COMP9447 15s2

5A - Fuzzing

What is fuzzing?

- Software testing technique used to find bugs by feeding random/pseudo random or purposely bad data to an application
- Usually generated by another program (the fuzzer)
- If it crashes, then awesome. Save that ‘fuzz’ and move onto the next test

History of Fuzzing

- Barton Miller at University of Wisconsin "Operating System Utility Program Reliability - The Fuzz Generator“ - 1989
- Dave Aitel released two fuzzers in 2001 - SPIKE fuzzing framework, and sharefuzz. Spike in particular was the first block based fuzzer publicly known, and made fuzzing a huge thing in the security world – 2001
- Big gap there of nothing..
- Many fuzzing toolkits have been released since then

Benefits of fuzzing (pros) compared to more manual review

- Reappliable effort – Fuzzer creation is generally focused on protocol, file format, etc. – so you can take your PDF fuzzer and use it on every PDF parser, or take your SSL fuzzer and use it on every SSL implementation
- Write once, use many times
- Lower skill requirement than in-depth reverse engineering or source code review
- Bugs found are guaranteed to be reachable, if you are fuzzing from reachable interface
- Largely, better bang for back
- Can be used to estimate “watermark” of security – i.e. fuzzer library of common network protocols vs. firewall to see if the vendor has their SDLC act together
- Will find memory corruption, overflows, etc.

Cons

- Bugs are often shallow - Other people may find them. Now that exploit writing time is often very large (compared to vuln finding time)
- Hard to get full coverage of program – remember that the least used paths are more likely to be vulnerable
- Multistage vulns –I .e. those which require multiple concurrent things to happen to occur are hard to target – i.e. think about a network protoco, where if the first packet of a particular type of communication is slightly malformed in a certain way, and then the second packet is in a different way, then a bug occurs. Very hard to cover (but not impossible..)
- Will not find smarter things like backdoors or complex race conditions

Two Broad “Families” of Fuzzing + Targets

Families:

- **Generation Based:** The fuzzer itself generates the bad data
- **Mutation Based:** The fuzzer mutates good data in some way

Targets:

- Network protocols (server:apache/smb, client:browsers)
- File format (acrobat/office/flash – includes programming languages i.e. JavaScript)
- API (system calls)

Anatomy of a Fuzzer

- Fuzzing engine – creates the test cases either by generation or mutation
- Test harness
  - Process monitor
  - Starts the application each time
  - Feeds it the input
  - Rinse and repeat
  - Records the conditions under which a crash occurred
  - Tries to grab some useful diagnostic information to speed the bug hunt process up (or vuln verification process)
- Do not underinvest in your test harness!

The simplest possible form of a generation fuzzer

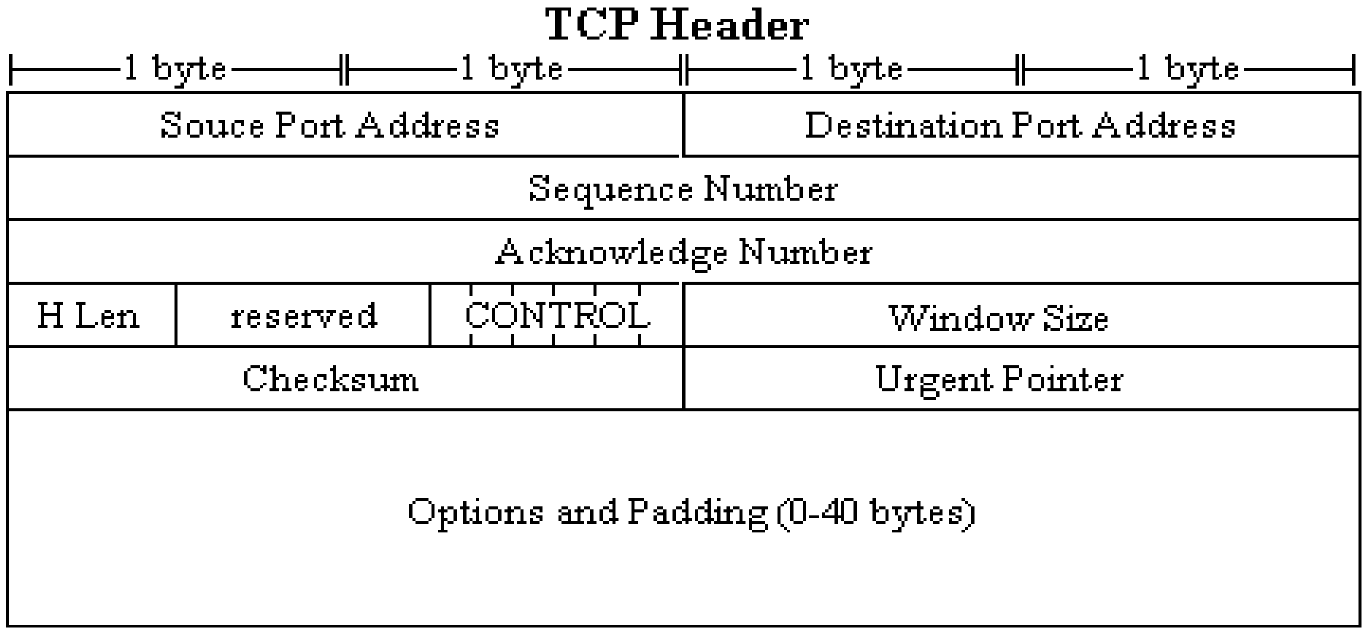
cat /dev/urandom | ./program

- Or even
  cat /dev/urandom | nc <ip> <port>
- Extremely basic
- Don't even need to know the protocol
- Need to get lucky - not very effective as most protocols/file formats need specific data in specific places (e.g. a header/footer/checksums)
- We have found real exploitable remote vulnerabilities doing this ;(
- Adding a header makes it a lot more productive ☺

