

# USS1

Vamos nos aprofundar mais

Tradução automatizada usando o serviço IBM Globalization

- [CD CIRCLE && CD... && CD - &&. /SHAKE IT ALL ABOUT.SH](#)
- [1 LOCALIZE O TERMINAL](#)
- [2 CONECTAR-SE POR MEIO DE SSH](#)
- [3 VERIFIQUE-ME, OBRIGADO](#)
- [4 VOCÊ ESTÁ DENTRO! E AGORA?](#)
- [5 ORIENTAÇÃO](#)
- [6 MUDAR DE DIRETÓRIO COM CD](#)
- [7 ARQUIVOS E PASTAS NOVOS](#)
- [8 CONVIDAR ZOWE PARA A FESTA](#)
- [9 EU TENHO UM SEGREDO](#)
- [10 REDIRECIONAMENTO DA SAÍDA](#)
- [11 EXPLORAÇÃO ESPACIAL](#)
- [12 FAÇA VALER A PENA](#)

# CD CIRCLE && CD .. && CD - && ./SHAKE\_IT\_ALL\_ABOUT.SH

## O Desafio

O UNIX System Services ([USS](#)) é uma [POSIX](#) -implementação compatível de um ambiente UNIX no site [z/OS](#) que permite uma experiência semelhante à do site [UNIX](#) e, ao mesmo tempo, usa as mesmas APIs de sistema do site z/OS que você tem usado até agora.

Tudo isso fará mais sentido à medida que você percorrer o shell, executar scripts de shells e se apaixonar perdidamente pelo preenchimento de guias.

## Antes De Começar

Estamos começando com uma nova faceta do [z/OS](#), portanto, você não precisa saber muito mais, mas precisará do seu ambiente [VSCode](#).

O conhecimento de como os conjuntos de dados e os membros funcionam certamente ajudará a colocar as coisas em perspectiva à medida que você aprender sobre [USS](#).

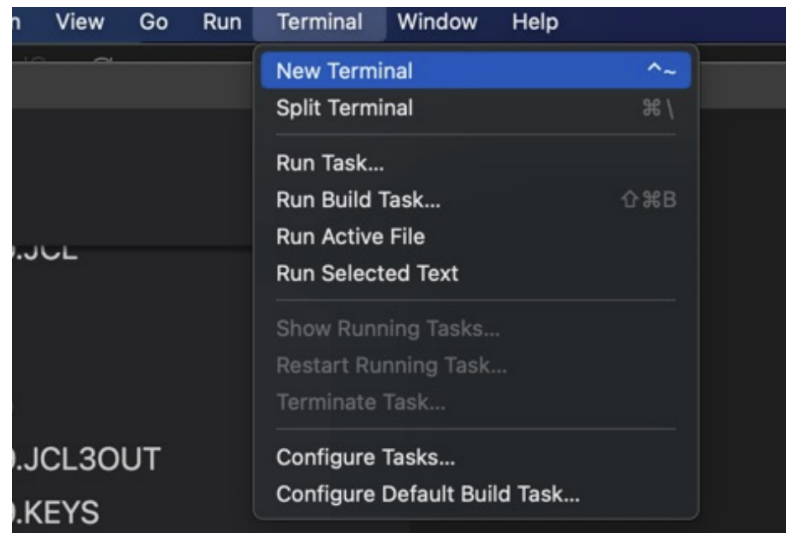
## Investimento

Etapas	Duração
12	60 minutos

# 1 FIND THE TERMINAL

Procure a seção Terminal na parte inferior da janela `VSCode` .

Se você não a vir, tente usar a opção de menu Terminal e selecionar `New Terminal` .



Esse é um método baseado em texto para interagir com seu próprio sistema pessoal.

Se você tiver uma versão mais antiga do `Windows` , talvez seja necessário fazer download de um cliente `SSH` separado, como `PuTTY` .

## Observação: Usuários do Windows

Se o seu terminal de comando padrão for PowerShell (geralmente indicado pelo prompt de comando que começa com `PS C:` ), você poderá descobrir que o login ssh falha com o erro `Permission denied, please try again` .

