Fonctionnement du mécanisme d'arrêt d'urgence de noyau

Ce mécanisme d'arrêt d'urgence repose sur un principe simple : offrir aux administrateurs disposant des droits requis la possibilité de neutraliser temporairement, via l'interface securityfs, une fonction du noyau jugée vulnérable. Lorsqu'un chemin de code dangereux est ciblé par l'arrêt d'urgence, la fonction cesse de s'exécuter normalement et renvoie simplement une erreur, bloquant ainsi l'exploitation de la vulnérabilité par les acteurs de la menace.

```
From: Sasha Levin <sashal@kernel.org>
To: corbet@lwn.net, akpm@linux-foundation.org
Cc: skhan@linuxfoundation.org, linux-doc@vger.kernel.org,
        linux-kernel@vger.kernel.org, linux-kselftest@vger.kernel.org,
        gregkh@linuxfoundation.org, Sasha Levin <sashal@kernel.org>
Subject: [PATCH] killswitch: add per-function short-circuit mitigation primitive
Date: Thu,  7 May 2026 03:05:45 -0400   [thread overview]
Message-ID: <20260507070547.2268452-1-sashal@kernel.org> (raw)

When a (security) issue goes public, fleets stay exposed until a patched kernel
is built, distributed, and rebooted into.

For many such issues the simplest mitigation is to stop calling the buggy
function. Killswitch provides that. An admin writes:

    echo "engage af_alg_sendmsg -1" \
        > /sys/kernel/security/killswitch/control

After this, af_alg_sendmsg() returns -EPERM on every call without
running its body. The mitigation takes effect immediately, and is dropped on
the next reboot.

A lot of recent kernel issues sit in code paths most installs only have enabled
to support a relative minority of users: AF_ALG, ksmbd, nf_tables, vsock, ax25,
and friends.

For most users, the cost of "this socket family stops working for the day" is
much smaller than the cost of running a known vulnerable kernel until the fix
land.

Assisted-by: Claude:claude-opus-4-7
Signed-off-by: Sasha Levin <sashal@kernel.org>
```

Le noyau Linux est constitué de milliers de petites fonctions, chacune chargée d'une tâche spécifique, comme le traitement d'un paquet réseau ou la communication avec un périphérique USB. Lorsqu'une faille apparaît dans l'une de ces fonctions, la solution consiste à corriger le code et à publier un nouveau noyau. Le kill switch implique une approche plus radicale, où l'administrateur fournit au noyau Linux un nom de fonction et une valeur de retour.

À partir de ce moment, la fonction continue d'être appelée par ce qui l'appelait auparavant, mais elle se contente de renvoyer cette valeur et de se fermer. Le code qu'elle contient ne s'exécute jamais. Selon la documentation fournie par Sasha Levin, ce mécanisme cible principalement des chemins de code sur lesquels la plupart des systèmes ne s'appuient pas quotidiennement, comme les fonctions AF_ALG, ksmbd, nf_tables, vsock et ax25.

Concrètement, cela se traduit par une seule ligne dans le terminal, comme le montre l'image ci-dessous. À partir de là, tout programme tentant d'envoyer des données via AF_ALG (l'interface de cryptographie du noyau également exploitée par la vulnérabilité critique Copy Fail) renvoie une erreur. Quelle que soit la vulnérabilité présente dans la fonction af_alg_sendmsg, elle est désormais inaccessible, car cette fonction ne s'exécute plus jamais.

L'effet s'applique immédiatement à tous les cœurs du processeur et persiste jusqu'à ce que l'administrateur le désactive ou que le système redémarre. Toute activation nécessite des privilèges root. Par ailleurs, la proposition contient également un paramètre de démarrage `killswitch=fn1=val,fn2=val,...`, destiné aux cas où un opérateur doit appliquer la mesure de mitigation à l'ensemble d'un parc de machines via le chargeur d'amorçage.

Risques et limites de ce mécanisme d'arrêt d'urgence

L'utilisation de cette fonctionnalité n'est pas sans danger et ne comporte aucune vérification de sécurité automatique pour déterminer si une fonction peut être bloquée sereinement. En cas de mauvaise manipulation, la désactivation d'une fonction inadéquate pourrait gravement perturber le comportement du système ou créer de nouvelles pannes. Sasha Levin note que ce mécanisme ne constitue en aucun cas un correctif définitif ou du "live patching".

```
echo "engage af_alg_sendmsg -1" \
        > /sys/kernel/security/killswitch/control
```

Ce mécanisme ne remplace pas non plus le code défectueux et requiert toujours une mise à jour complète du noyau pour résoudre la vulnérabilité de manière permanente. Pour le moment, cette rustine logicielle est toujours en phase d'évaluation et n'a pas encore été acceptée dans le noyau Linux. Son activation corrompt également le noyau, ce qui est sa façon de signaler que le code en cours d'exécution n'est plus le code Linux « en amont » pur.

Un drapeau (H, bit 20) est activé dès qu'un kill switch est actif, et il persiste même après sa désactivation, jusqu'au prochain redémarrage. Tout plantage survenant par la suite comporte un H dans son en-tête, ce qui sert de signal à l'administrateur, indiquant que l'image a été modifiée. Le correctif consacre également une section entière intitulée « Choisir la bonne cible », avertissant les opérateurs de ne pas sélectionner la mauvaise fonction.

