Qubes, un système d'exploitation pas comme les autres

22.5.15
brève, innovation, marché, tool

A l'occasion de la sortie de la version 3.0 RC1, revenons sur le fonctionnement du système d'exploitation Qubes, un système à multi-niveaux de sécurité.

Présentation et fonctionnement de Qubes

Philosophie

Le système d'exploitation Qubes promet un niveau de sécurité élevé via une approche par compartimentalisation. Cela signifie que différents environnements, de niveaux de sécurité différents, vont pouvoir cohabiter au sein d'un même ordinateur physique, en étant isolés. Ainsi, la compromission d'un des environnements ne permettra pas d'accéder aux autres environnements.

Pourquoi cette approche de cloisonnement ?

Cette volonté de séparer les différents environnements découle simplement du constat d'échec de la sécurité logicielle actuelle. Historiquement, trois grandes approches de sécurité se sont

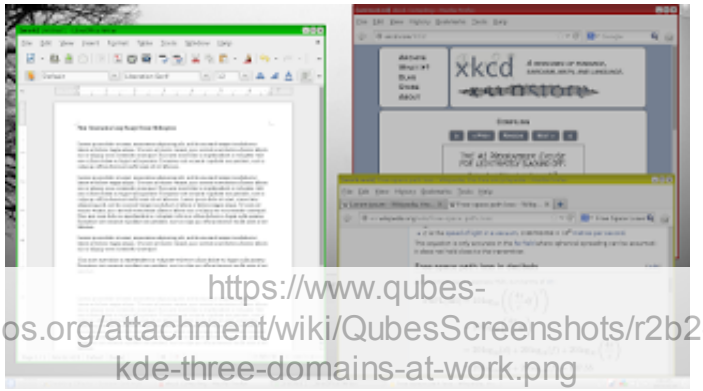
affrontées :

- la sécurité par l'obscurité ;
- la sécurité basée sur la qualité du code ;
- la sécurité par l'isolation.

Globalement, la sécurité par l'obscurité n'est pas efficace, et l'approche par une qualité de l'ensemble des développements applicatifs est bien trop coûteuse et difficile à assurer au regard de la richesse et de la complexité des logiciels que l'on retrouve de nos jours sur un poste de travail.
La seule approche efficace sur le terrain et permettant d'assurer un bon niveau de sécurité sans pour autant rompre la compatibilité avec l'existant est celle par isolation. La compromission d'un composant étant inévitable, il s'agit d'en limiter l'impact. Plutôt que de tenter en vain de sécuriser l'ensemble des composants logiciels de l'ordinateur, chacun d'entre eux va être placé dans un compartiment, isolé des autres, et ne pourra pas les impacter.

Un cloisonnement -presque- transparent pour l'utilisateur

Un des principaux atouts de Qubes est sa capacité à proposer à l'utilisateur un environnement unifié, qui gomme le cloisonnement technique sous-jacent. En effet, l'utilisateur a la possibilité d'exécuter des applications dans des environnements cloisonnés, de niveaux de sécurité hétérogènes, mais qui sont pourtant affichés dans une unique interface graphique. Chaque fenêtre peut appartenir à un environnement différent, repéré par la couleur de la bordure de la fenêtre et un préfixe dans le titre de celle-ci. L'utilisateur peut ainsi facilement identifier avec quel environnement il interagit.

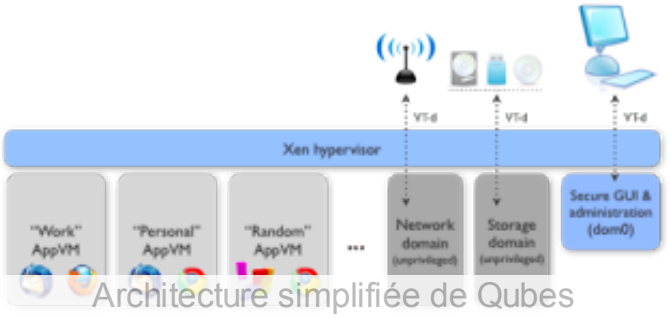


https://www.qubes-os.org/attachment/wiki/QubesScreenshots/r2b2-kde-three-domains-at-work.png

Architecture système de Qubes

Techniquement, Qubes repose sur l'emploi de la virtualisation pour séparer ces différents environnements. C'est l'hyperviseur Xen qui

est employé, en utilisant des machines virtuelles pour les différents environnements ainsi que pour la plupart des composants système.
L'architecture système est représentée de manière simplifiée dans le schéma ci-dessous :



Architecture simplifiée de Qubes

Le domaine d'administration dom0

Le domaine dom0 est la machine virtuelle qui dispose des privilèges les plus élevés sur le système. Il est en charge de gérer l'interface graphique, ainsi que les périphériques d'entrée tels que le clavier et la souris. C'est via ce domaine que se gèrent les machines virtuelles : création, mise à jour, suppression, ainsi que les réglages systèmes de Qubes.

Les domaines "techniques"

Des machines virtuelles spécifiques sont utilisées pour gérer les accès au stockage et au réseau, via la virtualisation d'entrées/sorties IOMMU. Il est donc nécessaire d'avoir un processeur compatible, comme les processeurs Intel disposant de la technologie VT-d.

