Fork Bomb

Une attaque informatique qui sature les ressources d'un système en créant une prolifération exponentielle de processus jusqu'à son blocage complet.



Qu'est-ce qu'une Fork Bomb ?

La fork bomb est un type d'attaque visant à rendre un système inutilisable en saturant ses ressources. Elle repose sur la création récursive de processus, chaque processus engendrant plusieurs copies de lui-même à vitesse exponentielle.

Bien que simple à concevoir, elle peut avoir des conséquences graves sur les serveurs ou ordinateurs personnels sans protection adéquate.





Principe de Fonctionnement

1

Création de processus

Exploitation de la fonction "fork" pour dupliquer un processus en cours.

2

Prolifération exponentielle

Chaque nouveau processus engendre plusieurs copies, augmentant rapidement le nombre total.

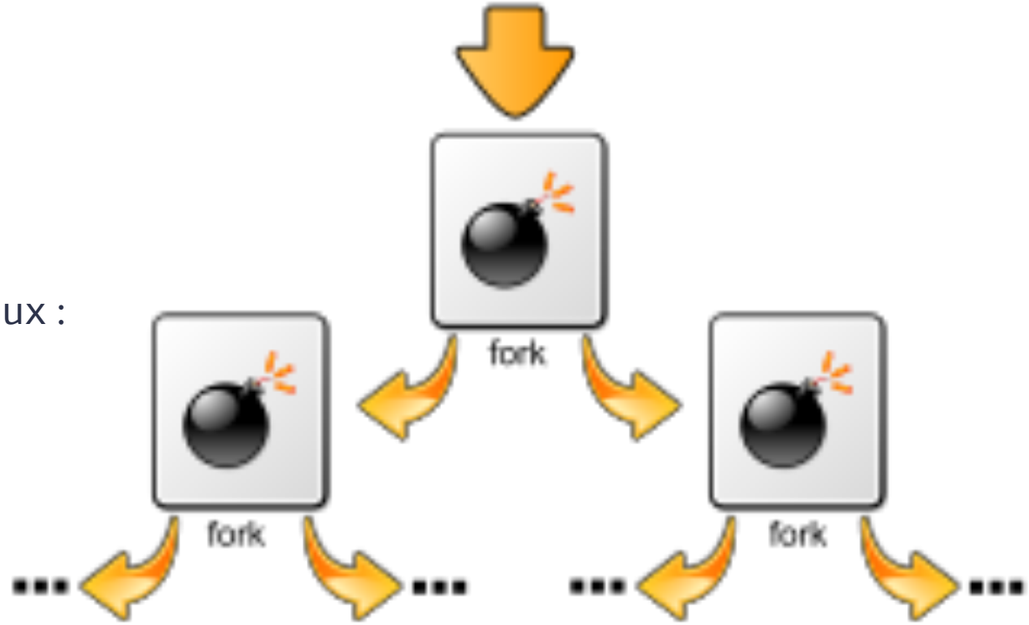
3

Saturation des ressources

Le système atteint sa limite de capacité, entraînant blocage ou arrêt forcé.

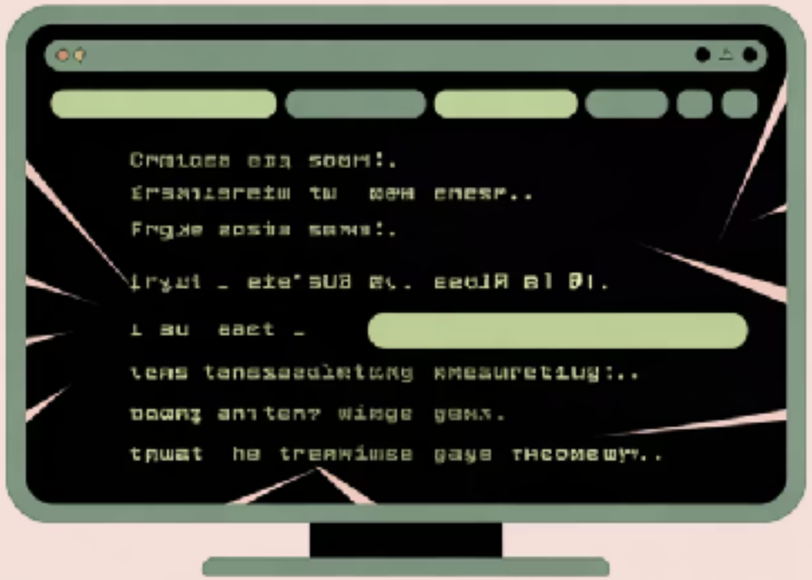
Exemple Classique

Voici un exemple de fork bomb écrit en langage C pour les systèmes Unix/Linux :




```c
#include <unistd.h>

int main(void)
{
  while(1) {
    /* ici on peut ajouter un malloc pour utiliser plus de ressources */
    fork();
  }
  return 0;
}
```



Impacts Techniques

Blocage des systèmes

Saturation de la mémoire et des processeurs empêchant l'exécution de tâches normales.

Redémarrage nécessaire

Un redémarrage complet du système est souvent requis pour rétablir le fonctionnement.

Perturbations serveurs

En environnement multi-utilisateurs, affecte plusieurs clients ou services critiques simultanément.

Les fork bombs sont particulièrement efficaces sur :

- Les systèmes sans restrictions de processus où aucun mécanisme de limitation n'est en place
- Les serveurs partagés où les attaques peuvent perturber plusieurs utilisateurs à la fois



Prévention : Limitation des Ressources

01

Limitation des processus par utilisateur

Les systèmes Unix/Linux permettent de limiter le nombre de processus via la commande ulimit.

```
ulimit -u 100
```

Limite chaque utilisateur à 100 processus simultanés maximum.

02

Groupes de contrôle (cgroups)

Les cgroups peuvent restreindre l'utilisation des ressources comme le CPU et la mémoire pour chaque groupe d'utilisateurs.

Surveillance et Bonnes Pratiques

Détection de comportements anormaux

Identifier les pics soudains dans le nombre de processus via des systèmes de surveillance.

Alertes automatisées

Configurer des alertes et scripts pour tuer automatiquement les processus suspects.

Éducation des utilisateurs

Sensibiliser les utilisateurs aux risques liés aux fork bombs et aux bonnes pratiques.



Utilisations éthiques

Tests de résilience

Les fork bombs peuvent être utilisées de manière contrôlée pour tester la robustesse des systèmes et s'assurer qu'ils résistent à des scénarios de saturation.



Éducation

Dans les programmes de formation en cybersécurité, les fork bombs sont présentées comme un exemple simple mais puissant des attaques de déni de service.



Travail demandé

- Dans un terminal lancer la commande « top » pour voir les processus en cours
- Dans un autre terminal, écrire un programme de « fork bomb », le compiler et l'exécuter
- Observer le comportement des processus
- Que faire ensuite ?





Conclusion

La fork bomb est une démonstration classique de l'impact des attaques de saturation des ressources. Bien qu'utilisée à des fins pédagogiques ou de test, ses effets peuvent être dévastateurs en environnement réel.

Gestion rigoureuse

Une gestion des ressources est essentielle pour minimiser les risques.

Sensibilisation

L'éducation des utilisateurs reste la clé de la prévention.

Protection active

Des mesures de prévention adéquates doivent être en place.