Engineer

Système d'exploitation

Logiciel qui gère le matériel pour les autres programmes

Ce que c’est

Un système d’exploitation est le logiciel qui démarre en premier sur un ordinateur et reste actif sous tout le reste. Il répartit le processeur, la mémoire, le stockage et le réseau entre les programmes et leur donne à tous les mêmes moyens de lire des fichiers, d’utiliser des périphériques et d’accéder au réseau. Son abréviation courante est OS. Windows, macOS, Android, iOS et les distributions Linux comme Debian et Ubuntu sont des systèmes d’exploitation.

Le glossaire du NIST contient deux descriptions complémentaires. La première en fait l’application de contrôle principale qui fait fonctionner l’ordinateur : « le premier programme chargé lorsque l’ordinateur est allumé », dont le composant principal, le noyau, reste en mémoire en permanence, et qui « fixe les normes pour tous les programmes d’application », comme le serveur web. La seconde, tirée du guide du NIST sur la sécurité des technologies opérationnelles, le définit comme « un ensemble intégré de routines de service supervisant l’enchaînement des programmes ». Cet ensemble peut assurer le contrôle des entrées et sorties, l’ordonnancement des ressources et la gestion des données, et fournit aux applications les commandes fondamentales pour contrôler l’ordinateur.

Le noyau en est le cœur, mais un système d’exploitation ne se limite pas à son noyau. Autour se trouvent les bibliothèques système qu’appellent les programmes, le shell et les outils courants en ligne de commande, un superviseur qui lance les services au démarrage et les redémarre, un gestionnaire de paquets qui installe et met à jour les logiciels, et une vaste configuration par défaut qui décide de ce qui est activé, de la destination des journaux et des personnes autorisées à se connecter. Une distribution Linux assemble tout cela autour du noyau Linux : deux systèmes partageant le même noyau peuvent donc se comporter très différemment.

L’essentiel de ce qu’un système d’exploitation offre à un programme tient dans quelques abstractions qui représentent le matériel. POSIX, la norme sur laquelle reposent les systèmes de type Unix, les définit précisément. Un processus est un espace d’adressage où s’exécutent un ou plusieurs fils d’exécution, avec les ressources système dont ces fils ont besoin. Un fichier est un objet dans lequel on peut écrire, que l’on peut lire, ou les deux, avec des droits d’accès et un type : fichier ordinaire, répertoire, lien symbolique, FIFO, socket, fichier spécial en mode caractère ou bloc représentant un périphérique. Un fichier spécial en mode bloc donne accès à un périphérique en masquant ses caractéristiques matérielles. Un périphérique lui-même est défini comme « un périphérique informatique ou un objet qui apparaît comme tel à l’application ». L’objectif déclaré de POSIX est de rendre les applications portables au niveau du code source, afin qu’un programme écrit pour ces interfaces puisse être compilé et exécuté sur un autre système conforme.

La mémoire reçoit le même traitement. La documentation du noyau Linux part du fait que la mémoire physique est une ressource limitée et décrit la mémoire virtuelle comme la couche qui en masque les détails aux applications. Chaque programme voit son propre espace d’adressage. Le système d’exploitation décide à chaque instant quelles parties de quels programmes se trouvent dans la mémoire réelle et lesquelles sont compressées, écrites sur disque ou retirées de la mémoire pour être relues plus tard.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce qu'un système d'exploitation ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. L'entretien quotidien des serveurs, de la messagerie, des comptes et des mises à jour, pour que les outils indispensables restent sans histoire.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. 10 serveurs PostgreSQL et MS SQL sur Ubuntu — 10 serveurs de bases de données déployés et maintenus

10 serveurs PostgreSQL et MS SQL sur Ubuntu Contactez-nous →

Pourquoi c’est important

Choisir un système d’exploitation règle beaucoup de choses à la fois : les logiciels qui fonctionneront, les outils que les exploitants connaissent déjà, le fournisseur des correctifs de sécurité et la durée de leur disponibilité, ainsi que le prix d’un éventuel contrat d’assistance. Changer ensuite implique de transposer tous les programmes et toutes les habitudes qui reposent dessus. Le choix survit donc généralement au matériel sur lequel il a été fait.

C’est aussi la principale frontière de confiance à l’intérieur d’une machine. Utilisateurs, groupes, droits sur les fichiers et séparation entre processus sont des fonctions du système d’exploitation. Ils déterminent jusqu’où une faille dans un programme peut se propager. Une machine où tous les services s’exécutent avec tous les privilèges n’a plus qu’un rempart : la périphérie du réseau.

Ses valeurs par défaut sont les décisions de quelqu’un d’autre. Une nouvelle installation comprend des services activés, des fichiers à des emplacements convenus et des réglages déjà choisis. La plupart sont raisonnables ; certains ne conviendront pas au travail, et personne ne dit lesquels. Bien exploiter une machine suppose de savoir quelles valeurs par défaut vous avez acceptées.

Le système décide aussi jusqu’où le code peut voyager. Un programme qui respecte les interfaces portables passe sur un autre système après reconstruction. Un programme ou un script qui dépend des extensions d’un système ne va pas plus loin que ces extensions.

