CSPN security target

nftables

Version 1.3

Ref.: **CSPN_TARGET_NFTABLES_1.3**

Identification of the document

Document specifications

Object	CSPN security target - nftables
Number of pages	11
Diffusion	Public

Document history

Version	Date	State
1.0	2023/02/28	First version
1.1	2023/03/14	Revision after proofreading and remarks
1.2	13/12/2023	Revision of the version of the product to match what was audited.
1.3	25/04/2024	Modification on ANSSI's request

Table of contents

1. Introduction

1.1. Product identification

Company Editor	The netfilter.org project
Product Name	Linux Kernal subsystem nftables
Evaluated version	Nftables on Debian 12.1 / Kernel 6.1.55-1
Potential fixes applied	**CVE-2023-5197**
Product category	Firewall

1.2. Document structure

This document is divided up in 5 parts (this introduction excluded) describing:

- the target of evaluation (TOE);
- the environment in which the product is evaluated;
- the sensitive assets that are to be protected by the product;
- the threat and the threat actors;
- the security features handling the threats previously described.

1.3. References

Identifier	Title	Reference	Version	Classification
CSPN	Certification de Sécurité du Premier Niveau	ANSSI-CSPN-CER-P-01_v4.0	4.0	Public
CRYPTO_GUIDE	Guide for the cryptographic mechanisms.	ANSSI-PG-083	2.04	Public

2. Product description

2.1. General description

nftables is a subsystem of the Linux kernel providing filtering and classification of network packets, datagrams and frames.

It is part of the netfilter project, which is a community-driven collaborative FOSS project that provides packet filtering software. *nftables* is the successor of *iptables* and allows for much more flexible, scalable and performance packet classification.

The netfilter project recommends the usage of *nftables* instead of the legacy version *iptables*.

nftables is a “stateful” firewall and supports advanced features of protection, filtering and classification of network packets. A detailed guide on how to use *nftables* is available on the official website.

nftables is available on all Linux kernels since version **3.13**.

2.2. Product usage

As a subsystem of the Linux kernel, *nftables* exists as a kernel module named *nf_tables.* It can be loaded during the boot of the host system or when a user tries to interact with it.

Communication between the user space and *nftables* in kernel happens via *netlink*, a packet-based IPC mechanism for that purpose. *netlink* is a kernel interface and protocol that is used to communicate networking information from usermode to kernel and vice versa. Every kernel subsystem that wants to communicate using *netlink* defines an associated socket family that the user has to specify. To communicate with *nftables* and *netfilter* in general, one can use the **NETLINK_NETFILTER** family.

nft is the default command line tool in order to interact with *nftables* at userspace. It is the main tool used by administrator to configure *nftables*. Other tools exist that can directly interact with the *netlink* socket so create complex rules, such as the *nftables* tool developed by Google.

2.3. Environment description

nftables can be used in a great variety of configurations, from a personal use to the complex information network of great companies. The TOE will be evaluated as a firewall deployed as a network bridge, used to control the inbound and outbound packets between several networks, by applying the rules defined by the administrator. This configuration also allows the TOE to be evaluated as a local firewall, filtering the input and output packets of the host.

For the evaluation, the host system will be configured with the property *kernel.unprivileged_userns_clone* set to **1**.

This configuration allows any unprivileged user to create arbitrary Linux namespaces. An unprivileged user can use this feature to create a process running under two new namespaces of kinds *pid* and *net*. Such a process would be able to operate under the *uid* 0 in the given namespace, and therefore to communicate with *nftables* using a *netlink* socket. One should note that such a process should not be able to view or modify a configuration defined in another namespace.

2.4. Description of dependencies

nftables is used in a great variation of tools to ease its use, but only the features provided by the *nftables* Linux’s subsystem will be considered for the evaluation.

However*, nftables* relies on several components of the netfilter project to be able to filter, route and log packets. For example, the support of NAT is done by the kernel module *nf_nat*, which itself relies on the module *nf_conntrack* to be able to track connections before making decisions on a packet.

A list of dependencies considered in the scope of the evaluation can be found in section Evaluation scope page 6.

2.5. Description of users and roles

2.5.1. Administrators

Typical users of *nftables* are the system administrators responsible for the installation, configuration and maintenance of *nftables* and the host system in general. To be able to communicate with *nftables*, these users must have the most privileged accounts of the system (*root* user or equivalent). These users will be called administrators in the rest of this document.

2.5.2. Unprivileged users

Considering the system configuration described in section Environment description page 5, any unprivileged user of the system is able to communicate with *nftables,* by creating its own *pid* and *net* namespaces. This kind of users will be called unprivileged users in the rest of this document.

2.5.3. Remote users

Finally, any user that can interact with a network interface where *nftables* is used to route, log, or filter packets will be called remote user in the rest of this document.

2.6. Hypothesis on the environment

H1. Administrators

- It is supposed that administrators (as described in section Description of users and roles page 6) are competent and non hostile. They are trained to correctly configure and administrate the host system and follow the recommended guidelines and good practices.

H2. Physical access

- It is supposed that the hardware running the TOE is located in a secure environment the access to which is restricted to trusted users considered non hostile.

