Modifier pdf sur linux

droit, sécurité, fichier, administration Voir aussi les droits. Les systèmes d'exploitation inspirés d'Unix (dont Linux fait partie) ont la possibilité de déterminer soigneusement la gestion des droits d'accès à divers fichiers de votre système d'exploitation. Les droits d'accès établis sur les fichiers et les répertoires ne sont pas les mêmes : ils peuvent être modifiés pour répondre aux nouveaux besoins qui surviennent au fil du temps dans votre système Ubuntu. Cet article explique les différentes manipulations qui peuvent être appliquées aux fichiers et répertoires afin de changer de propriétaire et d'autorisations. Les droits d'accès déterminent la propriété d'un fichier ou d'un répertoire pour l'utilisateur et le groupe d'utilisateurs. Ils contrôlent également les actions que les utilisateurs ont le droit de faire sur les fichiers (lecture, écriture et exécution), selon qu'ils possèdent ou non le fichier. La propriété et la gestion des autorisations connexes se font individuellement avec chaque fichier. Ce document décrit les différentes manipulations qui peuvent être effectuées sur les fichiers et les répertoires afin de modifier leurs différents droits d'accès. Il tient compte de votre connaissance des catégories des propriétaires de fichiers (propriétaire de l'utilisateur, groupe propriétaire, autres) et de trois types d'autorisations (lecture, écriture et exécution); toutes ces informations sont regroupées dans une explication au document « Linux Access Rights: File Access Management ». La description de ces attributs ne sera pas discutée ci-dessous; Prenez également le temps de lire l'explication du document avant de continuer à lire cet article. Sous Nautilus (Ubuntu), cliquez directement sur le fichier ou le répertoire, puis sélectionnez les propriétés. Accédez à l'onglet autorisation. Pour le fichier catalogue du propriétaire et du groupe, vous pouvez choisir parmi le menu d'automne approprié pour donner le droit de lire, d'écrire ou de lire seul. Pour d'autres, vous pouvez choisir entre la lecture et l'écriture, la lecture seule, et personne. Vous pouvez choisir le groupe auquel appartient le fichier (par défaut, le groupe propriétaire, sauf dans des cas spécifiques). En ce qui concerne le répertoire, pour le propriétaire et le groupe, vous pouvez choisir entre la création et la suppression de fichiers, l'accès aux fichiers et la liste uniquement des fichiers. Pour d'autres, vous pouvez choisir entre la création et la suppression de fichiers, l'accès aux fichiers, la liste des fichiers uniquement et non. La section suivante (dans la ligne d'ordre) détaille un peu plus d'autorisations différentes. Les droits de fichiers d'annuaire peuvent apparaître par l'équipe Les droits d'accès apparaissent dans une liste de 10 caractères. : drwxr-xr-x Le premier symbole peut être - d ou l, entre autres (toutes les variantes de la page d'autorisations Unix dans Wikipedia)). Il souligne la nature du fichier: Ensuite, suivez 3 groupes de 3 caractères chacun, indiquant si le fichier (ou le répertoire) est autorisé à lire, écrire ou exécuter. Dans cet ordre, ces trois groupes correspondent aux droits du propriétaire, du groupe et du reste des utilisateurs. Dans le paragraphe d'introduction, vous remarquerez des lettres en gras en anglais. Ce sont ces lettres qui sont utilisées pour symboliser ces autorisations. Si la permission n'est pas accordée, la lettre en question est remplacée par une lettre. Si nous prenons les lettres, les données à lire / écrire / performance (lire / écrire / effectuer), nous obtenons: rwx. Une autre équipe pratique vous permet de visualiser à la fois les droits (et les propriétaires) de tous les répertoires parentaux (voir les chemins) d'une ressource particulière: namei-mo/chemin/ Prenons l'exemple théorique précédent: drwxr-xr-x Cela se traduit comme suit: Dans la pratique, l'exécution de la commande suivante: ls-one reçoit une liste du contenu du catalogue actuel, par exemple: drwxr-xr-x 6 cyrille cyrille cyrille 4096 2008-10-29 23:09 Bureau drwxr-x--- 2 cyrille cyrille 4096 2008-10-22 22:46 Documents lrwxrwxrx Cyril 1 Cyril 1 Cyril 26 22 008-09-22 22:30 Exemples ---usr/share/example-content-rw-r-r-r-r-xr-x 7 Cyril 1544881 2008-10-18 15:37 forum.xcf drwxr-xr-x 7 cyrille cyrille cyrille 00 8-0 9-23 18:16 Images drwxr-xr-x 2 cyrille cyrille 4096 2008-09-22 22:45 Modèles drwxr-xr-xr-x 267 Cyrille Cyril 20480 