i.e. { echo -n 'lol' & dd if=/dev/urandom bs=4 count=1 2>/dev/null ;} | nc <ip> <port>

Smarter Generational Fuzzing

- Attempt to model the protocol and then try your bad/random data in the most likely areas to cause a problem
- Most protocols are well defined, etc. or look at the code a little bit (to get headers for example)



Block / Based Fuzzers

Attempts to model a protocol in blocks.

Originally made public/popular with Spike in 2001, there's now a huge number.

"GET / HTTP/1.0" could be represented as:

```
s_string("GET");
space();
s_string("/");
space();
s_string("HTTP");
delimiter("/");
s_interger("1")
delimiter(".");
s_interger("0");
```

GET / HTTP/1.0

AAAAAAAAA / HTTP/1.0

BBBBBBBBBB / HTTP/1.0

%S%S%S%S%S / HTTP/1.0

../../../../../../ / HTTP/1.0

..

Later

GET %S%S%S%S%S HTTP/1.0

GET ../../../../../../../ HTTP/1.0

Later

GET / BBBBBBBBB/1.0

etc.

Slightly better, block-based, sulley example

- s_initialize
- s_initialize("HTTP")
- s_group("verbs", values=['GET','HEAD','POST','TRACE'])
- s_block_start("header", group="verbs")
- s_delim(' ')
- s_delim('/')
- s_string('index.html')
- s_delim(' ')
- s_string('HTTP')
- s_delim('/')
- s_string('1')
- s_delim('.')
- s_string('1')
- s_static('\r\n\r\n')
- s_block_end()

Block-Based Fuzzers

- The better you can map a protocol (in particular multi packet exchanges) the more likely you are to succeed
- Enrich your data set as much as possible. Example here is HTTP request header, i.e.

Expect: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Cache-control: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Request fields list from Wikipedia: Good

Common non-standard request fields from Wikipedia: Better

Looking at the binary and stripping out all the requests fields it actually knows about, including some that aren’t on Wikipedia, and eliminating all the fields in protocol that it doesn’t support: Best

Look for the undocumented stuff

- Once again, you really want to search deep in the parts of the target that will not be exercised often
- This is even more so on closed-source things
- Open them up in IDA and look at string collections (i.e. for example above, all the HTTP request headers) and see if there are any that are unused
- Then dig around where they are used, and see what kind of stuff is going on

Mutation / Mangling

- Taking an existing file/packet, mangling it, feeding it back to the program
- Usually done with file formats
- Started with literally, flip a random of bit in a valid file.
- Will only fuzz the features in the initial file
- Hopefully with enough initial files you can fuzz all features!
- Lazy way of doing it as you don’t have to learn the file format / protocol
- Still get owned by crc/checksums

Mangling continued

- Still finds bugs!
- Charlie Miller found tonnes of bugs in real software in 2010 with 5 lines of python – still works now too
- Babysitting my army of Monkeys: https://fuzzinginfo.files.wordpress.com/2012/05/cmiller-csw-2010.pdf
- Very reliant on the initial files chosen

Charlie Millie Corpus Technique

- You need to build a good “corpus” of data/files to fuzz, generally based on code coverage
- Directly copy and pasted from the monkey pack:
- Example PDF:
  - Found 80,000 PDF’s on Internet
  - Used Adobe Reader + Valgrind in Linux to measure code coverage
  - Reduced to 1,515 files of ‘equivalent’ code coverage
  - Same bang as fuzzing all 80k in 2% of the time

Maximizing Code Coverage

- A good fuzzer will maximize how many code paths are executed
- You generally do this via instrumenting the target and watching which basic blocks it triggers
- You will find that some input files and paths hit supersets of others; you can remove the ones which don’t add anything new

Harden the client + Useful tools

- You want to detect any corruption or badness as soon as possible
- Thus, you want to turn memory protections all the way up on the system your target is on (i.e. the target)
- Grsecurity with all memory protections
- Guard pages/something to catch access violations – i.e. libgmalloc, libefence, memcheck

- There are tools like !exploitable as part of windbg which will tell you if something is likely exploitable

