

Forum Programmation.shell : Ligne de code qui refuse d'être factorisée

Posté par [gzgrhe](#) le 17/11/23 à 21:33. [Licence CC By-SA](#).

Étiquettes : aucune

Bonjour à tous les pro du shell... et du bash en particulier,

J'ai écrit un script générant une chaîne de caractères de longueur et complexité variable selon les besoins, histoire de faciliter la génération de login/password pour les services accessibles sur Internet.

J'ai voulu factoriser les sections redondantes pour avoir une meilleure lisibilité et simplifier d'éventuels changements futurs et je fais face à une difficulté.

Initialement, la ligne à factoriser dans une fonction ressemblait à ça :

```
strings --bytes=1 < /dev/urandom | tr --delete a-z | head --bytes="${LENGHT}" | xargs --null
```

Pour avoir plus de complexité; je changeais le a-z en azA-Z, A-Z0-9, '[:alnum:]', etc.

Et ça se factorise sans problèmes dans cette fonction :

```
_get_random()
{
    local char_set
    local symbol

    symbol="$1"
    char_set="$2"

    printf "%s\n" "The generated password is : "
    strings --bytes=1 < /dev/urandom | tr --delete --complement "${symbol}" | head --bytes="${LENGHT}" | xargs --null
    printf "%s\n" "The password entropy is : $(_entropy ${char_set}) bits."
}
```

Et ça s'utilise avec par exemple :

```
_get_random a-z 26
```

Le soucis est avec cette ligne qui génère les chaînes les plus complexes :

```
strings --bytes=1 < /dev/urandom | tr --delete '[:space:]' | head --bytes="${LENGHT}" | xargs --null
```

Curieusement le '[:space:]' ne passe pas alors que le '[:alnum:]' passe comme une lettre à la poste.

Selon vous, d'où vient mon erreur ?

PS : j'ai du mal avec la mise en page du code, il y a le `symbol` qui a merdé.

Question XY ?

Posté par [jeanas](#) le 17/11/23 à 22:38. Évalué à 3 (+2/-0).

J'ai écrit un script générant une chaîne de caractères de longueur et complexité variable selon les besoins, histoire de faciliter la génération de login/password pour les services accessibles sur Internet.

Première question : tu ne peux pas utiliser la commande `pwgen` ?

PS : j'ai du mal avec la mise en page du code, il y a le `symbol` qui a merdé.

Le code se met dans un bloc encadré par ````. N'hésite pas à utiliser l'aide-mémoire Markdown qui s'affiche en-dessous de la fenêtre de rédaction.`

Re: Question XY ?

Posté par [gzgtrhe](#) le 17/11/23 à 22:51. Évalué à 1 (+0/-0).

tu ne peux pas utiliser la commande pwgen ?

Je ne connaissais pas cette commande.
Elle n'est pas installée sur ma machine...

Le code se met dans un bloc encadré par ```

J'avais vu mais ça ne marche pas :(

Re: Question XY ?

Posté par [jeanas](#) le 17/11/23 à 23:16. Évalué à 2 (+1/-0).

Je ne connaissais pas cette commande. Elle n'est pas installée sur ma machine...

Essaie de l'installer, elle sera probablement dans le paquet `pwgen`. Sous Debian ou dérivés (comme Ubuntu), ça devrait être `sudo apt install pwgen`, sous Fedora, `sudo dnf install pwgen`, ...

J'avais vu mais ça ne marche pas :(

PourMoiÇamarche@_(ツ)_/

```
_get_random()
{
    local char_set
    local symbol

    symbol="$1"
    char_set="$2"

    printf "%s\n" "The generated password is :"
    strings --bytes=1 < /dev/urandom | tr --delete --complement "${symbol}" | head --bytes="{LENGHT}" | xargs --null
    printf "%s\n" "The password entropy is : (_entropy{char_set}) bits."
}
```

Re: Question XY ?

Posté par [Cyril Brulebois \(site web personnel\)](#) le 18/11/23 à 02:14. Évalué à 3 (+1/-0).

