

EXAMENSARBETE Evaluation of Rust Codebases Using Public Information**STUDENT** Emil Eriksson**HANDLEDARE** Alexandru Dura (LTH) & Nikolaos Korkakakis (Volvo Car Sverige AB)**EXAMINATOR** Christoph Reichenbach

Att nysta ut höstacken: Vad finns egentligen i ditt mjukvaruprojekt?

POPULÄRVETENSKAPLIG SAMMANFATTNING **Emil Eriksson**

Väldigt sällan skrivs moderna mjukvaruprojekt i ett vakuum. Program kan innehålla flera tusen delar, från ännu fler programmerare, som enkelt laddas ner från nätet. Med mitt program blir det lättare att nysta i dessa delar, och med det har jag undersökt trender för över 2000 populära projekt i programmeringsspråket Rust.

Idag skrivs nästan inga program av en ensam person, team, eller ens företag. Istället förlitar man sig ofta på kod skriven av utvecklare och organisationer som väljer att dela med sig till allmänheten. Detta har bidragit till den stora mängd olika program, hemsidor, och applikationer vi idag tar för givet. Men det krävs stor tillit till att det man använder håller hög kvalitet. Detta har historiskt visat sig vara ett minst sagt naivt antagande. *cargo-indicate* är ett verktyg jag utvecklat som gör det lättare att ställa frågor om vad som ingår i ens program. Med hjälp av data från hemsidor, databaser, och koden själv gör det det möjligt att ställa frågor som *Använder jag någon skräpig kod ingen annan verkar använda?* genom sökningarna som utvecklarna själva kan skriva.

Detta verktyg har jag sedan använt för att lista ut hur det ser ut för program skrivna i det pop-

ulära programmeringsspråket Rust. Genom att kolla på data från en stor mängd populära projekt hittade jag flera trender. Till exempel verkar vissa projekt vara populära främst på grund av en mindre PR-kampanj, samtidigt som de innehåller kod som enkelt kan leda till farliga buggar. Det verkar också, något motsägelsefullt, som att ett projekt med många rapporter i databaser över kända problem är ett tecken på att projektet håller *hög* kvalitet och inte tvärtom. Det låter konstigt, men en uttråkad programmerare lär knappast ägna speciellt mycket tid att rapportera alla sina misstag.

Verktyget kan komma till användning för både utvecklare och forskare i framtiden som ett sätt att analysera Rust-projekt, och en van programmerare kan utan större besvär utöka det med ännu fler datakällor.