2008-10-27 22:22 :22:22 17 Musique dr drwxr-xr-x 2 Cyrille Cyril 4096 2008-09-22 22:45 Public drwxr-xr-x 2 Cyrille 4096 2008-10-26 13:14 Vidéo Première colonne contient un groupe de 10 caractères pour connaître les droits de chaque fichier. Donc, pour le fichier forum.xcf, nous avons: -rw-r--- Les informations ci-dessous est très important pour la compréhension et la maîtrise des autorisations. Toutefois, si vous voulez calculer rapidement le coût des outils d'autorisation le faire pour nous. Par exemple: la calculatrice CHMOD à Nautilus, il vous suffit de modifier les valeurs du menu de dépôt dans l'onglet résolution (voir ci-dessus) ou non. Le fichier a le propriétaire et le groupe. On peut les changer. L'équipe chown (changement de propriétaire, changement de propriétaire) vous permet de changer le propriétaire du fichier. Seul un super-utilisateur ou le propriétaire actuel du fichier peut utiliser chown. L'équipe utilisée comme suit : le fichier file1 file1 sudo chown toto appartient désormais au toto utilisateur. L'équipe chgrp (pour le groupe de changement) vous permet de changer le groupe dans lequel Tous les membres de ce groupe seront concernés par la résolution de la 2ème série rwx. Encore une fois, seul un super utilisateur ou le propriétaire actuel du fichier peut utiliser chgrp (un membre du groupe ne peut pas changer le groupe du propriétaire). L'équipe utilisée comme suit: sudo chgrp mesPotes file2 File2 fait maintenant partie du groupe mesPotes. Tous les membres de l'équipe mesPotes auront accès à ce fichier conformément aux autorisations du groupe. Lorsque l'utilisateur actuel n'est pas le propriétaire actuel du fichier, il sera nécessaire de précéder l'ordre du navire, car cela doit être fait avec les droits de l'administration. Vous pouvez utiliser une certaine syntaxe d'équipe chown pour modifier à la fois le propriétaire et le groupe de propriété. Encore une fois, seul le super-utilisateur ou le propriétaire actuel du fichier peut utiliser chown (un membre du groupe ne peut pas apporter une modification au propriétaire). La commande est utilisée comme suit: chown nouveau_propriétaire:nouveau_groupe_propriétaire nom_du.file Lorsque l'utilisateur actuel n'est pas le propriétaire actuel du fichier, vous devrez précéder l'ordre sudo, comme cela doit être fait avec les droits d'administration. Imaginez le même fichier foo.txt appartenant à user1 et appartenant à un groupe de propriétaires de Group1. Le propriétaire doit devenir un utilisateur2, et la propriété du groupe du fichier doit aller au groupe2. Connectée au compte d'utilisateur1, cette commande effectuera l'opération demandée : chown user2:group2 foo.txt Tool chmod (mode de modification, résolution de modification) vous permet de modifier les autorisations dans le fichier. Il peut être utilisé de deux façons : soit en spécifiant les autorisations octally, à l'aide de numbers1) en ajoutant ou en supprimant des autorisations à une ou plusieurs catégories d'utilisateurs à l'aide des caractères r w et x que nous avons introduits ci-dessus. Nous préférons introduire cette deuxième méthode (ajouter ou supprimer des autorisations avec des symboles) car elle est probablement plus intuitive pour les néophytes. Il suffit de savoir que ces deux méthodes sont équivalentes, ce qui signifie qu'elles affectent toutes deux les résolutions de manière égale. Ainsi, vous pouvez choisir: Par exemple, chmod o-w file3 supprime le droit d'écrire pour les autres. chmod a-x ajoutera le droit d'exécution à tout le monde. Vous pouvez également combiner plusieurs actions en même temps : chmod u-rwx, g-rx-w, o-r-wx file3, etc. : Revenons au catalogue de documents. Ses autorisations: drwxr-x--- En octal, Nous aurons 750: rwx r-x --- 7 (4-2-1) 5 (4-0-1) 0 (0-0-0) Pour mettre ces autorisations dans le catalogue, nous entrerons la commande: chmod 750 Documents pour chacune de ces équipes, ils peuvent être re-catalogue. Autrement dit, l'action sera effectuée sur le répertoire spécifié et sur tous les fichiers ou catalogues qu'elle contient. Cela se fait en ajoutant une option -R. Attention! Le chmod-R mal utilisé peut rendre votre système inutilisable de façon permanente. Voir chmod -R / Par exemple: chmod -R 750 myRepertory donnera tous les droits au propriétaire, le droit de lire et d'exécuter le groupe et aucun droit à d'autres ... En effet, si les répertoires ont besoin d'avoir la permission x d'être ouvert, permission