Picking bad data

- Certain kinds of data (and certain lengths of data) are more likely to trigger bugs
- Searching a 4-byte int for 1-0xFFFFffff takes too long. The best values are all of the powers of 2, plus or minus {1..5}.
- i.e. 1024, 1023, 1022, 1021, 1020, 1019, 1025, 1026, 1027, 1028, 1029.
- 2048, 2047, 2046, 2045, 2044, 2043, 2049, etc.
- It is also a very good idea on mutagenic fuzzers to rather than pick a completely random value, modify the existing value in a certain way, i.e. add 1 or 2 or 3 or 4 to it, or double it, or double it and add one.
- Research on mutagenic stuff (what works)

http://lcamtuf.blogspot.com.au/2014/08/binary-fuzzing-strategies-what-works.html

Picking bad data - Strings

All the usual suspects:

- AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
- PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
- Ppppppppppppppppppppppppppppppp
- PPpppPppPpppPppPPppPPpPPPpPpPpPpppppP
- 3333333333333333333333333333333333333
- :::
- \/
- ..\
- \`/sbin/sudo /sbin/halt\`
- \`halt\`

Retrospective Selection

- Retrospective selection is important when making a fuzzer – particularly generation based fuzzers.
- What this means is basically, make sure that all the actual vulnerabilities identified in the target protocol can be found with your tool
- I.e. if you are writing a HTTP server fuzzer, look at all the reported vulnerabilities that a fuzzer could find (i.e. memory corruption) in the last several years, and make sure your fuzzer can find them.
- If not, add the bits to your fuzzer that mean it can.
- Repeat. :-)

Jones

- Proxy block based fuzzer fionnbharr wrote which had some success
- Pulls packets off the wire
- If they match a pattern -> pull them apart based off the matched class
- Fuzz particular variables, fix up checksums/lengths/whatever
- Inject it back into the stream
- Fixes the problem of having to create the whole conversation\

There's now an open-source version similar to this called proxyfuzz which has found some good bugs

Mycotoxin

- A client modification fuzzer I wrote whilst pretty drunk
- Requires you to recompile any client with –oS (so all instructions use stack)
- It then does random interchanges (i.e. wait a random number of syscalls, step a random number of instructions, and change the currently being pushed integer or string to a fuzz value)
- Defeats compression, encryption etc. etc.
- Found some decent bugs I was too lazy to do anything about
- Code is available if anyone wants to rewrite it in C

american fuzzy lop

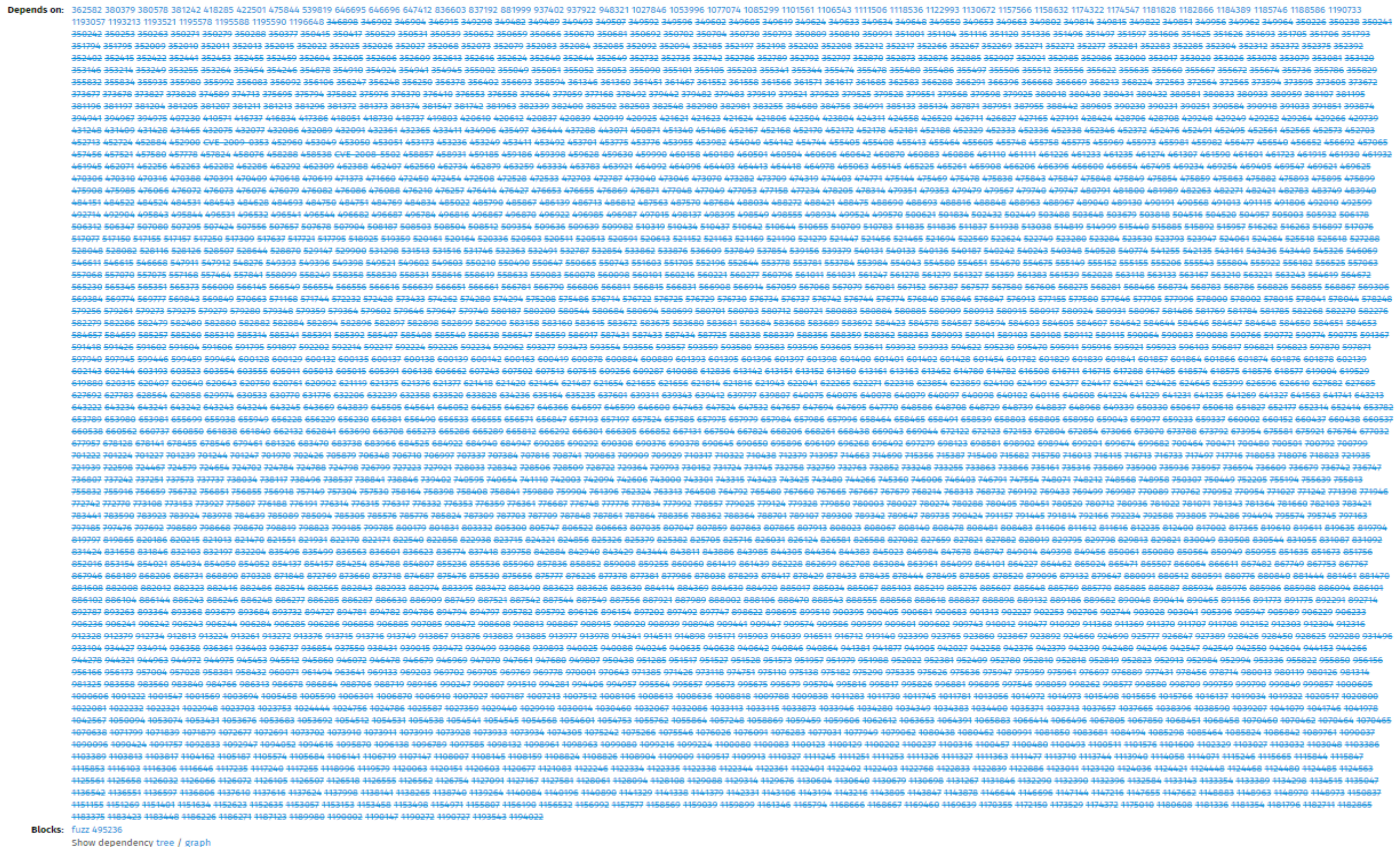
American fuzzy lop is a security-oriented fuzzer that employs a novel type of compile-time instrumentation and genetic algorithms to automatically discover clean, interesting test cases that trigger new internal states in the targeted binary. This substantially improves the functional coverage for the fuzzed code. The compact synthesized corpora produced by the tool are also useful for seeding other, more labor- or resource-intensive testing regimes down the road.