À quoi cela ressemble en pratique

Sur un ordinateur portable, les rapports de diagnostic d’une application macOS ont enregistré 111 Go de pages compressées sur une machine de 36 Go le 2026-09-01. Un échantillon distinct de l’empreinte mémoire a relevé une croissance d’environ 41 Mo par seconde pendant une boucle de tentatives de reconnexion qui émettait des événements du cœur de l’application. Ce sont des observations distinctes, pas une mesure continue unique. Le chemin de traitement des événements a été modifié pour borner les tampons, filtrer les abonnements par type d’événement et fixer un budget de débit à la journalisation. Une comparaison d’exécutions complètes des tests a enregistré une baisse des lignes de journal du cœur de 610 996 à 1 411. Les tests protègent cette conception bornée, mais cette comparaison n’est ni un benchmark en production ni une mesure de la mémoire après correction. Des délais d’expiration distincts à l’arrêt ont nécessité leur propre correction.

Sur l’hôte de l’entreprise, un rôle d’automatisation réécrivait ses paramètres SSH à chaque exécution depuis des mois lorsqu’un rapport d’accès a lu les paramètres effectivement résolus par le démon et découvert que deux ne correspondaient pas aux paramètres attendus. Le récit du durcissement SSH avec validation en trois étapes en donne la cause : le système d’exploitation fournit ses propres valeurs par défaut sans les mettre en commentaire, avant les paramètres du fichier de configuration complémentaire, et la première occurrence l’emporte

pour ces deux paramètres. La correction a été un préfixe de nom de fichier qui se classe avant celui du fournisseur. Désormais, 24 directives sont vérifiées par assertions contre ce que rapporte le démon.

Le même hôte a rencontré un autre choix du système lors de la mise en place de son service de bannissement des intrusions. La source des journaux est réglée sur detect, car le système d'exploitation ne fournit pas de journal d'authentification traditionnel. Un mauvais choix échoue silencieusement en surveillant un fichier qui n'apparaît jamais.

Où les problèmes surviennent

Un script transporte les habitudes de son système d'origine. Un script shell écrit sur un Mac puis exécuté sur un serveur Linux, ou l'inverse, retrouve les mêmes noms de commandes avec des options différentes. Le `sed` de POSIX accepte `-E` et `-n` ; modifier un fichier sur place avec `-i` est une extension que les deux versions courantes interprètent différemment :

```
sed -i 's/old/new/' notes.txt     # GNU sed, on most Linux systems
sed -i '' 's/old/new/' notes.txt  # BSD sed, on macOS
```

Chaque forme échoue sur l'autre système. Tenez-vous à ce que définit POSIX lorsque cela suffit, exécutez le script sur le système qui le fera tourner en production et installez le même outil des deux côtés lorsqu'une extension vaut la peine.

Deux machines avec le même système d'exploitation diffèrent. « Debian » désigne une famille de versions. Deux hôtes décrits comme l'utilisant peuvent avoir des versions de distribution, de noyau et de paquets séparées de plusieurs années. Les paquets installés et les modifications locales peuvent aussi différer. Lisez ce qu'est réellement chaque machine avant de les comparer :

```
cat /etc/os-release
uname -srm
```

La première commande affiche le nom et la version de la distribution ; la seconde, le nom et la version du noyau, ainsi que le type de processeur de la machine. Construisez chaque hôte à partir de la même description écrite et conservez la version dans l'inventaire.

Les services s'exécutent en tant qu'administrateur. Sur un système de type Unix, un service exécuté en tant que root peut lire tous les fichiers et modifier tous les réglages. Toute faille de ce service devient donc une faille de la machine entière. Donnez à chaque service son propre utilisateur non privilégié, avec un accès en écriture uniquement là où il écrit, puis accordez les droits supplémentaires un par un lorsque le service en montre le besoin.

Une version reste utilisée après la fin de son suivi de sécurité. Chaque version cesse de recevoir des correctifs selon un calendrier publié. Par exemple, l'équipe de sécurité de Debian prend en charge chaque distribution stable pendant trois ans à compter de sa publication. Un hôte laissé sur une ancienne version continue de fonctionner : c'est précisément pourquoi on l'oublie. Notez la date de fin de prise en charge de chaque hôte à côté de ses autres échéances de renouvellement et prévoyez la mise à niveau bien avant.

Pour aller plus loin

- Système d'exploitation — glossaire NIST CSRC — les deux définitions, de l'application de contrôle principale à l'ensemble de routines de service.
- POSIX.1-2024, définitions de base : définitions — la formulation exacte de processus, fichier, périphérique et des autres notions qui constituent un système d'exploitation.
- POSIX.1-2024 : sed — les options acceptées par tout `sed` conforme, pour délimiter ce qu'un script portable peut utiliser.
- Documentation du noyau Linux : vue d'ensemble des concepts — mémoire virtuelle, cache de pages et récupération de mémoire, décrits par le noyau lui-même.
- FAQ de sécurité Debian — comment Debian traite les correctifs de sécurité et combien de temps une version stable les reçoit.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le parcours de déploiement, les risques opérationnels et les personnes d'astreinte. Nous pouvons définir un changement avec des vérifications, une procédure de reprise et une responsabilité claire.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce qu'un système d'exploitation ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/operating-system/