

[SW75] Boas práticas na Sandbox

- [Introdução](#)
- [Problemas Comuns](#)
 - [Estouro de Memória](#)
 - [Erro SSH/SFTP](#)
- [Boas Práticas e Prevenção](#)
 - [Uso Consciente dos Front-Ends - 3 por usuário](#)
 - [Desativar FEs ao finalizar seu teste](#)
 - [Monitoração, Monitoração, Monitoração](#)
- [Resolução](#)
 - [Estouro de Memória](#)
 - [Transferência SFTP](#)
- [Conclusão](#)

Introdução

O SW75 é a solução monolítica para Autorização da Rede e foi desenvolvida para rodar numa única máquina AIX nos ambientes de Homologação e Produção. Para desenvolvimento de novos projetos temos poucas máquinas para desenvolvimento disponíveis. Isso resultava em concorrência pelas máquinas ou ainda em conflitos em que uma equipe alterava o mesmo fonte que outra equipe já estava trabalhando, ocasionando erros, retrabalho e bastante dor de cabeça.

Para trazer agilidade no desenvolvimento e nos testes, foi criada a solução da Sandbox SW75. Essa máquina compartilhada (soxx105cto/10.59.178.20) realiza o isolamento de fontes e binários, logs e portas de testes entre diversos usuários que podem, por fim, trabalhar de forma paralela sem atrapalhar outra equipe. A Sandbox é utilizada por diversas equipes e usuários para desenvolvimento de novos projetos, execução de testes e correções de sustentação.

No entanto, a solução do SW75 foi desenvolvida com apenas uma instância do Switch rodando por máquina. Como os recursos computacionais da máquina soxx105cto são finitos, alguns problemas começaram a ocorrer.

Problemas Comuns

Com grandes poderes vem grandes responsabilidades. Nessa sessão, vamos ver alguns problemas comuns que ocorrem devido ao mal uso da Sandbox.

Estouro de Memória

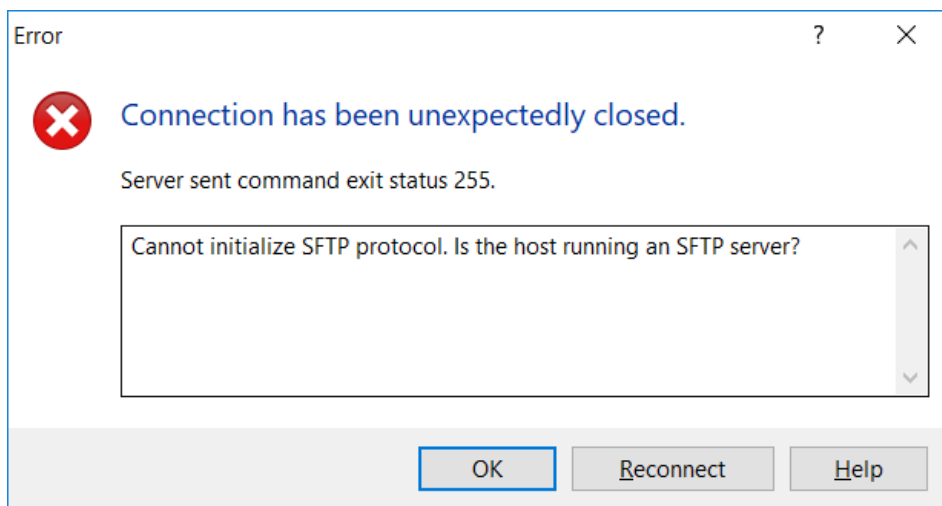
O SW75 consome bastante memória da máquina. Muitos dos fontes XML são carregados em memória e quanto mais front-ends estiverem ativos, maior a quantidade de memória é consumida. Se tivermos diversos usuários simultâneos rodando diversos front-ends ao mesmo tempo, o estrago está feito!

Se a memória de uma máquina fica totalmente ocupada, não é possível abrir novos processos ou sessões. Dessa forma, o servidor não aceita novas conexões nem do usuário root e não é possível desligar processos que estão abertos. Nada pode ser feito para resolver esse problema além de solicitar o reboot da máquina.

Erro SSH/SFTP

Utilizamos diariamente o SSH para logar na máquina de desenvolvimento e o SFTP para compartilhar fontes da nossa máquina pessoal para o ambiente de desenvolvimento, ou ainda para rodar a pipeline de CI/CD. Sendo assim, é um grande problema quando o SSH ou o SFTP estão fora, ocasionando bastante atraso para a equipe de desenvolvimento.

```
317017@8000000081404-NB MINGW64 ~  
$ sftp -P 4098 swoper12@10.59.178.20  
Connection closed
```



Boas Práticas e Prevenção

A fim de evitar o Estouro de Memória, nosso principal vilão, foram levantadas algumas atividades para monitorarmos e prevenirmos novos estouros.

Uso Consciente dos Front-Ends - 3 por usuário

Na maioria dos casos, não é necessário subir o SW inteiro para testar uma nova funcionalidade. Aqui, gostaríamos de **propor a boa prática de subir no máximo 3 Front-Ends por usuário**. Não é complicado, segue exemplo!

