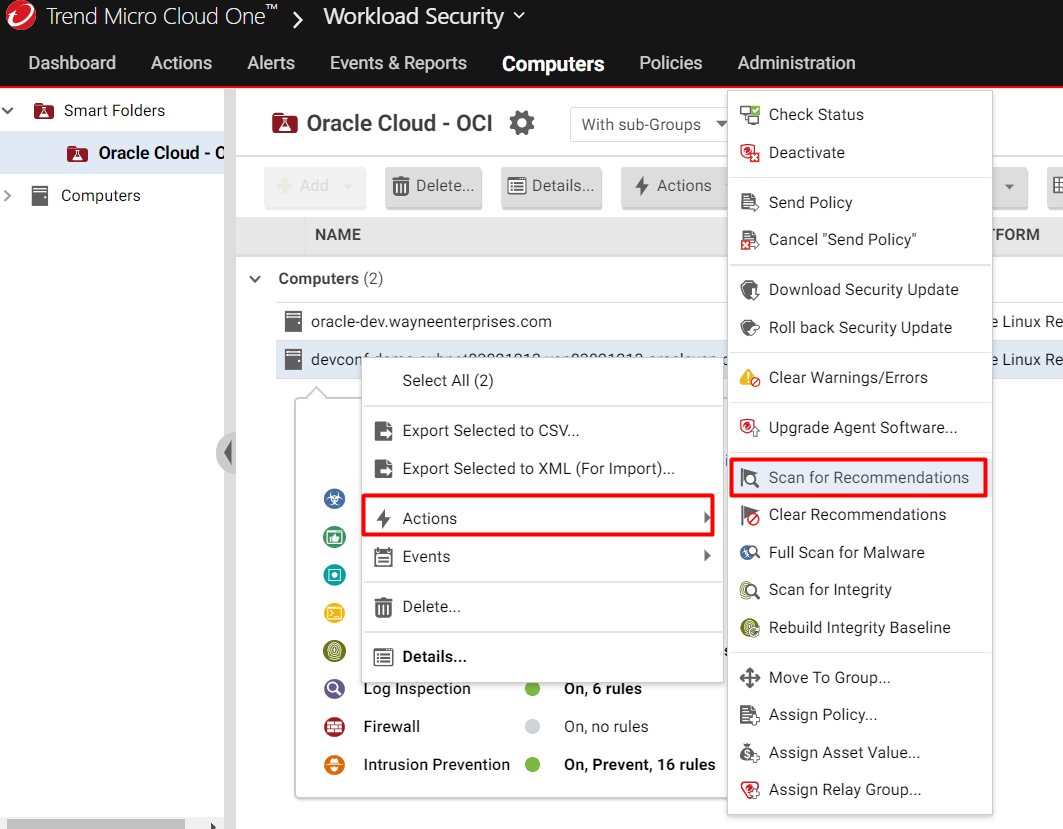
Cada módulo do Workload Security oferece uma camada de proteção, que agora serão habilitados na política.



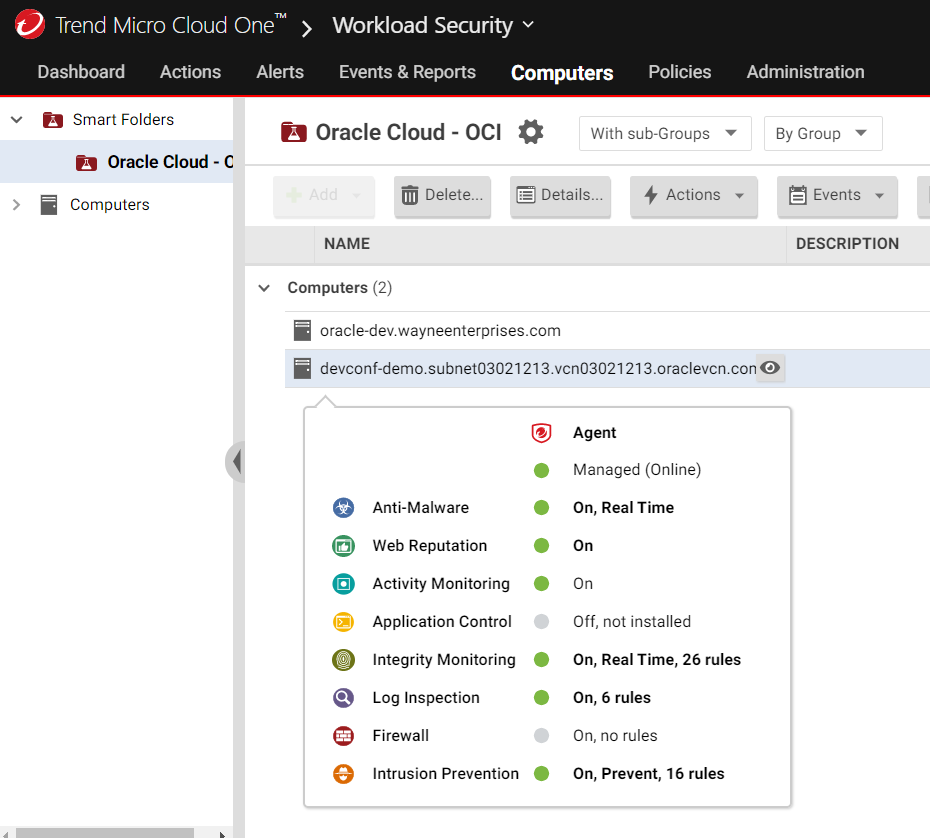
Em Assign Policy selecione a política que acabamos de criar. Para os módulos Log Inspection, Integrity Monitoring e Intrusion Prevention mude o status do Recommendations para yes dessa forma a ferramenta irá identificar falhas de segurança e aplicações para iniciar o monitoramento.



