Security Review Report

mptcpd – Multipath TCP Daemon

V 1.0
Amsterdam, March 25th, 2025
Public

Document Properties

Client	mptcpd – Multipath TCP Daemon
Title	Security review report
Target	mptcpd Multipath TCP Daemon https://github.com/multipath-tcp/mptcpd/tree/68fd9a156669ca7029a101f32282706df069cb4f
Version	1.0
Pentester	Tim Hummel
Authors	Tim Hummel, Marcus Bointon
Reviewed by	Marcus Bointon
Approved by	Melanie Rieback

Version control

Version	Date	Author	Description
0.1	February 27th, 2025	Tim Hummel	Initial draft
0.2	March 19th, 2025	Tim Hummel	Ready for internal review
0.3	March 24th, 2025	Marcus Bointon	Review
1.0	March 25th, 2025	Marcus Bointon	1.0

Contact

For more information about this document and its contents please contact Radically Open Security B.V.

Name	Melanie Rieback
Address	Science Park 608 1098 XH Amsterdam The Netherlands
Phone	+31 (0)20 2621 255
Email	info@radicallyopensecurity.com

Table of Contents

1 Introduction

1.1 Introduction

Between January 29, 2025 and March 19, 2025, Radically Open Security B.V. carried out a security review of mptcpd – Multipath TCP Daemon.

This report contains our conclusions as well as explanations how we performed the review.

1.2 Scope of work

The scope of the penetration test was limited to the following repository:

- mptcpd Multipath TCP Daemon https://github.com/multipath-tcp/mptcpd/tree/68fd9a156669ca7029a101f32282706df069cb4f

Other repositories and 3rd-party libraries are not in scope, except for the parts of the Embedded Linux library that are used in this project in version https://web.git.kernel.org/pub/scm/libs/ell/ell.git/commit/?id=170c49ade7620d45c2dd7dd2a1dd5be3f0879722. We also used the documentation found at https://www.mptcp.dev/ and https://mptcp-apps.github.io/mptcp-doc/index.html as references, but the documentation itself is not in scope.

1.3 Project objectives

ROS performed a security review of mptcpd – Multipath TCP Daemon. The main goal was to assess whether the user-space daemon offers any way to compromise it and potentially gain elevated privileges.

1.4 Timeline

The security audit took place between January 29, 2025 and March 19, 2025. In that time we used a total of 5 working days for the evaluation.

2 Results

During this crystal-box penetration test we found no issues.

To analyse the daemon we reviewed the documentation and the source code. The repository is small enough that it is possible to fully assess the code. C is a language prone to memory bugs, but parsing, string handling, and other critical code is outsourced to libraries, in particular the Embedded Linux library.

The project is well-designed and offers a limited attack surface, which is explored below:

- **The Netlink interface**
  This is the interface between the kernel and this daemon. We do not see how that can be abused by an attacker unless they already have higher privileges than the daemon.
- **Command line interface and config files**
  The command line and the config files offer overlapping functionality. We performed code review on the command line/config file parsing and found no issues with the parsing. We found no inherently vulnerable config options. Some functionality is implemented by the Embedded Linux Library, so we assessed the functions used within that, and likewise found no issues.
- **Plugin loading**
  The daemon has the ability to load plugins via `dlopen()`. Users have to be careful with this feature, as it allows running code with the same context/privilege level as the daemon and the `CAP_NET_ADMIN` capability. The daemon takes appropriate precautions, ensuring that neither the config nor the plugin directory is world-writeable. In line with these checks, the developers could consider also checking that the plugin itself is not world writeable. The default confirmation file and plugin directory are typically both only root accessible, but ultimately this does not prevent running malicious plugins.
  From a security standpoint, removing the ability to load plugins would be better, but this is an intentional feature, so we do not class it as a vulnerability.
  We recommend adding a note to the documentation to the effect that users/integrator/configurators of the daemon need to be careful to not misconfigure anything that might allow less-privileged users to add, load, or replace any plugins.

Beyond that, we only have positive remarks about the project:

- Running this as a separate daemon with limited privileges is an excellent security practice, and does not expand the kernel's codebase.
- The code is very well-structured, well-documented and self-explanatory; The comments are elaborate and show awareness of potential security issues.
- The project makes good use of library functions and avoids reimplementing existing well-tested functionality; This is good practice and avoids many pitfalls that could lead to memory corruption in C.

We also reviewed the code of mptcpize from the same repository and similarly found no issues. Even if there were vulnerabilities in here, their impact would be limited because it is run as a command line program and not a privileged daemon like mptcpd.

Appendix 1 Testing team

Tim Hummel	Tim Hummel is a security expert and developer. His specialty is hardware, crypto, and related software security. In his work he tests everything from apps, car components, payment solutions, white-box crypto, pay TV, smart cards, mobile devices, IoT, TPMs, TEEs, bootloaders, entertainment systems, transport cards, to web services and APIs.
Melanie Rieback	Melanie Rieback is a former Asst. Prof. of Computer Science from the VU, who is also the co-founder/CEO of Radically Open Security.