http://lcamtuf.coredump.cx/afl/

IJG jpeg [1]	libjpeg-turbo [1,2]	libpng [1]
libtiff [1,2,3,4,5]	mozjpeg [1]	PHP [1,2,3,4]
Mozilla Firefox [1,2,3,4]	Internet Explorer [1,2,3,4]	Apple Safari (1) (2)
Adobe Flash / PCRE [1,2]	sqlite [1,2,3,4]...	OpenSSL [1,2,3,4]
LibreOffice [1,2,3,4]	poppler [1]	freetype [1,2]
GnuTLS [1]	GnuPG [1,2,3,4]	OpenSSH [1,2,3]
bash (post-Shellshock) [1,2]	tcpdump [1,2,3,4,5,6,7,8]	JavaScriptCore [1,2,3,4]
pdfium [1,2]	ffmpeg [1,2,3,4]	libmatroska [1]
libarchive [1,2,3,4,5,6] ...	wireshark [1,2]	ImageMagick [1,2,3,4,5,6,7,8] ...
BIND [1]	QEMU [1,2]	lcms [1]
Oracle BerkeleyDB [1,2]	Android / libstagefright [1,2]	iOS / ImageIO [1]
FLAC audio library [1,2]	libsndfile [1,2,3]	less / lesspipe [1,2,3]
strings (+ related tools) [1,2,3,4,5,6,7]	file [1,2,3,4]	dpkg [1]
rcs [1]	systemd-resolved [1,2]	libyaml [1]
Info-Zip unzip [1,2]	libtasn1 [1,2]	OpenBSD pfctl [1]
NetBSD bpf [1]	man & mandoc [1,2,3,4,5] ...	IDA Pro [reported by authors]

clamav [1,2,3,4,5]	libxml2 [1]	glibc [1]
clang / llvm [1,2,3,4,5,6,7,8] ...	nasm [1,2]	ctags [1]
mutt [1]	procmail [1]	fontconfig [1]
pdksh [1,2]	Qt [1]	wavpack [1]
redis / lua-cmsgpack [1]	taglib [1,2,3]	privoxy [1]
perl [1,2,3,4,5,6,7]...	libxmp	radare2 [1,2]
SleuthKit [1]	fwknop [reported by author]	X.Org [1]
exifprobe [1]	jhead [?]	capnproto [1]
Xerces-C [1]	metacam [1]	djvulibre [1]
exiv [1]	Linux btrfs [1,2,3,4]	Knot DNS [1]
curl [1,2]	wpa_supplicant [1]	libde265 [reported by author]
dnsmasq [1]	libbpg (1)	lame [1]
libwmf [1]	uudecode [1]	MuPDF [1]
imlib2 [1]	libraw [1]	libbson [1]
libsass [1]	yara [1,2,3,4]	W3C tidy-html5 [1]
UPX [1]	FreeBSD syscons [1]	John the Ripper [1,2]