x inutile pour les fichiers non performants et peut irriter les fichiers texte (txt, HTML ...) parce que dans ce cas, lorsque vous les ouvrez, nous aurons à chaque fois un message demandant si nous voulons les ouvrir ou les exécuter (comment vous avez exécuté). En bref, le droit x ne doit être réservé qu'aux fichiers réellement exécutés. Annexe 1 : Soit un catalogue monrec contenant sous-répertoire et fichiers. Droits drwx-- (700) pour les annuaires et -rw—— (600) pour les fichiers. Nous voulons ré-ajouter les mêmes droits (resp. rwx et RW) au groupe. Autrement dit, nous voulons nous retrouver avec la situation suivante: drwxrwx- (770) pour les répertoires et -rw-rw-- (660) pour les fichiers. Si nous courons chmod -R 770 monrep: les fichiers auront la bonne exécution → mauvais Si nous courons chmod -R 660 monrep: catalogues n'auront plus la bonne exécution → catastrophique Si nous courons chmod -R g-rwx monrep: les fichiers ne seront plus éligibles Exécution → désastreuse Si nous courons chmod-R g-rwx monrep: les fichiers auront la bonne exécution → mauvais Si vous exécutez chmod-R g-rwX monrep: seuls les répertoires (et les fichiers déjà exécutés) auront la bonne exécution → une bonne application 2: Imaginez que nous avons déjà lancé le chmod -R 770 monrep commande. La situation est la suivante: droits drwxrwx- (770) pour les annuaires et -rwxrwx-- (770) pour les fichiers. Nous voulons supprimer les droits d'exécution uniquement sur les dossiers. Autrement dit, nous voulons nous retrouver avec la situation suivante: drwxrwx- (770) pour les répertoires et -rw-rw-- (660) pour les fichiers. Depuis chmod s'applique à la fois aux fichiers et répertoires, nous allons jongler avec x et X. Vous devez supprimer x, puis ajouter X. Si vous exécutez chmod-are you-x-X, g-x-X monrep il n'aura aucun effet parce que X touche à la fois les catalogues et les fichiers qui ont x quelque part. Donc, si you-x supprime le premier x (qui donne -rw-rwx-), la suite X passe immédiatement x parce qu'il ya encore x (l'un des groupes Donc, d'abord, vous devez supprimer tous les x: u-x,g-x avant de les transférer (à faire uniquement pour les répertoires cette fois), qui donne finalement: chmod-are you-x,g-x,u-x,g-X monrep Il convient de noter que seul le propriétaire du fichier ainsi que le super-utilisateur ont la possibilité de changer les autorisations pour le fichier. (Un membre de l'équipe de propriété ne peut pas modifier les autorisations dans le fichier.) Lorsque l'utilisateur actuel n'est pas le propriétaire actuel du fichier, il sera nécessaire de précéder l'ordre du navire, car cela doit être fait avec les droits de l'administration. Notez également que pour modifier les propriétaires et les autorisations pour le fichier qui leur appartient, l'utilisateur doit avoir la permission d'écrire dans ce fichier. S'il n'a qu'un permis de jeu, il ne pourra apporter aucune modification au droit d'accès au fichier. Les droits sont parfois répertoriés avec 4 chiffres, tels que file_mode-0777. Ce premier numéro ajouté à partir de l'avant peut être utilisé pour déterminer: le type de chaîne dans l'ordre: ls-l/usr/bin Vous devriez voir dans la liste des noms de fichiers sur un fond rouge ou jaune et le type droit ci-dessous ou s (special2)) remplace x-rwsr-xr-x 1 racine racine 155008 fév. 10 2014 sudo-rwxr-sr-x 1 root ssh 284784 12 mai 2014 ssh-agent Set-User-ID bit permet à l'utilisateur d'exécuter le programme avec les droits du propriétaire, c'est ainsi que sudo nous permet d'effectuer des commandes dans le bit racine Set-Group-ID même utilisateur-ID, mais par rapport à la limitation du groupe de suppression du bit ou collant permet la suppression du fichier ou le répertoire de son seul propriétaire. Il s'agit d'un cas de répertoire /tmp: ls -ld /t/ drwxrwxrwt 2 racine racine 4096 Novembre. 28 13:17 tmp/t au lieu de x pour les autres utilisateurs nous informe que ce répertoire ne peut être supprimé par l'utilisateur racine Quant à d'autres autorisations, vous pouvez accumuler l'activation en ajoutant du code pour chacun, de sorte que pour activer le bit collant et GroupID sur votre script renomme_mes_photos.sh, vous atteignez un: chmod 3777 renomme_mes_photos.sh renomme_mes_photos.sh. modifier user linux. modifier group user linux. modifier home user linux. modifier nom user linux. modifier pdf sur linux. modifier uid user linux. modifier password user linux. modifier un pdf sur linux