H3. Host system

- The operating system hosting *nftables* is correctly administrated and configured. It is considered sound and the system libraries and kernel behave as expected. The presence of potential unprivileged and malicious users on the system have no impact on the behavior of the operating system.

2.7. Evaluation scope

The assessment focuses on every security feature described in section Security features page 11.

The TOE is considered as an attack surface for unprivileged users trying to obtain a privilege escalation using *nftables* features. This includes potential attacks that requires the presence or legitimate actions of an administrator.

Like a lot of modern firewalls, *nftables* support features that can protect a system or a network from denial of service attacks. For example, the *nftables* supports *syncookies* and *synproxy* to protect against SYN flood attacks. However, to be able to realize the assessment in time, the ability of *nftables* to protect itself or another host from a denial of service attack is not considered.

Concerning the modules considered in the scope of the evaluation, it is complex to build an exhaustive list of every kernel module that *nftables* can rely on to apply the filtering rules. Generally, modules of the Netfilter project that are used by a main feature of *nftables* to filter, route or log packets are considered in the scope of the evaluation. Specifically, the following modules are considered in the scope of the evaluation:

- nf_tables
- nf_conntrack
- nf_defrag_ipv6
- nf_defrag_ipv4
- nf_nat
- nft_chain_nat
- nft_log

The following modules are **not** considered in the scope of the evaluation:

- nft_compat
- nft_synproxy

Finally, the network interfaces protected by *nftables* can be configured both in IPv4 or IPv6.

3. Description of the technical environment for the product to operate

3.1. Hardware prerequisites

nftables must be installed on a computer running on the architecture AMD64.

3.2. System prerequisites

The operating system chosen for the evaluation is the last stable version of Debian 12 bookworm. *nftables* is the default framework in use in Debian and should be installed using the operating system packages to get the last version available.

The system must be configured with the system property *kernel.unprivileged_userns_clone* to **1**:

```
sudo sysctl -w kernel.unprivileged_userns_clone=1
```

4. Sensitive assets

The sensitive assets protected by the TOE are:

A1: Integrity and conformity of the filtering core

The TOE must correctly apply the filtering rules specified by the administrator, correctly route the legitimate packets, and block the forbidden ones.

The TOE must apply the filtering rules while preserving the integrity:

- of the filtering core
- of the flow matrix; for a given flow, the core must preserve the integrity:
  - of the source
  - of the destination
  - of the protocol

A2: Confidentiality and integrity of the filtering rules

The filtering rules defined by the administrators must be protected in confidentiality and integrity from unprivileged users or remote users. In case of networking address translation, the network address mapping must be protected in confidentiality and integrity by the TOE.

A3: Conformity and integrity of the logging

An attacker must not be able to alter the logs or the mechanism allowing to log packets.

4.1. Security properties

Identifier	Confidentiality	Integrity	Authenticity
A1		X	
A2	X	X	
A3		X	

5. Threats

5.1. Threatening agents

The threatening agents are unprivileged users present on the host system, and any user having access to a network interface on the host running *nftables.*

5.2. Threats

T1: Bypass

An attacker successfully bypasses a filtering rule defined by an administrator and manages to route a packet that should have been blocked by *nftables*. More broadly, an attacker manages to bypass any rule (filtering, routing or logging) defined by the administrator or to route a packet to a destination that was not intended.

T2: Information leak

An attacker manages to leak information on the state of *nftables*, or on rules specified by the administrators.

T3: Take over

An attacker manage to exploit a vulnerability in *nftables* to execute code on the host, elevate its privileges, or modify the behavior of *nftables* that is expected as per the administrator's configuration.

T4: Log corruption

An attacker manages to corrupt the logs kept by *nftables*.

5.3. Threats on Sensitive Assets

	A1	A2	A3
T1	X		X
T2		X	
T3	X	X	
T4			X

6. Security features

6.1. List of security features

SF1: Application of filtering rules

nftables allows filtering the packets transiting on an interface based on a subset of predefined rules.

The filtering can be made on numerous criteria pertaining to a packet that can be contextual (being part of an already created session, number of established sessions…) or non-contextual (the source or destination IP address or port, the destination or source interface, the content of the packet, etc.)

The filtering engine of *nftables* is very flexible and allows applying numerous actions and transformations on a packet. Without being exhaustive, the rules allow at least :

- to apply actions on a packet: block it, drop it, let it pass or apply different rules on it
- to modify the routing of the packet (NAT, redirection, reassembly, forward on another interface, etc.)
- to follow the connection state and take decisions depending on it
- to modify the content of the packet.

nftables is able to reassemble fragmented IPv4 or IPv6 packets and apply the filtering rules on the transport layer.

SF2: Reliability of the filtering core

nftables is able to correctly handle malformed packets to protect the integrity of the host system and its configuration.

SF3: Reliability of the administration interface

nftables is able to correctly handle malformed rules or requests provided by unprivileged users to the administration interface that are looking for an elevation of their privileges.

SF4: Logging

Rules can specify to *nftables* that the information or content of a packet must be logged and kept on the system.

6.2. Coverage of the Threats by the Security Features

	T1	T2	T3	T4
SF1	X			
SF2		X	X	
SF3		X	X	
SF4	X			X