## Normalmente, isso não é um problema com seu ID de usuário/senha

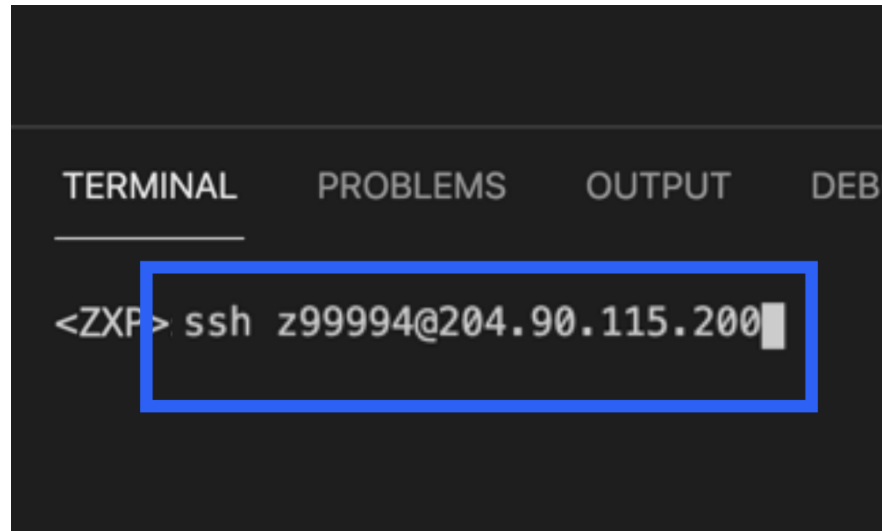
Tente mudar para um terminal de comando `CMD` - basta digitar `CMD` no terminal PowerShell e, em seguida, tentar o comando ssh novamente.

Para obter mais orientações sobre isso, consulte os fóruns de suporte.

## 2 CONNECT THROUGH SSH

Faça login no sistema `z/OS` com o seguinte comando:

`ssh z#####@204.90.115.200` (substituindo `z#####` pelo seu próprio Z-userid, é claro).



Em outras palavras, esse comando diz: "Use o comando `ssh` para me conectar (usando meu Z-userid) ao sistema remoto (no endereço IP de destino)"

Na primeira vez que você se conectar, poderá ver uma mensagem sobre a aprovação da conexão contínua:

```
The authenticity of host '204.90.115.200 (204.90.115.200)' can't be established.  
RSA key fingerprint is SHA256:1YtEA18or6MI0VQnVQn7ZUCtFVkJMRStN+DnqJZaxPk.  
This key is not known by any other names  
Are you sure you want to continue connecting (yes/no/[fingerprint])? █
```

Isso é normal - responda **yes** e continue.

## 3 VERIFY ME, THANK YOU

Agora será solicitada a sua senha, que é a mesma senha usada para fazer login no sistema z/OS por meio do site [VSCode](#).

Alguns pontos a serem observados:

- Você poderá ver "Autenticidade do host... não pode ser estabelecida"
- Você pode ser solicitado a confiar ou aceitar uma chave do sistema remoto.

Você pode responder com segurança "sim" a qualquer uma dessas solicitações.

**Você não verá nenhum caractere ao digitar sua senha!**

Isso faz com que as pessoas que olham por cima do seu ombro não consigam roubar sua senha, mas o sistema ainda pode vê-la.

**Usuários do Windows** - lembrem-se do possível problema com o ssh em PowerShell de duas etapas atrás?

## 4 YOU'RE IN! NOW WHAT?

Agora que você está conectado ao ambiente `USS` com `ssh`, pode dar uma olhada com o comando `ls`.

Isso mostrará todos os arquivos e pastas (também conhecidos como diretórios na terra do UNIX).

Seu diretório padrão ("home") estará vazio - digite o comando `uss-setup` e, em seguida, execute o comando `ls` novamente.

```

TERMINAL  PROBLEMS  OUTPUT  DEBUG CONSOLE

/z/z99994 > uss-setup
Hi and welcome the ZXPLRE UNIX System Service Challenge!
/z/z99994 > ls
code1.py      directory1    marbles.py   scramble.sh   test
code2.py      dslist.py    members.py   secret.txt
/z/z99994 > █
```

`USS` usa uma estrutura hierárquica, na qual há arquivos e diretórios dentro de outros diretórios.