On peut notamment citer les machines virtuelles "techniques" suivantes :

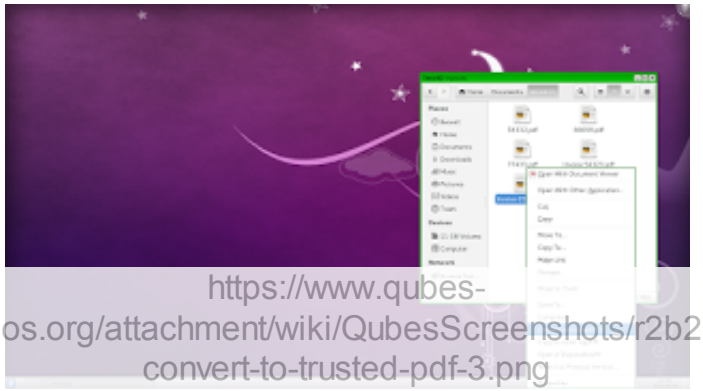
- netVM : en charge de la gestion du réseau.
- FirewallVM : permet de filtrer les accès réseau des différents environnements.
- ProxyVM/torVM : machine dédiée à la tunnelisation des flux via un proxy ou via TOR.

Les machines virtuelles et "AppVM"

Les machines virtuelles sous Qubes peuvent se gérer par l'utilisation de "templateVM", c'est-à-dire de modèle de machines virtuelles. La force de ce modèle est de pouvoir cloisonner des environnements qui reposent sur une même "souche" système, accessible à chacun d'entre eux en lecture seule. Ainsi, l'espace

disque nécessaire est limité et n'est pas multiplié à chaque ajout de machine virtuelle. De plus, des mécanismes spécifiques permettent la mise à jour de la souche ; ainsi, l'ensemble des environnements dépendant de cette souche bénéficient des correctifs de sécurité qui ont été appliqués.

Qubes a également poussé encore plus loin le concept de cloisonnement, en créant des machines virtuelles "jetables" (disposableVM). On peut ainsi utiliser un navigateur dans une machine virtuelle créée pour l'occasion, et qui sera détruite lorsqu'on fermera le navigateur. Ainsi, même en cas de compromission, l'attaquant n'aura pu récupérer, au mieux, que l'ensemble des données des différents onglets de cette instance du navigateur, et ne pourra jamais obtenir de persistance sur le système. Ce mécanisme peut également être utilisé pour consulter des documents d'origine douteuse, comme des fichiers PDF téléchargés depuis Internet ou reçus par email. L'intégration de cette fonctionnalité est faite par l'intermédiaire des "Qubes tools", qui permettent d'avoir un menu contextuel dans les machines Linux et Windows permettant d'ouvrir un fichier dans un environnement jetable.



https://www.qubes-os.org/attachment/wiki/QubesScreenshots/r2b2-convert-to-trusted-pdf-3.png

Quelles avancées à venir pour Qubes 3 et + ?

Les limitations actuelles de Qubes

Bien que séduisante, la solution Qubes n'est pas pour autant exempte de défauts. Le premier est la compatibilité matérielle, un peu limitée. Une base centrale répertorie les matériels capables de faire tourner Qubes. Bien qu'aucun élément spécifique ne soit nécessaire, le bon fonctionnement de Qubes est évidemment lié à la disponibilité des drivers adéquats. Notons toutefois que les ordinateurs portables Lenovo sont pour la plupart compatibles avec Qubes.

Par ailleurs, bien que Qubes soit un système open-source, ce n'est pas le cas de l'intégralité des composants. Les "Qubes tools" qui permettent la bonne intégration des différents domaines, dont le transfert de fichiers entre machines virtuelles, ne sont distribués que sous format binaire. Étant donné que ces outils doivent s'exécuter dans chacune des machines virtuelles, ils ont par conception un accès privilégié à l'ensemble des données d'un poste de travail sous Qubes.

Enfin, dans l'optique d'un déploiement en entreprise, il faut noter qu'actuellement, aucun outil de gestion à distance n'est disponible. La sécurité repose donc également sur la bonne volonté de chaque utilisateur, qui pourrait (volontairement ou non) dégrader le niveau de sécurité de son installation.

Une roadmap publiée

Lors de la sortie de la version Release Candidate 1 de Qubes 3.0, Joanna Rutkowska, créatrice du projet Qubes, a publié une roadmap officielle pour Qubes 3 sur le blog de sa société Invisible Things Lab.

http://blog.invisiblethings.org/resources/roadmap-2015-apr.png

Dès la prochaine version, 3.1 prévue avant l'automne, des fonctionnalité d'administration seront introduites, et serviront de base à une future édition "entreprise" de Qubes. C'est sans doute la fonctionnalité la plus attendue.

S'ajouteront à ces fonctionnalités de nombreuses améliorations au niveau sécurité, comme la publication de binaires déterministes (c'est-à-dire qu'un même code source produit un même binaire, permettant d'améliorer la confiance dans les binaires distribués par Qubes).
Enfin, les amateurs de jeu ou de cassage de mot de passe peuvent se réjouir puisque la technologie "GPU-passthru" devrait

permettre, dans la version 4.1 de Qubes, de profiter pleinement de la puissance des cartes graphiques depuis une machine virtuelle !

Concurrents

Il convient en conclusion de préciser que Qubes OS n'est pas le seul système d'exploitation sécurisé via la compartimentalisation.

Parmi les solutions françaises, CLIP, solution développée en interne par l'ANSSI, et présentée récemment à la JSSI, permet également un cloisonnement, mais limité à deux niveaux (haut/bas). Le système PolyXen, développé par Bertin Technologies, et présenté il y a peu lors d'une réunion de l'OSSIR, pousse encore la sécurité plus loin puisqu'il permet d'exécuter plusieurs instances de l'hyperviseur Xen, cloisonnées entre-elles par un micro-kernel propriétaire. Les gouvernements allemand et américain ont également adopté des solutions similaires, respectivement SINA et NetOP.

Pour se jeter à l'eau, la version 3.0 RC1 de Qubes est disponible ici, pensez à vérifier la signature après téléchargement !
Arnaud SOULLIE