Após mudar para Yes, iremos executar o Scan for Recommendations para o Workload Security aplique as regras.

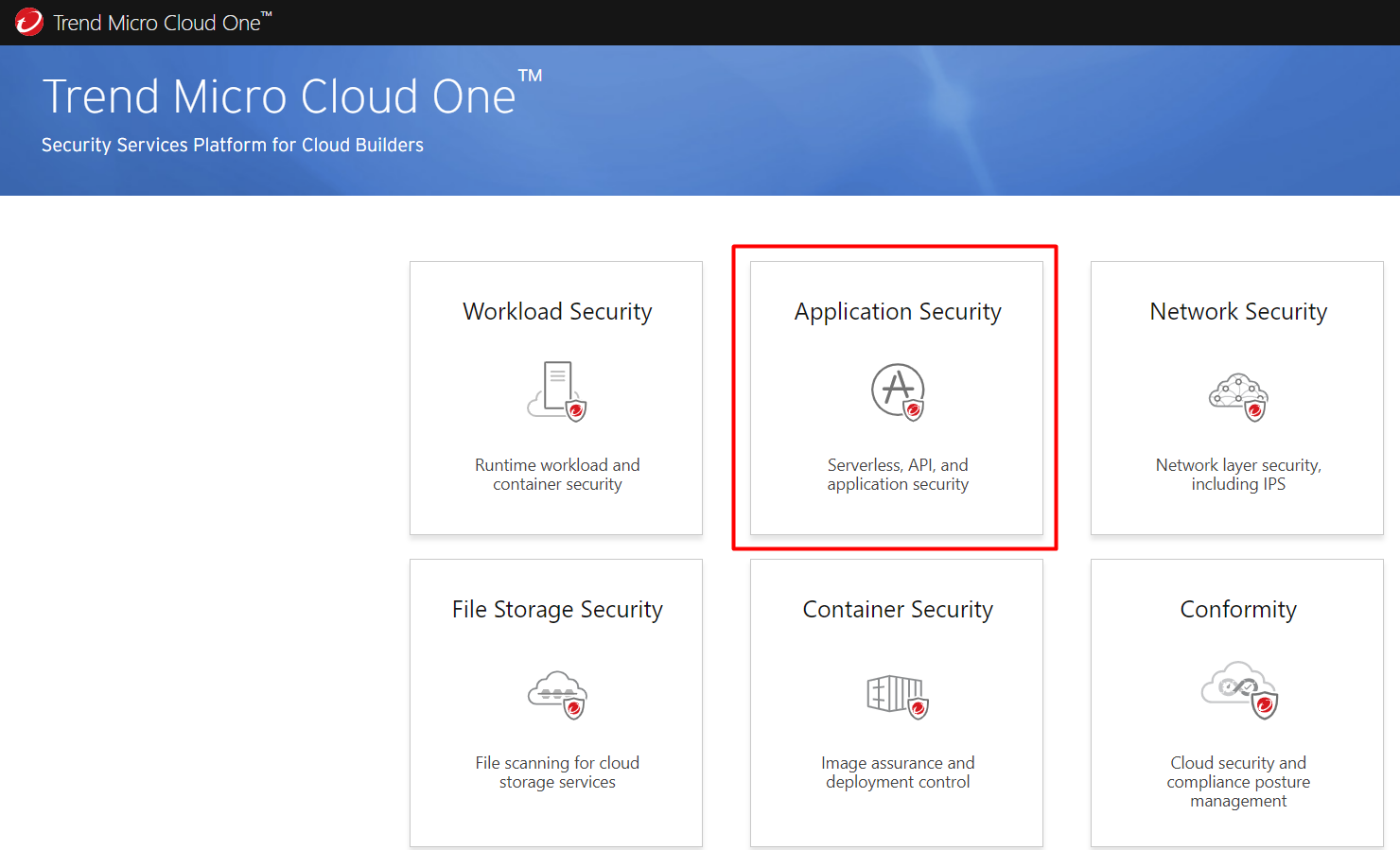


Após a execução podemos ver que já existem regras aplicadas no Servidor.