Talvez você esteja acostumado com esse tipo de sistema de arquivos em seu próprio computador, onde é possível manter arquivos em pastas, colocar pastas dentro de pastas e assim por diante.

Quando quiser se desconectar, basta digitar o comando `exit` e você será desconectado. Você pode usar o comando `ssh` do seu laptop para se reconectar mais tarde.

Se você deixar a sessão `ssh` inativa por mais de alguns minutos (geralmente de 3 a 5 minutos), ela deixará de responder.

Se isso acontecer, seu terminal parecerá travado.

USS1251111-11A3



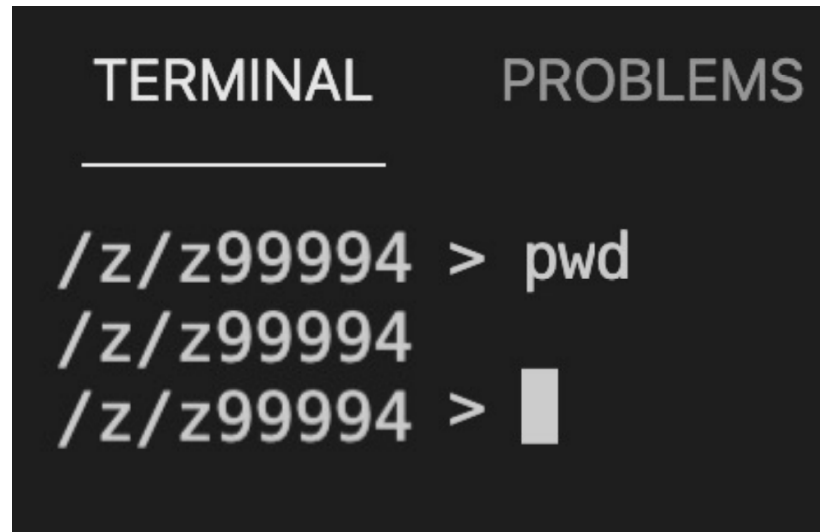
Para limpar isso, digite  - talvez seja necessário fazer isso mais de uma vez para que a sessão seja desconectada.

US51251111-1143

## 5 GETTING ORIENTATED

Além de olhar ao redor com `ls`, você também pode querer saber "Onde estou?".

O prompt de comando padrão geralmente mostra onde você está na estrutura de diretórios, mas você também pode digitar `pwd` para *imprimir o diretório de trabalho*.



A terminal window with a dark background. At the top, the words "TERMINAL" and "PROBLEMS" are displayed in a light gray font. Below "TERMINAL", there is a horizontal line. The terminal shows three lines of text: the first line is "/z/z99994 > pwd", the second line is "/z/z99994", and the third line is "/z/z99994 > " followed by a gray rectangular cursor block.

Isso será útil em breve, quando você começar a usar `cd` to *Change Directory* para se movimentar pelo sistema de arquivos.

Neste momento, você está em seu diretório pessoal, que é onde estão *seus* arquivos `USS`.

Você pode voltar a esse diretório a qualquer momento digitando `cd ~` (essa é a tecla til)

## 6 CHANGE DIRECTORY WITH CD

Para navegar para outro diretório, digite `cd`, seguido do nome do diretório.

Por exemplo, podemos digitar `cd directory1` e entraremos em `directory1`, supondo que esse seja um diretório que possamos ver com o comando `ls`.

Experimente e digite `pwd`

Você verá que o `pwd` agora mostra seu caminho como `/z/zxxxxx/directory1`

Para voltar ao seu diretório inicial, é necessário voltar um nível. Você pode fazer isso digitando `cd ..` (dois pontos) ou pode usar o atalho do til para ir direto ao seu diretório pessoal de qualquer lugar (`cd ~`)

Até agora, você tem andado para frente e para trás, um passo de cada vez. Você também pode alterar os diretórios especificando o caminho completo para o diretório para o qual deseja ir.

