

1.0 La commande od

Quelle option permet d'afficher les adresses en hexadécimal au lieu d'octal ? `-address-radix=x` Quelle option permet d'afficher chaque octet d'un fichier en hexadécimal ? `format=x` Quelle option permet de sauter un certain nombre d'octets ? `jN`, `N` étant le nombre d'octets
Quelle option permet de n'interpréter qu'un certain nombre d'octets ? `NN`, le second `N` étant le nombre d'octet

Utilisation de od

la valeur 192, nombre entier non signé codé sur 8 bits 192 sur 8 bit est 11000000 donc `c0` le resultat peut etre obtenu avec `od -tx1 -N0` valeurs la valeur -1, nombre entier signé codé sur 8 bits, -1 sur 8 bit est, en utilisant le complement a 2, $256-1 = 255$, donc `FF` `od -tx1 -j1 -N1` valeurs la valeur -2, nombre entier signé codé sur 16 bits, -2 sur 16 bit est $65536-2 = 65534$, donc `FFE` `od -tx1 -j1 -N1` valeurs
comment un nombre peut avoir differentes valeurs suivant la maniere dont il est lu la valeur 163837924, nombre entier non signé codé sur 32 bits, par divisions euclidiennes successives automatise, j'obtient `0x616F6C4C` la valeur 1.0, nombre réel codé au format IEEE 754 32 bits (nombre à virgule flottante), `0 1111111 00000...n23` `od -tff` la valeur -10.25, nombre réel codé au format IEEE 754 64 bits (nombre à virgule flottante),

1 100000010 01...

Encodage des caracteres

À la suite des valeurs numériques précédentes de l'exercice précédent, se trouve également dans le fichier la suite de caractères ASCII `abcdABCD0123`. Quel est le code ASCII en hexadécimal des caractères `a` et `A` (`od` accepte l'accumulation de plusieurs formats en paramètres) ? Donnez une méthode qui permet de passer rapidement des minuscules aux majuscules un caractère ASCII, et inversement. la valeur de `A` est 65, celle de `a` est 97 on peut additionner par -25 ou 25 pour passer rapidement de majuscule vers minuscule et vice versa l'ascii de 0 est 48, et de 9 57, on peut simplement additionner par 1 pour avoir chaque caractere representant un chiffre Vérifiez vos réponses à l'aide des instructions Python `ord` et `chr`. `[chr(48+i) for i in range(10)]` renvoie `['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']` `[chr(97+i) for i in range(26)]` renvoie l'alphabet

Dans le même répertoire `/home/partage/I22/` se trouve trois autres fichiers : `texte1`, `texte2` et `texte3`.

À l'aide des commandes `cat` et `less`, visualisez le contenu de ces fichiers. Voyez-vous une différence majeure dans leur affichage ? En fait ces trois fichiers comportent le même texte mais écrit dans un encoding différent. À l'aide de la commande `od` et des caractéristiques des encodings ISO-8859 (latin-1), UTF-8 et UTF-16,

quels indices vous permettent de déterminer quel encoding est utilisé par quel fichier. text1 et 3 utilisent utf-16, text2 utilise iso-8859 Étudiez la commande file et validez votre réponse à la question précédente. Que donne la réponse de la commande file pour le premier fichier valeurs ? Pourquoi data, car le fichier ne contient pas de headers, c'est donc seulement un fichier binaire pour l'os

1.6 Jeux de piste

Dans cet exercice, vous allez utiliser la commande od pour retrouver une information fragmentée et cachée dans le fichier valeurs.

Trouvez chaine1, la première chaîne de caractères encodée en ASCII, qui débute à l'adresse 37 et fait 9 caractères de long.

10.99.185.XXX Trouvez chaine2 la chaîne de deux caractères encodée en UTF-16 qui se situe 12 octets après chaine1. Trouvez la valeur val qui est un entier non signé codé sur deux octets, situé 17 octets après chaine2. Effectuez dans un terminal la commande curl suivi de la concaténation directe (sans insérer d'espace) de chaine1, chaine2 et de val. Qu'obtenez-vous ?