jsfunfuzz



Depends on: 362582 380379 380578 381242 418285 422501 475844 539819 646695 646696 647412 836603 837192 881999 937402 937922 948321 1027846 1053996 1077074 1085299 1101561 1106543 1111506 1118536 1122993 1130672 1157566 1158632 1174322 1174547 1181828 1182866 1184389 1185746 1188586 1190733
1193057 1193213 1193521 1195578 1195588 1195590 1196648 ~~346898 346902 346904 346915 349298 349482 349489 349493 349507 349592 349596 349602 349605 349619 349624 349633 349634 349648 349650 349653 349663 349802 349814 349815 349822 349851 349956 349962 349964 350226 350238 350241~~
~~350242 350253 350263 350271 350279 350288 350377 350415 350417 350529 350531 350539 350652 350659 350666 350670 350681 350692 350702 350704 350730 350793 350809 350810 350991 351001 351104 351116 351120 351336 351496 351497 351597 351606 351625 351626 351693 351705 351706 351793~~
~~351794 351795 352009 352010 352011 352013 352015 352022 352025 352026 352027 352068 352073 352079 352083 352084 352085 352092 352094 352185 352197 352198 352202 352208 352212 352217 352266 352267 352269 352271 352272 352277 352281 352283 352285 352304 352312 352372 352375 352392~~
~~352402 352415 352422 352441 352453 352455 352459 352604 352605 352606 352609 352613 352616 352624 352640 352644 352649 352732 352735 352742 352786 352789 352792 352797 352870 352873 352876 352885 352907 352921 352985 352986 353000 353017 353020 353026 353078 353079 353081 353120~~
~~353146 353214 353249 353255 353264 353454 354246 354878 354910 354924 354941 354945 355002 355049 355051 355052 355053 355090 355101 355105 355203 355341 355344 355474 355478 355480 355486 355497 355506 355512 355556 355622 355635 355660 355667 355672 355674 355736 355786 355829~~
~~355832 355834 355935 355980 355992 356083 356092 356106 356247 356248 356250 356378 356402 356693 358594 361346 361360 361451 361467 361552 361558 361566 361571 361617 361685 362583 366288 366291 366396 366668 366669 368213 368224 372563 372564 372565 373594 373595 373605 373672~~
~~373677 373678 373827 373828 374589 374713 375695 375794 375882 375976 376370 376410 376553 376558 376564 377059 377168 378492 379442 379482 379483 379519 379521 379523 379525 379528 379551 379568 379598 379925 380018 380430 380431 380432 380581 380833 380933 380959 381107 381195~~
~~381196 381197 381204 381205 381207 381211 381213 381296 381372 381373 381374 381547 381742 381963 382339 382400 382502 382503 382548 382980 382981 383255 384680 384756 384991 385133 385134 387871 387951 387955 388442 389605 390230 390231 390251 390584 390918 391033 391851 393874~~
~~394941 394967 394975 407230 410571 416737 416834 417386 418051 418730 418737 419803 420610 420612 420837 420839 420919 420925 421621 421623 421624 421806 422504 423804 424311 424558 426520 426711 426827 427165 427191 428424 428706 428708 429248 429249 429252 429264 429266 429739~~
~~431248 431409 431428 431465 432075 432077 432086 432089 432091 432361 432365 433411 434906 435497 436444 437288 443071 450871 451340 451486 452167 452168 452170 452172 452178 452181 452188 452329 452333 452336 452338 452346 452372 452476 452491 452495 452561 452565 452573 452703~~
~~452713 452724 452884 452900 CVE-2009-0353 452960 453049 453050 453051 453173 453236 453249 453411 453492 453701 453775 453776 453955 453982 454040 454142 454744 455405 455408 455413 455464 455605 455748 455758 455775 455969 455973 455981 455982 456477 456540 456652 456692 457065~~
~~457456 457521 457580 457778 457824 458076 458288 458538 CVE-2008-5502 458857 458931 459185 459186 459398 459628 459630 459990 460158 460180 460501 460504 460606 460642 460870 460883 460886 461110 461111 461226 461233 461235 461274 461307 461590 461601 461723 461915 461930 461932~~
~~461945 462071 462256 462263 462282 462286 462292 462309 462388 462407 462560 462734 462879 463259 463334 463783 463921 464092 464096 464403 464413 464418 464978 465063 465145 465225 465261 465908 466206 466396 466600 466654 467495 469234 469254 469405 469547 469621 469625~~
~~470306 470310 470316 470388 470391 470409 470618 470619 471373 471660 472450 472454 472508 472528 472533 472703 472787 473040 473046 473070 473282 473709 474319 474403 474771 475144 475469 475478 475838 475843 475847 475848 475849 475854 475859 475863 475882 475893 475895 475899~~
~~475908 475985 476066 476072 476073 476076 476079 476082 476086 476088 476210 476257 476414 476427 476653 476655 476869 476871 477048 477049 477053 477158 477234 478205 478314 479351 479353 479479 479567 479740 479747 480791 481800 481989 482263 482271 482421 482783 483749 483940~~
~~484151 484522 484524 484531 484543 484628 484693 484750 484751 484769 484834 485022 485790 485867 486139 486713 486812 487563 487570 487684 488034 488272 488421 488475 488690 488693 488816 488848 488963 488967 489040 489130 490191 490568 491013 491115 491806 492010 492599~~