Por exemplo, `cd /z/public/test` o levará diretamente para esse novo local.

```
/z/z99994 > cd directory1/
/z/z99994/directory1 > pwd
/z/z99994/directory1
/z/z99994/directory1 > cd ..
/z/z99994 > pwd
/z/z99994
/z/z99994 > cd directory1/
/z/z99994/directory1 > cd ~
/z/z99994 > pwd
/z/z99994
/z/z99994 > cd /z/public/test/
/z/public/test > pwd
/z/public/test
/z/public/test > cd ~
/z/z99994 > pwd
/z/z99994
/z/z99994 >
```

Em seguida, use o site `cd ~` para voltar para casa, depois de ter ido até lá e dado uma olhada.

(Novamente, se seu diretório pessoal estiver vazio, execute o comando `uss-setup` )

## FACILITAR A VIDA. USAR O PREENCHIMENTO DE GUIAS E AS SETAS PARA CIMA/PARA BAIXO

Percebeu como a palavra "diretório" se torna irritante quando é preciso digitá-la depois de algum tempo? Da próxima vez que tiver que digitar um comando ou nome de arquivo longo, digite as primeiras letras e pressione a tecla Tab.

Ao pressionar uma vez, o comando ou o nome será preenchido automaticamente o máximo possível, e ao pressionar duas vezes, você verá todas as possíveis conclusões. Portanto, se você tiver um "directory1" e um "directory2", poderá digitar `cd di` e, em seguida, pressionar a tecla Tab, e a parte "directory" será preenchida automaticamente.

Pressione-o mais uma vez e ele informará "there's directory1 and directory 2".

Experimente e você ficará surpreso com a rapidez com que pode usar o prompt de comando.

Mais um truque: se quiser usar um comando que digitou há pouco tempo, você pode recuperar comandos recentes pressionando a seta para cima no teclado. Em seguida, basta pressionar Enter para usá-lo. Muito legal, não é?

Se você precisar alterar um comando anterior, use a tecla Backspace para remover os caracteres indesejados e, em seguida, digite o que você precisa.

US51251111-1143

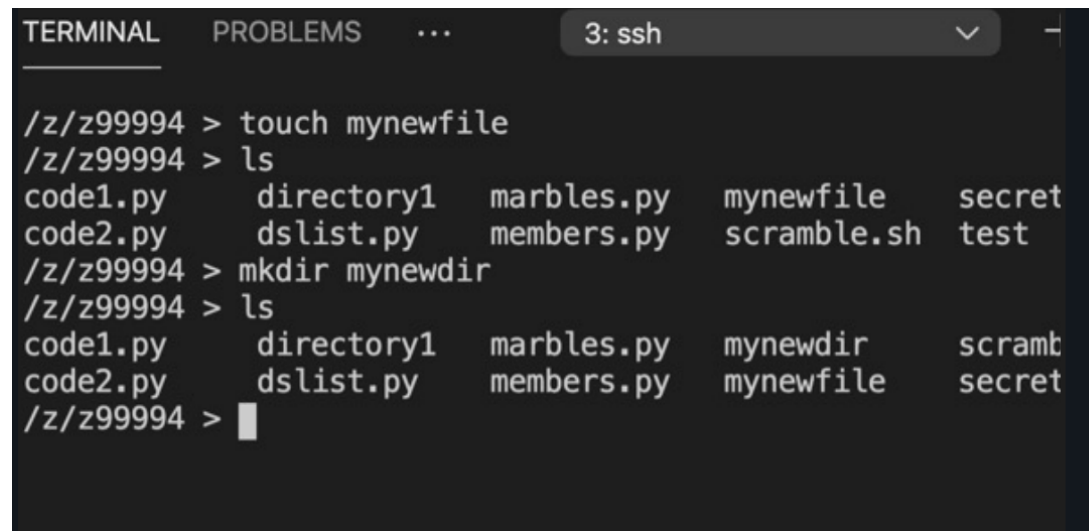
## 7 FRESH FILES AND FOLDERS

O comando `touch` é normalmente usado para atualizar o registro de data e hora da "última modificação" de um arquivo, mas também pode ser usado para criar um arquivo vazio.

