



Sven Defatsch

Studenten/-innen	Sven Defatsch
Dozenten/-innen	Prof. Dr. Andreas Steffen
Co-Betreuer/-innen	Tobias Brunner
Themengebiet	Sicherheit

Google OSS-Fuzz

Security Testing des strongSwan Codes



strongSwan

```
INFO: seed: 263417882
INFO: loaded 2 modules (7976 guards): {0x7f0e01b5d40, 0x7f0e01b19d0}, {0x7f0e01b, 0x7f0e01b}
INFO: heap limit is not provided, using 64
INFO: A corpus is not provided, starting from an empty corpus
#00 unit(s)
#1 INITED cov: 659 fct: 659 corp: 1/1b exec/s: 0 rss: 290k
#1 NEW cov: 663 fct: 663 corp: 1/20b exec/s: 0 rss: 320k L: 47 MS: 1 InsertRepeatedBytes-
#12 NEW cov: 663 fct: 663 corp: 1/58b exec/s: 0 rss: 320k L: 2 MS: 1 InsertByte-
#12 NEW cov: 663 fct: 663 corp: 1/25b exec/s: 0 rss: 320k L: 2 MS: 1 CopyPart-
#13 NEW cov: 663 fct: 668 corp: 1/55b exec/s: 0 rss: 330k L: 3 MS: 2 CopyPart-InsertByte-
#19 NEW cov: 664 fct: 672 corp: 1/120b exec/s: 0 rss: 320k L: 46 MS: 1 ShuffleBytes-CrossOver-InsertRepeatedBytes-
#49 NEW cov: 664 fct: 675 corp: 1/183b exec/s: 0 rss: 370k L: 44 MS: 1 ShuffleBytes-ChangeByte-CrossOver-
#63 NEW cov: 664 fct: 678 corp: 1/210b exec/s: 0 rss: 340k L: 46 MS: 1 ChangeByte-ShuffleBytes-CopyPart-Cmp-Cross-
#68 NEW cov: 664 fct: 688 corp: 1/313b exec/s: 0 rss: 400k L: 44 MS: 1 InsertByte-CrossOver-
#74 NEW cov: 664 fct: 689 corp: 1/325b exec/s: 0 rss: 430k L: 44 MS: 1 InsertByte-CrossOver-ChangeByte-ChangeInt-
#82 NEW cov: 668 fct: 688 corp: 1/1370b exec/s: 0 rss: 430k L: 1 MS: 1 ChangeByte-
#117 NEW cov: 668 fct: 688 corp: 12/600b exec/s: 0 rss: 400k L: 40 MS: 1 ChangeByte-
#118 NEW cov: 668 fct: 688 corp: 13/477b exec/s: 0 rss: 400k L: 37 MS: 1 ChangeByte-CrossOver-
#123 NEW cov: 668 fct: 693 corp: 15/1570b exec/s: 0 rss: 400k L: 48 MS: 1 CopyPart-InsertRepeatedBytes-
#126 NEW cov: 669 fct: 694 corp: 15/176b exec/s: 0 rss: 400k L: 39 MS: 1 CopyPart-InsertRepeatedBytes-ShuffleBytes-
#152 NEW cov: 669 fct: 693 corp: 15/165b exec/s: 0 rss: 400k L: 48 MS: 1 InsertByte-ChangeInt-ChangeByte-Cross-
#153 NEW cov: 669 fct: 699 corp: 17/650b exec/s: 0 rss: 540k L: 40 MS: 2 Cmp-ChangeInt-DE: "\xff\xff\xff\xff\xff\xff\xff\xff"
#156 NEW cov: 669 fct: 701 corp: 18/720b exec/s: 0 rss: 570k L: 42 MS: 1 CopyPart-ChangeInt-ChangeByte-ChangeInt-
#158 NEW cov: 669 fct: 702 corp: 19/780b exec/s: 0 rss: 1120k L: 44 MS: 1 ShuffleBytes-CopyPart-CopyPart-ChangeInt-
#162 NEW cov: 671 fct: 703 corp: 21/780b exec/s: 0 rss: 1270k L: 2 MS: 1 CopyPart-ChangeByte-
#227 NEW cov: 673 fct: 708 corp: 22/820b exec/s: 0 rss: 150k L: 41 MS: 1 InsertRepeatedBytes-
#248 pulse cov: 673 fct: 708 corp: 22/820b exec/s: 1024 rss: 200k
#258 pulse cov: 674 fct: 709 corp: 23/880b exec/s: 1188 rss: 400k L: 51 MS: 1 InsertRepeatedBytes-
#277 NEW cov: 676 fct: 709 corp: 23/880b exec/s: 1239 rss: 410k L: 2 MS: 1 CopyPart-ChangeInt-
#278 NEW cov: 676 fct: 709 corp: 23/880b exec/s: 1249 rss: 410k L: 24 MS: 1 CopyPart-ChangeInt-CopyPart-Insert-
#281 NEW cov: 676 fct: 709 corp: 23/880b exec/s: 1249 rss: 410k L: 3 MS: 1 CopyPart-
#287 NEW cov: 703 fct: 705 corp: 27/920b exec/s: 959 rss: 400k L: 39 MS: 1 InsertRepeatedBytes-
#289 pulse cov: 703 fct: 705 corp: 27/920b exec/s: 1024 rss: 420k
#316 pulse cov: 703 fct: 706 corp: 28/930b exec/s: 1084 rss: 430k L: 0 MS: 1 CrossOver-CopyPart-CopyPart-ChangeInt-
#319 NEW cov: 703 fct: 709 corp: 29/370b exec/s: 1024 rss: 460k L: 3 MS: 1 InsertByte-ChangeInt-
#318 NEW cov: 702 fct: 708 corp: 30/960b exec/s: 1037 rss: 460k L: 26 MS: 1 Cmp-ShuffleBytes-Cmp-ChangeInt-ChangeInt-
#320 NEW cov: 702 fct: 709 corp: 31/960b exec/s: 1037 rss: 460k L: 0 MS: 1 ChangeInt-
#321 NEW cov: 705 fct: 772 corp: 32/1010b exec/s: 1033 rss: 460k L: 47 MS: 1 CrossOver-EraseBytes-PerAutoDict-Cmp-
#323 NEW cov: 709 fct: 775 corp: 33/1025b exec/s: 1045 rss: 460k L: 7 MS: 1 ShuffleBytes-PerAutoDict-DE: "-----"
#327 NEW cov: 708 fct: 775 corp: 34/1060b exec/s: 1159 rss: 460k L: 3 MS: 1 CrossOver-
#367 NEW cov: 708 fct: 778 corp: 36/1113b exec/s: 1177 rss: 460k L: 45 MS: 1 PerAutoDict-DE: "-----"
#367 pulse cov: 709 fct: 778 corp: 37/1113b exec/s: 1024 rss: 460k
#369 NEW cov: 716 fct: 788 corp: 38/1140b exec/s: 1024 rss: 460k L: 1 MS: 1 ChangeByte-
#392 NEW cov: 709 fct: 779 corp: 37/1113b exec/s: 1024 rss: 460k
#399 NEW cov: 716 fct: 790 corp: 39/1150b exec/s: 1044 rss: 460k L: 2 MS: 1 CrossOver-CopyPart-EraseBytes-
#419 NEW cov: 716 fct: 790 corp: 39/1150b exec/s: 1044 rss: 460k L: 2 MS: 1 ShuffleBytes-ChangeInt-CopyPart-
```