~~492714 492904 495843 495844 496531 496532 496541 496544 496682 496687 496784 496816 496867 496870 496922 496985 496987 497015 498137 498395 498549 498555 498934 499524 499570 500621 501834 502432 502449 503488 503648 503679 503818 504516 504520 504957 505003 505932 506178~~
~~506312 506347 507080 507295 507424 507556 507657 507678 507904 508187 508503 508504 508512 509354 509636 509639 509982 510319 510434 510437 510642 510644 510655 510709 510783 511835 511836 511837 511938 513038 514819 514999 515440 515885 515892 515957 516262 516263 516897 517076~~
~~517077 517150 517155 517157 517250 517309 517637 517721 517795 518925 519359 520161 520164 520336 520503 520511 520513 520591 520613 521152 521163 521169 521190 521279 521447 521456 521465 521694 522569 522624 522749 523280 523284 523530 523793 523947 524061 524264 525518 525618 527288~~
~~528048 528082 528116 528126 528507 528644 528870 529147 529900 531298 531513 531516 531746 532363 532491 532787 532854 533862 533876 536609 537849 537854 539156 539379 540131 540133 540136 540187 540242 540243 540348 540528 540774 541255 542135 543161 543436 543440 545326 546069~~
~~546611 546615 546668 547911 547912 548276 549393 549396 549398 549521 549602 549603 550210 550490 550647 550665 550743 551603 551705 552196 552644 553778 553781 553784 553984 554043 554580 554651 554670 554675 555149 555152 555155 555206 555543 555804 555922 556182 556525 557063~~
~~557068 557070 557075 557168 557464 557841 558099 558249 558358 558530 558531 558616 558619 558633 559083 560078 560098 560101 560216 560221 560277 560796 561011 561031 561247 561278 561279 561327 561359 561383 561539 562028 563118 563133 563167 563210 563221 563243 564619 564672~~
~~565230 565345 565351 565373 566000 566145 566549 566554 566556 566616 566639 566651 566661 566781 566790 566806 566811 566815 566831 566908 566914 567059 567068 567079 567081 567152 567387 567577 567580 567606 568275 568281 568466 568734 568783 568786 568826 568855 568867 569306~~
~~569384 569774 569777 569843 569849 570663 571168 571744 572232 572428 573433 574262 574280 574294 575208 575486 576714 576722 576725 576729 576730 576734 576737 576742 576744 576774 576840 576846 576847 576913 577155 577580 577646 577705 577996 578000 578002 578015 578041 578044 578248~~
~~579256 579261 579273 579275 579279 579280 579348 579359 579364 579602 579646 579647 579740 580187 580200 580544 580684 580694 580699 580701 580703 580712 580721 580883 580884 580885 580909 580913 580915 580917 580924 580931 580967 581486 581769 581784 581785 582268 582270 582276~~
~~582279 582286 582479 582480 582880 582882 582884 582894 582896 582897 582898 582899 582900 583158 583160 583615 583672 583675 583680 583681 583684 583688 583689 583692 584423 584578 584587 584594 584603 584605 584607 584642 584644 584646 584647 584648 584650 584651 584653~~
~~584657 584659 585257 585260 585310 585314 585341 585391 585392 585407 585408 585540 586538 586547 586559 586917 587431 587433 587434 587725 588338 588339 588356 588359 588362 588363 589093 589101 589103 589108 589112 589115 590064 590083 590088 590766 590772 590774 590775 591367~~
~~591418 591426 591602 591604 591606 591795 591897 592202 592214 592217 592224 592226 592234 592962 593277 593473 593554 593556 593557 593559 593580 593583 593596 593605 593611 593932 593933 594622 595230 595470 595911 595916 595921 595923 596103 596817 596821 596823 597870 597871~~
~~597940 597945 599446 599459 599464 600128 600129 600132 600135 600137 600138 600139 600142 600163 600419 600878 600884 600889 601393 601395 601396 601397 601398 601400 601401 601402 601428 601454 601782 601829 601839 601841 601857 601864 601866 601874 601876 601878 602139~~
~~602143 602144 603193 603523 603554 603555 605011 605013 605015 605391 606138 606662 607243 607502 607513 607515 609256 609287 610088 612836 613142 613151 613152 613160 613161 613163 613452 614780 614782 616508 616711 616715 617288 617485 618574 618575 618576 618577 619004 619529~~
~~619880 620315 620407 620640 620643 620750 620761 620902 621119 621375 621376 621377 621418 621420 621464 621487 621654 621655 621656 621814 621816 621943 622041 622265 622271 622318 623854 623859 624100 624199 624377 624417 624421 624426 624645 625399 626596 626610 627682 627685~~
~~627692 627783 628564 629858 629974 630533 630770 631776 632206 632239 632358 633520 633828 634236 635164 635235 637601 639311 639343 639412 639797 639807 640075 640076 640078 640079 640097 640098 640102 640116 640608 641224 641229 641231 641235 641269 641327 641563 641741 643213~~
~~643222 643234 643241 643242 643243 643244 643245 643669 643839 645505 645641 646052 646255 646267 646366 646597 646599 646600 647463 647524 647532 647657 647694 647695 647770 648586 648708 648729 648739 648837 648968 649339 650330 650617 650618 651827 652177 652314 652414 653782~~