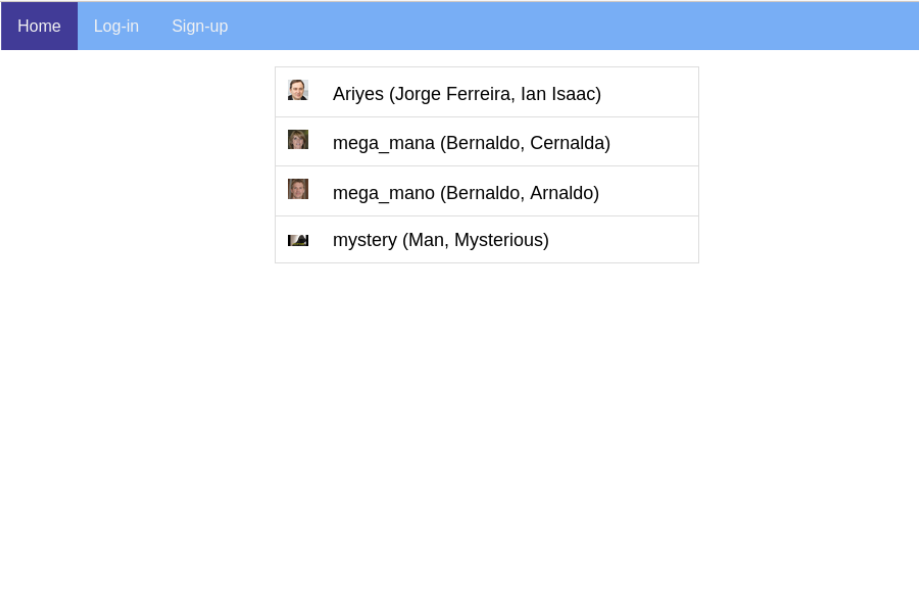
# Protegendo uma aplicação vulnerável em Python

Não podemos instalar um agente quando falamos de código, o Application Security ajuda nesses desafios protegendo aplicações com apenas duas linhas de código.



## Preparando o ambiente

Iremos utilizar um aplicativo da web django simples e vulnerável para testar e aprender como se proteger usando o Trend Micro Application Security.



Acesse o Servidor que criamos a cima via ssh e eleve o privilegio caso necessário.

$ yum install git -y

$ git clone <https://github.com/caf3ina/HeadPage.git>

$ cd HeadPage/

Edite a seguinte linha em src /headpage /settings.py para servidor do HeadPage ouvir em todas as interfaces. Isso pode ser perigoso, se possível, executado dentro de uma VM na interface somente de host.

ALLOWED\_HOSTS = ['\*']

Crie um arquivo com o nome trend\_app\_protect.ini e coloque as informações abaixo. Informações abaixo:

[trend\_app\_protect]

key = my-key

secret = my-secret

Obtenha a chave e o segredo do console do Cloud One Application Security <https://cloudone.trendmicro.com/docs/application-security/python/#install-the-agent>

$ docker build --tag=headpage:latest .

$ docker run -d --rm -p 8000:8000 --name headpage headpage:latest

Acesse a página <http://meu-ip:8000/social/>

## Realizando um ataque

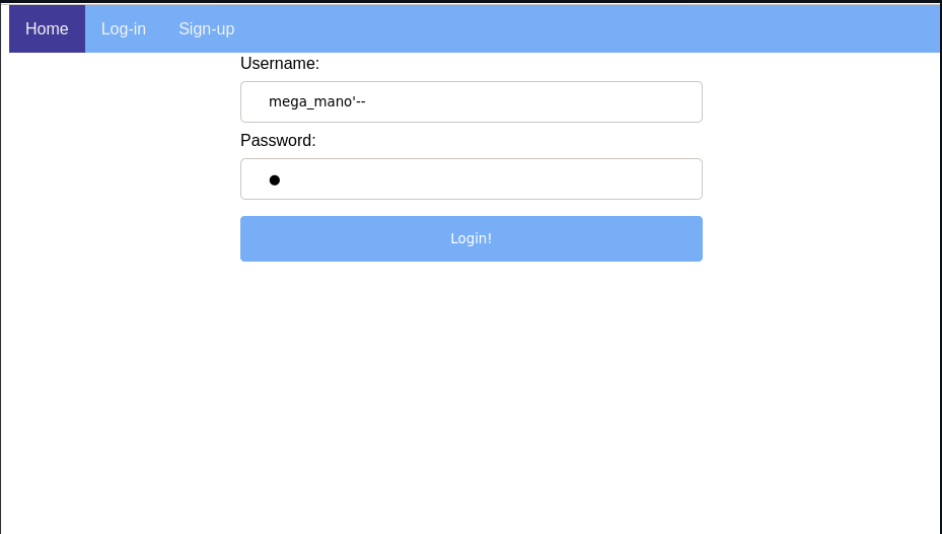
Iremos explorar uma falha de **SQL Injection** que é uma falha bem comum em aplicações Web.

