

# Automatic Integration of D Code With the Linux Kernel

*Friday 5 November 2021 15:20 (20 minutes)*

The Linux kernel is implemented in C, an unsafe programming language, which puts the burden of memory management, type and bounds checking, and error handling in the hands of the developer. Hundreds of buffer overflow bugs have compromised Linux systems over the years, leading to endless layers of mitigations applied on top of C.

In contrast, the D programming language offers automated memory safety checks and modern features such as OOP, templates and functional style constructs. In addition, interoperability with C is supported out of the box. However, to integrate a D module with the Linux kernel it is required that the needed C header files are translated to D header files. This is a tedious, time consuming, manual task. Although a tool to automate this process exists, called DPP, it does not work with the complicated, sometimes convoluted, kernel code.

In this paper, we improve DPP with the ability to translate any Linux kernel C header to D. Our work enables the development and integration of D code inside the Linux kernel, thus facilitating a method of making the kernel memory safe.

**Authors:** STANILOIU, Constantin Eduard (University POLITEHNICA of Bucharest); NITU, Razvan (University POLITEHNICA of Bucharest); BECERESCU, Cristian (Politehnica University of Bucharest); RUGHINIȘ, Răzvan Victor (University Politehnica of Bucharest)

**Presenter:** STANILOIU, Constantin Eduard (University POLITEHNICA of Bucharest)

**Session Classification:** Network Management && Open Source and GNU in Education and Research

**Track Classification:** Open Source and GNU in Education and Research