~~653789 653980 653981 655699 655938 655949 656228 656229 656230 656381 656490 656533 656555 656571 656847 657193 657197 657524 657585 657975 657979 657984 657986 657996 658464 658465 658491 658539 658803 658805 658950 659043 659077 659233 659337 660002 660152 660437 660438 660537~~
~~660538 660562 660737 660850 661838 661840 662132 662841 663690 663708 665273 665286 665289 665812 666292 666301 666305 666852 667131 667504 667824 668206 668261 668438 669043 669044 672122 672123 672153 672804 672854 673066 673070 673788 673792 673954 675581 675921 676764 677032~~
~~677957 678128 678141 678455 678546 679461 681326 683470 683738 683966 684525 684922 684940 684947 690285 690292 690308 690376 690378 690645 690650 695896 696109 696268 696492 697279 698123 698581 698902 698944 699201 699674 699682 700464 700471 700480 700501 700792 700799~~
~~701222 701224 701227 701239 701244 701247 701970 702426 705879 706348 706710 706997 707337 707384 707816 708741 709863 709909 709929 710317 710322 710438 712379 713957 714663 714690 715356 715387 715400 715682 715750 716013 716115 716713 716733 717497 717716 718053 718076 718823 721935~~
~~721939 722598 724467 724579 724654 724702 724784 724788 724798 726799 727223 727921 728033 728342 728506 728509 728722 729364 729793 730152 731724 731745 732758 732759 732763 732852 733248 733255 733863 733866 735161 735316 735869 735900 735936 735957 736594 736609 736679 736742 736747~~
~~736807 737242 737251 737573 737737 738034 738117 738496 738537 738841 738846 739402 740595 740654 741110 742003 742094 742606 743000 743301 743315 743423 743425 743480 744266 745360 746006 746403 746791 747554 748071 748212 748568 748958 750307 750449 752205 755104 755639 755813~~
~~755832 755916 756659 756732 756851 756855 756918 757149 757304 757530 758164 758398 758408 758841 759880 759904 761396 762324 763313 764508 764792 765480 767660 767665 767667 767679 768214 768313 768732 769192 769433 769499 769987 770089 770762 770952 770954 771027 771242 771398 771946~~
~~772742 772770 773108 773153 773927 775807 776188 776191 776314 776315 776317 776332 776353 776359 776361 776687 776748 777776 777834 777992 778557 779025 779124 779328 779850 780003 780027 780274 780288 780405 780451 780520 780712 780936 781022 781071 781343 781364 781660 782103 783421~~
~~783441 783590 783923 783924 783978 784639 785089 785094 785305 785576 785776 785824 787309 787703 787709 787848 787861 787864 788356 788362 788364 788701 789107 789300 789342 789647 789735 790424 791157 791445 791814 792166 792234 792588 793805 794286 794494 795574 795745 797163~~
~~797185 797476 797692 798589 798668 798670 798819 798823 799185 799785 800179 801831 803332 805300 805747 806522 806663 807035 807047 807859 807863 807865 807913 808023 808067 808140 808478 808481 808483 811606 811612 811616 812235 812400 817002 817365 819610 819611 819635 819794~~
~~819797 819865 820186 820215 821013 821470 821551 821931 822170 822171 822540 822858 822938 823715 824321 824856 825326 825379 825382 825705 825716 826031 826124 826581 826588 827082 827659 827821 827882 828019 829795 829798 829813 829821 830049 830508 830544 831055 831087 831092~~
~~831424 831658 831846 832103 832197 832204 835496 835499 836563 836601 836623 836774 837418 839758 842884 842940 843429 843444 843811 843886 843985 844305 844364 844383 845023 846984 847678 848747 849014 849398 849456 850061 850080 850564 850949 850955 851635 851673 851756~~
~~852016 853154 854021 854034 854050 854052 854137 854157 854254 854788 854807 855236 855536 855960 857836 858852 859008 859255 860060 861419 861439 862228 862699 862708 863084 863961 864099 864101 864227 864462 865024 865471 865507 866064 866611 867482 867749 867753 867767~~
~~867946 868189 868206 868731 868890 870328 871848 872769 873660 873718 874687 875476 875530 875656 875777 876226 877378 877381 877986 878038 878293 878417 878429 878433 878435 878444 878495 878505 878520 879096 879132 879647 880091 880512 880591 880776 880840 881444 881461 881470~~
~~881608 882008 882012 882323 882416 882486 882514 882565 882843 882933 882974 883395 883472 883490 883623 883626 883630 884114 884369 884630 884920 885017 885034 885067 885103 885219 885276 885607 885648 885769 885770 885885 885887 885934 885976 885986 885988 886094 886101~~
~~886102 886104 886144 886243 886246 886248 886277 886285 886287 886630 886909 887459 887521 887542 887544 887549 887556 887921 887989 888002 888106 888470 888543 888555 888568 888618 888837 888898 889132 889186 889682 890048 890414 890465 891156 891773 891775 892291 892714~~
~~892787 893263 893364 893368 893679 893684 893732 894727 894781 894782 894786 894794 894797 895782 895792 896126 896154 897202 897492 897747 898622 898695 899510 900395 900405 900681 900683 901313 902227 902253 902706 902744 903028 903041 905396 905947 905989 906229 906233~~