Digamos que a sua alteração seja somente no FE-POS. É necessário subir também o FE-PDV? Provavelmente não. Dê preferência aos serviços que você precisa testar.

Do ✓	Don't ✗
swUp POS (ativar somente o FE que vamos utilizar)	swUp (vai subir todos os FEs. É realmente necessário?)

Caso você tenha alguma alteração mais sistêmica, por exemplo uma alteração que afeta todas as transações de crédito. Digamos que é necessário testar transações POS e PDV nas bandeiras Master, Visa, Elo e Amex. São 5 Front-Ends, é bastante coisa! A boa prática é fracionar seu teste:

- Ativar POS, PDV e CRTM.
- Testar a transação Crédito Master no POS e no PDV.
- Desativar CRTM
- Ativar CRT
- Testar a transação Crédito Visa no POS e no PDV.
- Etc...

Desativar FEs ao finalizar seu teste

Implementamos na Sandbox uma rotina que desativa o SW de todos os usuários as 23:00. Isso foi necessário pois muitos usuários esqueciam de desativar sua Sandbox e as vezes ela ficava ativa por dias. No entanto, é durante o dia que a maioria dos estouros de memória ocorre e não no período da noite. Sendo assim, é importante **desativar o seu ambiente após a conclusão do seu teste, para diminuir a chance de um estouro de memória ocorrer**.

Digamos que finalizei o meu teste no FE-POS, coletei minhas evidências e afins. Hora de encerrar o minha Sandbox com "swKill":

```

#/home/swoper12# swKill
#####   ###   #####   #   #####   #   ###   #   #   #####
#   #   #   #   #   #   #   #   #   #   #   #   #   #
#   #   #   #   #   #   #   #   #   #   #   #   #   #
#   #   #   #####   #   #   #####   #   #   #   #   #   #
#   #   #   #   #####   #   #   #   #   #   #   #   #
#   #   #   #   #   #   #   #   #   #   #   #   #   #
#####   ###   #####   #   #   #####   #####   ###   #   #   #####

Tue Jun 22 11:29:41 -03 2021
##### Desativando FE CRT ....
##### FE CRT já estava desativado.
##### Desativando FE CRTM ....
##### FE CRTM já estava desativado.
##### Desativando FE CRTO ....
##### FE CRTO já estava desativado.
##### Desativando FE DBT ....
##### FE DBT já estava desativado.
##### Desativando FE DBTM ....
##### FE DBTM já estava desativado.
##### Desativando FE DBTO ....
##### FE DBTO já estava desativado.
##### Desativando FE HST ....
##### FE HST já estava desativado.
##### Desativando FE WEB ....
##### FE WEB já estava desativado.
##### Desativando FE POS ....
##### FE POS DESATIVADO!

```

É bem fácil! 😊

Monitoração, Monitoração, Monitoração

Se todos fizerem a sua parte nas boas práticas, vamos diminuir a ocorrência de estouros de memória. No entanto, é bastante importante **monitorarmos a quantidade de memória que está ativa antes de iniciarmos novos FEs**.

O [James Henry David Cooley](#) desenvolveu a chamada "freemem" que verifica a quantidade de memória livre na Sandbox. Essa verificação é importante para termos a noção de quanto "perigo" estamos correndo ao ativar Front-Ends:

```

#/home/swoper12# freemem
Memória Total:
8388608
Memória Livre:
2184577
-----
Porcentagem Livre:
26.0422%
#/home/swoper12#

```

Alguns comandos interessantes disponibilizados pelo [Fabio Devai](#):

```

# obtem estatísticas do servidor a cada 2 segundos
vmstat -t 2

# Processos Consumo + CPU
ps aux | sort -r -n +2 | more

# Processos Consumo + Memória
ps aux | sort -r -n +3 | more

```

Resolução

Abaixo a forma de resolução dos nossos problemas.

Estouro de Memória

Prevenir é o melhor remédio mas e se o estouro de memória voltar a acontecer, o que devemos fazer? Na página [Falha no acesso SSH](#) temos uma descrição do chamado que deve ser aberto e das pessoas que são responsáveis por executar o reboot do servidor.

Transferência SFTP

Ainda não descobrimos o motivo da falha do serviço SFTP. É provável que o arquivo de configuração abaixo tenha sido alterado por algum serviço externo.

```
soxxl05cto:[/etc/ssh]$ ll /etc/ssh/sshd_config
-r--r--r-- 1 root system 2511 Feb 18 10:51 /etc/ssh/sshd_config
```

Conseguimos realizar a correção com ajuda do [Fabio Devai](#), em que o Tivoli copiou o arquivo utilizado em homologação.

Conclusão

A Sandbox SW75 foi uma revolução para nossas equipes de desenvolvimento e trouxe bastante ownership aos desenvolvedores, que podem implementar e testar novas features, ativar ou desativar o SW conforme desejarem. No entanto é necessário ser também responsável pela saúde do nosso ambiente, pois ele é compartilhado. É um dever de todos utilizá-lo da melhor maneira para que todos continuem produzindo bem. 😊