Dosáhnout stability za jakékoli situace je cílem spousty aplikací. Tento problém se týká také síťového stacku ANIS, který je součástí operačního systému reálného času PikeOS vyvíjeného společností SYSGO. PikeOS vyžaduje bezpečnost a stabilitu svých komponent, protože je používán v průmyslu jako je např. letectví, kde by nedostatek těchto vlastností mohl způsobit veliké škody. Vyzkoušená cesta pro ověření stability a bezpečnosti programu je jeho testování. Fuzz testování je technika automatického testování, která se snaží v programu najít chyby skrz generování náhodných vstupů. Jejím cílem je najít zranitelnosti a odhalit potenciální chyby, které mohou mít závažné důsledky na provoz aplikace. Další testovací technikou je long-run testing, přes který je aplikace vystavena náporu po delší časový úsek.

Jelikož ANIS je běžně dodávám jako součást PikeOS, musí také splňovat stejné bezpečnostní standardy jako PikeOS. My jsme s pomocí long-run a fuzz testování vytvořili testovací program pro síťový stack ANIS. Při tvorbě jsme kladli důraz na možnost nastavování našeho testu. Tento test vystavuje ANIS různým scénářům, které mají za úkol zatížit ANIS. Test používá fuzzing jako nástroj pro generování kombinací těchto scénařů a snaží se s jejich pomocí donutit ANIS k chybám. V rámci vývoje jsme opatřili test malým vzorkem scénářů, které vystavují ANIS různému síťovému provozu. Všechny scénáře mají předem definovanou množinu parametrů. Změnou hodnot těchto parametrů jsme schopni generovat různorodější scénáře. Scénáře s jejich parametry jsou generovány vždy před začátkem testování. Jako možnost konfigurace má fuzzer svou vlastní množinu parametrů, kterou je uživatel schopen ovlivnit způsob, jakým budou data pro test generována.