Digite o comando `touch mynewfile` e siga-o com um `ls` para ver o novo arquivo.

Você pode criar diretórios totalmente novos com o comando `mkdir`.

Por exemplo, experimente `mkdir mynewdir` e você verá um novo diretório brilhante depois de executar `ls` novamente.



```
TERMINAL  PROBLEMS  ...  3: ssh  v  -
/z/z99994 > touch mynewfile
/z/z99994 > ls
code1.py    directory1  marbles.py  mynewfile   secret
code2.py    dslist.py  members.py  scramble.sh  test
/z/z99994 > mkdir mynewdir
/z/z99994 > ls
code1.py    directory1  marbles.py  mynewdir    scramb
code2.py    dslist.py  members.py  mynewfile   secret
/z/z99994 > 
```

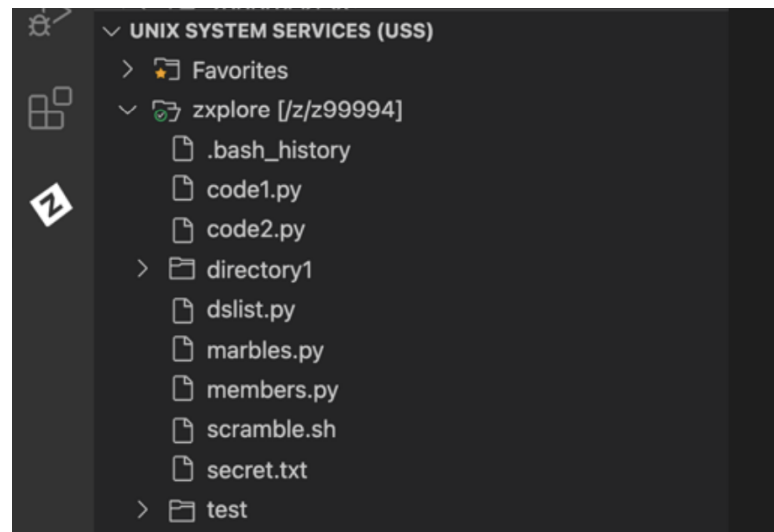
US51251111-1143

## 8 INVITE ZOWE TO THE PARTY

Você pode visualizar seus arquivos e diretórios **USS** em **Zowe**.

Basta clicar na lupa ao lado de seu perfil na visualização **USS** e digitar o caminho completo de seu diretório pessoal.

Seu diretório pessoal é **/z/z#####**



Certifique-se de usar **todas as letras minúsculas** e de usar *seu* Z-userid, e não z##### ou z99994 )

## 9 I'VE GOT A SECRET

Você tem um programa chamado `scramble.sh` em seu diretório pessoal.

Você pode saber que se trata de um programa executável porque, quando você digita o comando `ls -l`, ele aparece com um "x" no quarto lugar das permissões. Isso significa que, além de poder *ler* e *gravar*, você também pode *executá-lo*.

Há muito mais a saber sobre os bits de permissão, mas deixaremos isso para uma aula mais avançada de USS, talvez.

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

/z/z99994 > ls -l scramble.sh
-rwxr-xr-x  1 Z99994  IPGROUP    1015 Jul 26 15:18 scramble.sh
/z/z99994 > ./scramble.sh
Usage Example: ./scramble.sh file.txt 13

This program takes the input from a file (first argument)
and rotates the letters by a number of positions (second argument)
If only works for lowercase characters.
The first argument needs to be a file.

Your task is to figure out the correct number of rotations needed
to decode the secret message in /z/public/secret.txt. Good luck!
/z/z99994 > ./scramble.sh /z/public/secret.txt 9
/z/public/secret.txt exists.
Processing.....Done!