~~906236 906241 906242 906243 906244 906284 906285 906286 906858 906885 907085 908472 908608 908813 908867 908915 908920 908939 908948 909441 909447 909574 909586 909599 909601 909602 909743 910012 910477 910929 911368 911369 911370 911707 911708 912152 912303 912304 912316~~
~~912328 912379 912734 912813 913224 913261 913272 913376 913715 913716 913740 913867 913876 913883 913885 913977 913978 914341 914511 914898 915171 915903 916039 916511 916712 919140 923390 923765 923860 923867 923892 924660 924690 925777 926847 927389 928426 928450 928625 929280 931496~~
~~933104 934427 934914 936358 936361 936403 936737 936854 937550 938431 939015 939472 939499 939868 939893 940025 940088 940246 940635 940638 940642 940846 940864 941381 941877 941905 942027 942258 942376 942379 942390 942480 942496 942547 942549 942550 942604 944153 944266~~
~~944278 944321 944963 944972 944975 945453 945512 945860 946072 946478 946679 946969 947070 947661 947680 949807 950438 951285 951517 951527 951528 951573 951957 951979 951988 952022 952381 952409 952780 952810 952818 952819 952823 952913 952984 952994 953336 955822 955850 956156~~
~~956166 956173 957004 957028 958381 958432 960071 961494 963641 969133 969203 969702 969705 969769 969778 970001 970643 971385 971426 973118 974751 975110 975138 975182 975290 975335 975626 975636 975947 975959 975961 976697 976889 977431 978456 978714 980013 980119 980126 981314~~
~~981325 983558 983560 983840 984766 986313 986678 986864 988706 988719 989166 990247 990807 991510 994281 994406 994957 995564 995657 995673 995675 995679 995704 995816 995817 995826 996881 996895 997546 998059 998262 998577 998580 998709 999759 999790 999849 999857 1000605~~
~~1000606 1001222 1001547 1001569 1003694 1005458 1005590 1006301 1006870 1006910 1007027 1007187 1007213 1007512 1008106 1008613 1008636 1008818 1009788 1009838 1011283 1011730 1011745 1011781 1013056 1014972 1014973 1015498 1015656 1015766 1016137 1019034 1019322 1020517 1020800~~
~~1022081 1022232 1022321 1022948 1023703 1023753 1024444 1024756 1024786 1025587 1027359 1029440 1029910 1030014 1030460 1032067 1032086 1033113 1033115 1033873 1033946 1034280 1034349 1034383 1034400 1035371 1037313 1037657 1037665 1038396 1038590 1039207 1041079 1041746 1041978~~
~~1042567 1050094 1053074 1053431 1053676 1053683 1053692 1054512 1054531 1054538 1054541 1054545 1054568 1054601 1054753 1055762 1055864 1057248 1058860 1059459 1059606 1062612 1063653 1064391 1065883 1066414 1066496 1067805 1067850 1068451 1068458 1070460 1070462 1070464 1070465~~
~~1070638 1071799 1071839 1071879 1072677 1072691 1073702 1073910 1073911 1073919 1073928 1073933 1073934 1074305 1075242 1075266 1075546 1076026 1076091 1076283 1077031 1077949 1079062 1080438 1080462 1080991 1081850 1083681 1084194 1085298 1085464 1085824 1086842 1089761 1090037~~
~~1090096 1090424 1091757 1092833 1092947 1094052 1094616 1095870 1096138 1096789 1097585 1098132 1098961 1098963 1099080 1099216 1099224 1100080 1100083 1100123 1100129 1100202 1100237 1100316 1100457 1100480 1100493 1100511 1101576 1101600 1102329 1103027 1103032 1103048 1103386~~
~~1103389 1103813 1103817 1104162 1105187 1105574 1105684 1106141 1106719 1107147 1108007 1108145 1108159 1108824 1108826 1108904 1109009 1109517 1109913 1110327 1111245 1111251 1111253 1111326 1111327 1111363 1111477 1113710 1113744 1113940 1114058 1114071 1115246 1115665 1115844 1115847~~
~~1115853 1116103 1116306 1116646 1117235 1117240 1117255 1118996 1119579 1120063 1120151 1120603 1120677 1121083 1122246 1122334 1122335 1122338 1122344 1122361 1122401 1122402 1122403 1122768 1122833 1122839 1122886 1123011 1123120 1124036 1124421 1124448 1124468 1124480 1124485 1124563~~
~~1125561 1125658 1126032 1126066 1126072 1126105 1126507 1126518 1126555 1126562 1126754 1127091 1127167 1127581 1128061 1128094 1128108 1129088 1129314 1129676 1130604 1130640 1130679 1130698 1131267 1131846 1132290 1132390 1132396 1132584 1133143 1133354 1133389 1134298 1134515 1135047~~
~~1136542 1136551 1136597 1136806 1137610 1137616 1137624 1137998 1138141 1138265 1138740 1139264 1140084 1140196 1140890 1141329 1141338 1141379 1142331 1143106 1143194 1143216 1143805 1143847 1143878 1146644 1146696 1147144 1147216 1147655 1147662 1148883 1148963 1148970 1148973 1150837~~
~~1151155 1151269 1151401 1151634 1152623 1152635 1153057 1153153 1153458 1153498 1154971 1155807 1156190 1156532 1156992 1157577 1158569 1159039 1159899 1161346 1165794 1168666 1168667 1169460 1169639 1170355 1172150 1173529 1174372 1175010 1180608 1181336 1181354 1181796 1182711 1182865~~
~~1183375 1183423 1183448 1186226 1186271 1187123 1189980 1190002 1190147 1190272 1190727 1193543 1194022~~

Blocks: fuzz 495236
Show dependency tree / graph

GO FUZZ SOME STUFF