Na página Home da HeadPage, utilize as credenciais abaixo e depois clique em login:

Username: mega\_mano'—

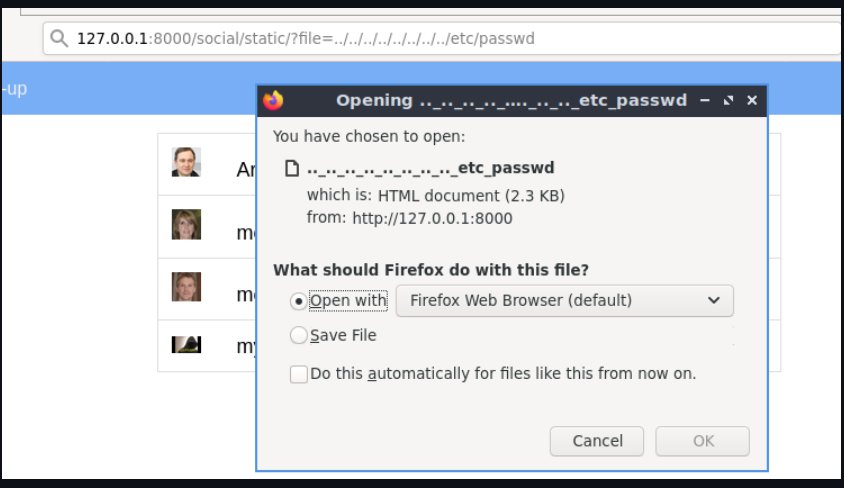
Senha: qualquer\_uma

$ docker run -d --rm -p 8000:8000 --name headpage headpage:latest



Um segundo ataque que podemos realizar é o **Path Traversal,** alguns arquivos estáticos são retornados após solicitações GET.

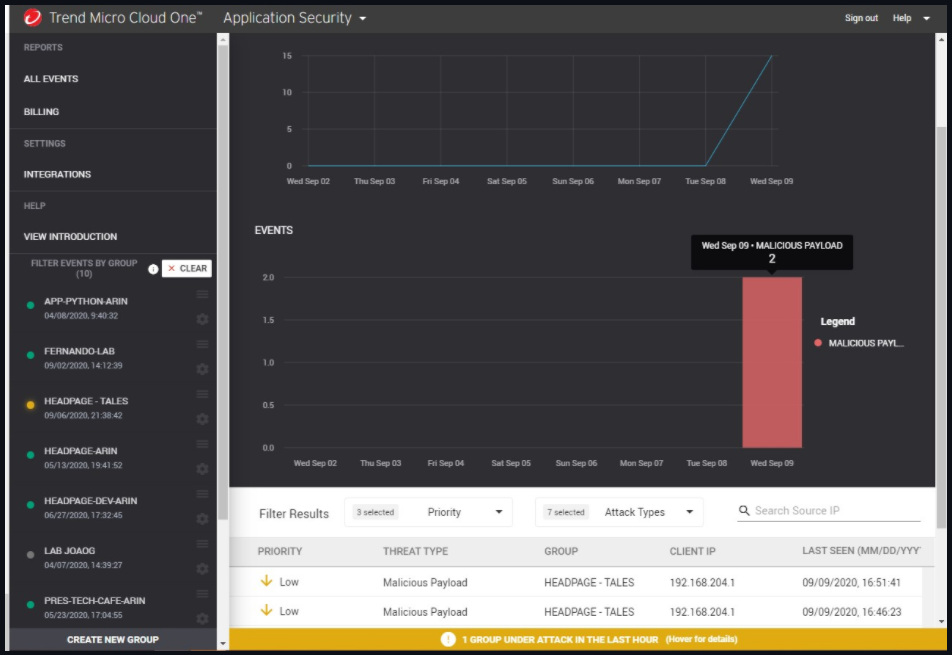
Acesse a url: http://meuip: 8000/social/static/?file=privacy.txt



## Visualizando os eventos

Na console do Cloud One dentro do Application Security podemos ver os eventos conforme abaixo, é possível criar políticas para apenas gerar log ou bloquear.

Os eventos podem ser enviados para o Slack, Pagerduty, New Relic Insights ou até mesmo um Tópico SNS.



# Concluindo

É importante adicionarmos camadas de proteção em todas as fases da criação de uma aplicação, dessa forma teremos visibilidade do que acontece no host, no código e teremos uma aplicação estável e segura.