Output:
ugfyjslmdslagfk! qgm mfkujsetdvw lzw ewkksyw af lzw mkk uzsddwfyw!!
/z/z99994 >
```

Por enquanto, saiba que você pode executar o programa com o comando

`./scramble.sh`

e o resultado do programa lhe dirá tudo o que você precisa saber. Boa sorte!

*Dica* : O valor correto para o número de rotações está entre 1 e 26. Use suas habilidades de dedução para tentar descobrir o valor no menor número possível de tentativas.



## "CRIEI UM MONTE DE ARQUIVOS E PASTAS; COMO FAÇO PARA ME LIVRAR DELES?"

Para remover um arquivo, digite `rm` seguido do nome do arquivo que deseja excluir. Por exemplo, `rm mynewfile`

Você também pode usar o comando `rmdir` para apagar um diretório vazio. Por exemplo, `rmdir mynewdir` eliminará o diretório que você criou na Etapa 7.

Há maneiras de especificar diretórios não vazios, bem como maneiras de excluir muitos arquivos de uma só vez. Se você tem alguma experiência em um ambiente de shell, provavelmente conhece esses comandos. Caso contrário, provavelmente é melhor não mostrá-los a você tão cedo. Não gostaríamos que você excluísse todo o seu trabalho árduo.

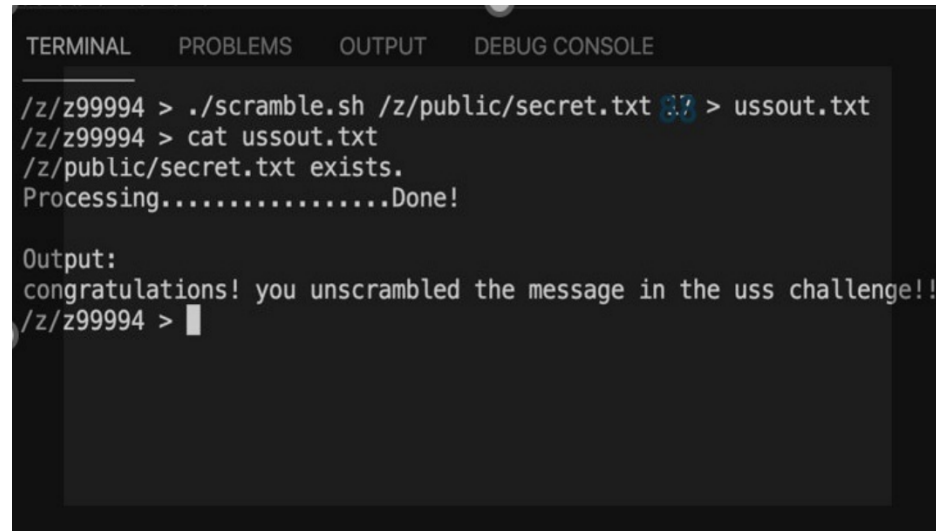
Se você *excluir* acidentalmente um arquivo, geralmente poderá encontrar a cópia original dele em `/z/public`. Use o comando `cp` para copiá-lo.

*Exemplo :* `cp /z/public/test ~/test` lhe dará uma nova cópia de 'test' em seu diretório pessoal.

US51251111-11A3

## 10 REDIRECTING THE OUTPUT

Agora que você decifrou o código, coloque a saída do programa em um arquivo. Isso é muito fácil de fazer, usando o "redirecionamento".



```
TERMINAL  PROBLEMS  OUTPUT  DEBUG CONSOLE

/z/z99994 > ./scramble.sh /z/public/secret.txt > ussout.txt
/z/z99994 > cat ussout.txt
/z/public/secret.txt exists.
Processing.....Done!