Merci pour tes efforts de reformatage mais attention :

- `${symbol}` (permutation des deux premiers caractères) ;
- `{LENGHT}` (manque le dollar) — dont ça n'est pas l'orthographe au passage.

Je ne comprends pas le problème de [gzgtrhe](#) :

1. Que signifie « ne passe pas » ?
2. Le code en dehors de la fonction utilise `tr --delete`, le code dans la fonction utilise `tr --delete --complement`.

Alors oui, si on passe en paramètre `[:space:]`, on demande à supprimer tout ce qui n'est pas espace, et il reste des espaces. Et retour au premier point.

--

Debian Consultant @ DEBAMAX

Re: Question XY ?

Posté par [gztgrhe](#) le 18/11/23 à 11:05. Évalué à 2 (+1/-0).

Le code en dehors de la fonction utilise `tr --delete`, le code dans la fonction utilise `tr --delete --complement`.

C'était tellement évident que je suis passé à côté.

Merci !

Re: Question XY ?

Posté par [gUI \(Mastodon\)](#) le 18/11/23 à 08:23. Évalué à 3 (+0/-0).

J'ai reformaté le journal pour plus de lisibilité.

Le code se met dans un bloc encadré par ```

Pour rappel :

- il faut sauter une ligne pour que ce soit reconnu
- on peut même donner un langage afin d'avoir la coloration :

```
```bash
echo "toto"
```
```

donnera :

```
( echo "toto"
```

--

En théorie, la théorie et la pratique c'est pareil. En pratique c'est pas vrai.

Re: Question XY ?

Posté par [gztgrhe](#) le 18/11/23 à 10:49. Évalué à 1 (+0/-0).

Merci pour avoir corrigé la mise en forme.

Re: Question XY ?

Posté par [cg](#) le 18/11/23 à 11:01. Évalué à 3 (+1/-0).

J'ajoute que si tu utilises un gestionnaire de mots de passes (keepassxc, bitwarden...), il y a des générateurs intégrés assez pratiques. Et si tu as plein de mots de passes, je t'encourage à t'y mettre :).

Ceci dit ça ne fait jamais de mal de faire un peu de shell.

Re: Question XY ?

Posté par [Krunch \(site web personnel\)](#) le 18/11/23 à 11:48. Évalué à 2 (+0/-0).

Ceci dit ça ne fait jamais de mal de faire un peu de shell.

C'est vraiment très faux.

--
pertinent adj. Approprié : qui se rapporte exactement à ce dont il est question.

Re: Question XY ?

Posté par [cg](#) le 18/11/23 à 11:58. Évalué à 2 (+0/-0).

Comme toutes les généralités, d'ailleurs. (attention, boucle infinie :D)

Re: Question XY ?

Posté par [Benoît Sibaud \(site web personnel\)](#) le 18/11/23 à 12:10. Évalué à 3 (+0/-0).

Deux salles deux ambiances

Il existe deux types d'informaticien-ne-s : ceux qui utilisent la ligne de commande et ceux qui sont à la retraite. Même les développeurs qui adoptent des interfaces de développement très intégrées disposent d'une sorte de terminal dans certains panneaux de leur IDE. Qu'on le veuille ou non, l'interpréteur de commandes fait partie intégrante de la vie quotidienne de la plupart des personnes qui attendent de leur ordinateur qu'il travaille pour elles.

(source <https://linuxfr.org/news/comparaison-critique-de-systemes-d-invite-de-commande>)

Re: Question XY ?

Posté par [toto2000](#) le 18/11/23 à 13:23. Évalué à 2 (+0/-0).

Si la ligne de commandes et le shell ne servait à rien, Microsoft n'aurait pas inventé Powershell.

Comme toujours, utiliser l'outil adapté au besoin. Il y a des cas où un bon vieux script shell suffira à couvrir un besoin - là où d'autres, ne connaissant pas l'outil, viendront à utiliser du python ou autre langage, qui constituera une brique nécessitant d'être suivie en terme d'obsolescence, alors que le script shell avec commandes de base en général n'a pas à bouger et sera mis à jour avec le système de base.

Note : les commentaires appartiennent à celles et ceux qui les ont postés. Nous n'en sommes pas responsables.