Aktiver Fuzzing Prozess

```
OVERVIEW
Crash State: strongswan_fuzz_certs
Crash Type: Timeout (exceeds 25 secs) Fuzzer: libFuzzer_strongswan_fuzz_certs
Crash Address: --- Job Title: libfuzzer_asan_strongswan
Issue: 1361 (from its group) Platform: linux
Created: 24.5.2017 10:45:59 Sanitizer: address (ASAN)
Project: oss-fuzz Deletion: Will be auto-deleted on 31.5.2017 10:45:59
Group: 5992309172731904

CRASH STACKTRACE
-- ORIGINAL STACKTRACE ON REVISION 0408FF5394C0547CF523675092C2C50CED0AD (124 LINES) -----
62 lines omitted
63 #7960 NEW cov: 1560 fct: 5888 corp: 1280/7730b exec/s: 812 rss: 1020k L: 1070 MS: 1 ChangeInt-
64 #8192 pulse cov: 1560 fct: 5888 corp: 1280/7730b exec/s: 630 rss: 1020k
65 #6240 NEW cov: 1560 fct: 5881 corp: 1281/7740b exec/s: 633 rss: 1020k L: 1260 MS: 1 Cmp-DE: "\x00\x00\x00\x00\x00\x00\x00\x00"
66 ALARM: working on the last unit for 37 seconds
67 and the timeout value is 25 (use -timeout=N to change)
68 MS: 1 InsertRepeatedBytes - base unit: 7f8b08b7167a263fc1335d11d99e2c9af069
69 artifact_prefix: /; Test unit written to /!timeout-7ec309f67819080c1968c131020d6d4e79141
70 --== ERROR: libFuzzer: timeout after 37 seconds
71 #0 0x4e9305 in __sanitizer_print_stack_trace_asan_rtl
72 #1 0x53ae02 in fuzzer:IAIARCallback() /src/libfuzzer/FuzzerLoop.cpp:234:7
73 #2 0x7f6450b08f in libpthread.so.0
74 #3 0x5046d1 in __sanitizer_cov_trace_pc_guard /src/libfuzzer/FuzzerTracePC.cpp:296:30
75 #4 0x633a39 in iterate /src/strongswan/src/libstrongswan/asn1/asn1_parser.c:133:6
76 #5 0x67e587 in x509_parse_crldistributionPoints /src/strongswan/src/libstrongswan/plugins/x509/x509_cert.c:896:9
77 #6 0x68106d in parse_certificate /src/strongswan/src/libstrongswan/plugins/x509/x509_cert.c:1480:12
```