Les réactions à cet outil sont partagées. Certains utilisateurs y voient un outil potentiellement utile pour pallier des dysfonctionnements matériels, comme le blocage d'un pilote. Mais d'autres expriment de sérieuses réserves, craignant que cet outil ne soit abusé par des acteurs malveillants, n'augmente la surface d'attaque globale, ou ne soit mal utilisé par les administrateurs au risque d'endommager involontairement leurs propres systèmes.

Un utilisateur a qualifié cela de « fonctionnalité de sécurité qui pourrait s'avérer pire que la faille elle-même ». « Pour être honnête, cette idée me semble augmenter la surface d'attaque et en faire une cible idéale pour un déni de service, mais je ne suis pas un expert en sécurité », a écrit un utilisateur sceptique.

L'IA a joué un rôle dans cette proposition de Sasha Levin

Il y a un autre point qui mérite d'être souligné. Au bas de la proposition de Sasha Levin figure la ligne « Assisted-by: Claude:claude-opus-4-7 ». Cette ligne indique qu'il s'agit d'un patch rédigé par Sasha Levin avec l'aide de l'assistant IA d'Anthropic. Ce n'est pas un cas isolé. La balise « Assisted-by » elle-même découle d'une nouvelle politique sur le développement du noyau Linux adoptée récemment. Sasha Levin a contribué à son élaboration.

Elle définit comment les contributions assistées par l'IA doivent être attribuées dans les messages de commit, ainsi que les fichiers de configuration pour des outils tels que Claude. Cela s'inscrit dans une tendance plus large. Greg Kroah-Hartman a utilisé son propre fuzzer IA sur le noyau, et les premiers bogues qu'il a détectés se trouvaient dans ksmbd, l'un des sous-systèmes que Sasha Levin a spécifiquement désignés comme candidat au « killswitch ».

Des vulnérabilités critiques identifiées dans le noyau Linux

Ces dernières semaines n'ont pas vraiment fait la promotion de l'approche traditionnelle consistant à « attendre les correctifs ». Nous avons d'abord assisté à la divulgation de CopyFail, une faille de Linux permettant l'escalade des privilèges en local, qui est rapidement passée de la divulgation à l'exploitation active. Quelques jours plus tard, Dirty Frag a fait son apparition : une autre vulnérabilité critique d'escalade des privilèges sous Linux.

Dirty Frag est accompagnée d'un code d'exploitation public et dépourvue de correctifs officiels, après que les efforts de divulgation coordonnée ont échoué avant que les correctifs ne soient prêts. Comme l'indique la proposition de Sasha Levin dans sa proposition, « les organisations restent souvent exposées jusqu'à ce qu'un noyau corrigé soit compilé, distribué et que le système soit redémarré ». Killswitch vise à combler cette lacune.

Naturellement, cette proposition a suscité un débat sur la stabilité, le risque d'abus et la question de savoir si l'on peut faire confiance aux utilisateurs pour ne pas couper accidentellement des membres importants en production. Mais après CopyFail et Dirty Frag, la communauté du noyau semble de plus en plus convaincue qu'il vaut désormais mieux exécuter des fonctionnalités défectueuses plutôt que des fonctionnalités transformées en armes.

Source : proposition de Sasha Levin

Et vous ?

- Quel est votre avis sur le sujet ?
- Que pensez-vous de la proposition visant à ajouter un kill switch au noyau Linux ?
- Cette proposition est-elle pertinente ? Ou augmente-t-elle la surface d'attaque comme certains le pensent ?

Voir aussi

- Copy Fail : cette vulnérabilité est considérée comme l'une des plus graves à toucher Linux depuis des années. Elle permet d'obtenir des privilèges root sur toutes les distributions disponibles depuis 2017.
- La version préliminaire de la distribution Linux Manjaro 26.1 « Bian-May » est disponible, avec le noyau Linux 7.0, GNOME 50, KDE Plasma 6.6, Xfce 4.20 et de nouveaux contrôles parentaux
- Quelqu'un a piraté sa PlayStation 5 pour y installer Ubuntu Linux 26.04 LTS et s'en est servi pour jouer à GTA V Enhanced sur Steam à une fluidité de 60 fps en 1440p. Vous pouvez désormais en faire autant

Le respect de votre vie privée est notre priorité

Nous et nos stockons et/ou accédons à des informations sur un appareil, telles que les cookies, et traitons des données personnelles telles que des identifiants uniques et des informations standards envoyées par un appareil pour des publicités et du contenu personnalisés, des mesures de publicité et de contenu, des études d'audience et le développement de services.Avec votre permission, nos 1722 partenaires et nous-mêmes pouvons utiliser des données de géolocalisation précises et d’identification par scan d'appareil. En cliquant, vous pouvez consentir aux traitements décrits précédemment. Vous pouvez également refuser de donner votre consentement ou accéder à des informations plus détaillées et modifier vos préférences avant de consentir. Veuillez noter que certains traitements de vos données personnelles peuvent ne pas nécessiter votre consentement, mais vous avez le droit de vous y opposer.Vos préférences s'appliqueront uniquement à ce site Web et seront stockées pendant 13 mois dans IABGPP_HDR_GppString cookie. Vous pouvez modifier vos préférences ou retirer votre consentement à tout moment en revenant sur ce site et en cliquant sur le bouton "Confidentialité" en bas de la page Web.

Veuillez noter que ce site Web/cette appli utilise un ou plusieurs services Google et peut recueillir et conserver des informations, y compris, mais sans s’y limiter, sur votre comportement en matière de visite ou d’utilisation. Vous pouvez cliquer pour accorder ou refuser votre consentement à ce que Google et ses balises tierces utilisent vos données aux fins indiquées ci-dessous dans la rubrique de consentement de Google.