Output:
congratulations! you unscrambled the message in the uss challenge!!
/z/z99994 > 
```

Digite ou recupere seu comando de embaralhamento bem-sucedido recente com os valores corretos e adicione `> ussout.txt` ao final do comando, para que ele se pareça com a captura de tela acima.

Você não verá a saída enquanto o comando estiver funcionando.

Quando você coloca `'>'` após um comando ou qualquer coisa que produza saída, isso significa "Em vez de colocar a saída na tela, salve-a neste arquivo".

Se o arquivo ainda não existir, um novo será criado para você, mas tenha cuidado... *o redirecionamento também substituirá qualquer coisa em um arquivo existente.*

Você pode verificar a saída com o comando `cat ussout.txt`.

Você pode usar `'>>'` para redirecionar a saída e anexá-la ao final do arquivo de saída, que é exatamente o que faremos na Etapa 11.

Você pode verificar o que está no arquivo abrindo-o com o VSCode na visualização USS.

USS1251111-1143

(Lembre-se de que talvez seja necessário clicar com o botão direito do mouse e selecionar "Pull from Mainframe" para atualizar a exibição depois de gravar nela)

US5125111-1A3

# 11 SPACE EXPLORATION

Depois de ter a saída decodificada em seu arquivo `ussout.txt`, use o redirecionamento para anexar ao final desse arquivo ( *não* substitua!) com a saída de `du -ak`

O comando `du` gera o *uso de disco* do diretório em que você está no momento, bem como de todos os diretórios dentro/abaixo desse diretório (se você especificar a opção '-a') e fornecerá o resultado em kilobytes (é para isso que serve a opção '-k').

```

TERMINAL  PROBLEMS  OUTPUT  DEBUG CONSOLE

/z/z99994 > du -ak
      8  ./bash_history
      8  ./code1.py
      8  ./code2.py
      0  ./directory1/you found me
      8  ./directory1
      8  ./dslist.py
      8  ./marbles.py
      8  ./members.py
      8  ./scramble.sh
      8  ./secret.txt
      0  ./test
      8  ./ussout.txt
     88  .
/z/z99994 > █
```

A saída desse comando variará à medida que fornecermos mais amostras. (Não se preocupe se a captura de tela for diferente para seus resultados)

Se você tiver feito isso corretamente, seu arquivo `ussout.txt` deverá ter:

1. mensagem secreta decodificada, seguida por
2. a saída do comando de uso do disco.

Verifique o conteúdo de sua saída com `cat ussout.txt`

USS1251111-11A3

## 12 MAKE IT COUNT

```
/z/z99994 > du -ak >> ussout.txt
/z/z99994 > date >> ussout.txt
/z/z99994 > cat ussout.txt
/z/public/secret.txt exists.
Processing.....Done!

Output:
congratulations! you unscrambled the message in the uss challenge!!
 8 ./bash_history
 8 ./code1.py
 8 ./code2.py
 0 ./directory1/you found me
 8 ./directory1
 8 ./dslist.py
 8 ./marbles.py
 8 ./members.py
 8 ./scramble.sh
 8 ./secret.txt
 0 ./test
 8 ./ussout.txt
88 .
Mon Jul 26 15:54:00 CDT 2021
/z/z99994 > |
```

Por fim, use o redirecionamento para anexar a saída do comando `date` ao mesmo arquivo.

Certifique-se de que seu arquivo `ussout.txt` agora tem

1. a mensagem secreta decodificada
2. sua saída de uso de disco
3. a saída do comando `date`

Se tudo estiver correto, envie o trabalho de validação `CHKUSS1` de `ZXP.PUBLIC.JCL` e, se tudo correr bem, você estará pronto!

Bom trabalho - vamos recapitular	A seguir.
<p>Você se conectou a USS por meio de um shell seguro (SSH) e também por meio do plug-in Zowe em VSCode. A partir daí, você navegou por diretórios, gerenciou a saída, invadiu alguns textos e aprendeu tudo sobre argumentos. Pode não parecer muito, mas pense no que você já fez e no que sabia quando começou. Se você gosta da maneira UNIX de fazer as coisas, provavelmente gostará muito do desafio baseado em LINUX que será disponibilizado mais tarde. Se essa não foi a sua "xícara de chá", pelo menos agora você terminou!</p>	<p>Continuaremos trabalhando no espaço <code>USS</code> no próximo desafio, portanto, não deixe de ter à mão os comandos do terminal e as guias <code>VSCode</code>.</p> <p>Faremos algumas codificações bem básicas usando o site Python.</p> <p>Não entre em pânico se você nunca programou antes! Mostraremos tudo o que você precisa.</p>

USS1251111-1143