Fuzzing Ergebnisse bei in OSS mit Stack Trace

Ausgangslage: Im Jahr 2016 hat Google das sogenannte OSS-Fuzz Programm angekündigt, welches sich zum Ziel gemacht hat, ausgewählte Open Source Projekte sicherer zu machen. Dafür wird die Möglichkeit geboten, die enorme Rechenleistung der Google Infrastruktur für kontinuierliches Fuzzing, inklusive übersichtlichem Reporting zu nutzen. Die IPsec-basierte strongSwan Open Source VPN Lösung wird an der HSR entwickelt und weltweit eingesetzt. Für eine Software im Sicherheitsbereich ist es besonders wichtig, die Anzahl Fehler im Code so gering wie möglich zu halten. Dabei bietet Fuzzing eine gute Möglichkeit, latente Fehler zu entdecken und somit die Codequalität zu steigern.

Vorgehen/Technologien: Für das Fuzzing wurde libFuzzer eingesetzt, die zu diesem Zeitpunkt einzige von Google unterstützte Fuzzing Engine. Als vielversprechendes und relativ leicht umzusetzendes Fuzz Target bot sich der in strongSwan eingebaute X.509 Zertifikatsparser an. Die Fuzzing Umgebung wurde zuerst lokal getestet, um Erfahrungen im Umgang mit der Nutzung von libFuzzer zu sammeln und die notwendigen Anpassungen im strongSwan Build-Prozess zu bestimmen. Anschliessend wurde das Fuzzing auf die mächtige Google Infrastruktur verlagert, um den Code auf Herz und Nieren zu testen.

Ergebnis: Als Hauptresultat dieser Arbeit wurden Erkenntnisse über die Integration von strongSwan in das Google OSS-Fuzz Projekt gewonnen, insbesondere welche Stolpersteine beachtet werden müssen. Erfreulicherweise wurden beim intensiven Fuzzing schon nach kurzer Zeit mehrere Fehler unterschiedlicher Tragweite im strongSwan Code gefunden, die sofort behoben wurden und in Form von Security Patches in strongSwan eingeflossen sind. Somit ist das Hauptziel des Fuzzings, die strongSwan Codequalität zu verbessern, klar erreicht worden.