A Bug Hunter's Reflections on Fuzzing

Alexander Popov

Positive Technologies

25.05.2024

- Alexander Popov
- Linux kernel developer since 2012
- Open-source maintainer
- Principal security researcher at
  - positive technologies
- Speaker at conferences including:
  OffensiveCon, Nullcon Goa, Linux Security Summit, Still Hacking Anyway, Zer0Con, Positive Hack Days, ZeroNights, HighLoad++, Open Source Summit, OS Day and Linux Plumbers

a13xp0p0v.github.io/conference_talks

- Been thinking about this topic for several years
- Have wanted to structure these thoughts
- Giving a talk and creating a discussion

  is a great way to do that
- Haven't found any conference talks about that 🤔
- People tend to keep their know-how to themselves 😉
- So, let's do this!

https://t.me/learningnets

What is fuzzing?

Fuzzing is...

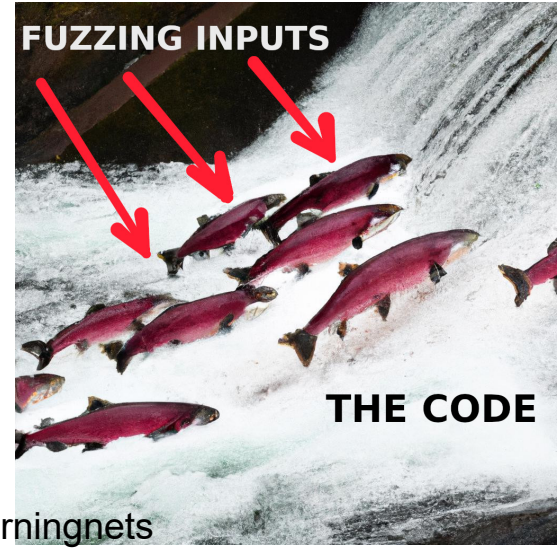
What is fuzzing?

Ha-ha! I'm joking,

everybody here already knows

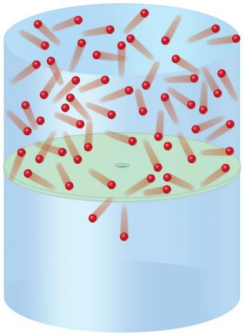
what fuzzing is.

I Think Fuzzing Is...



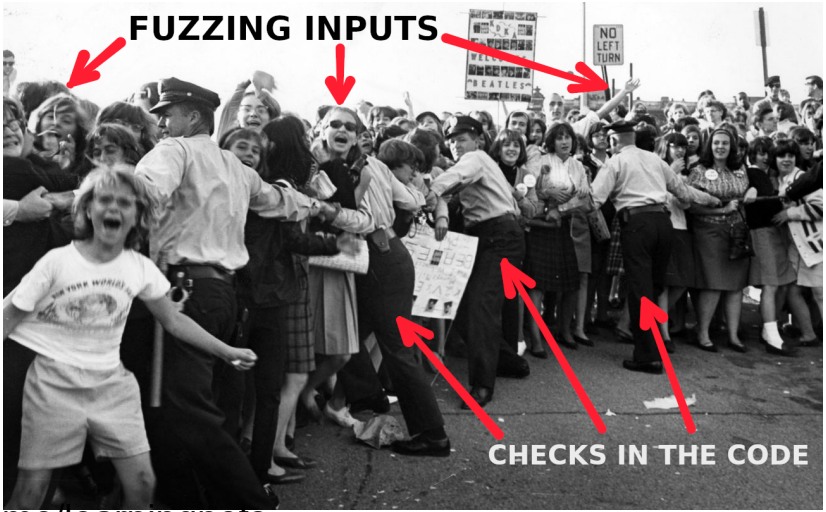
generated with DALL-E 2

Effusion

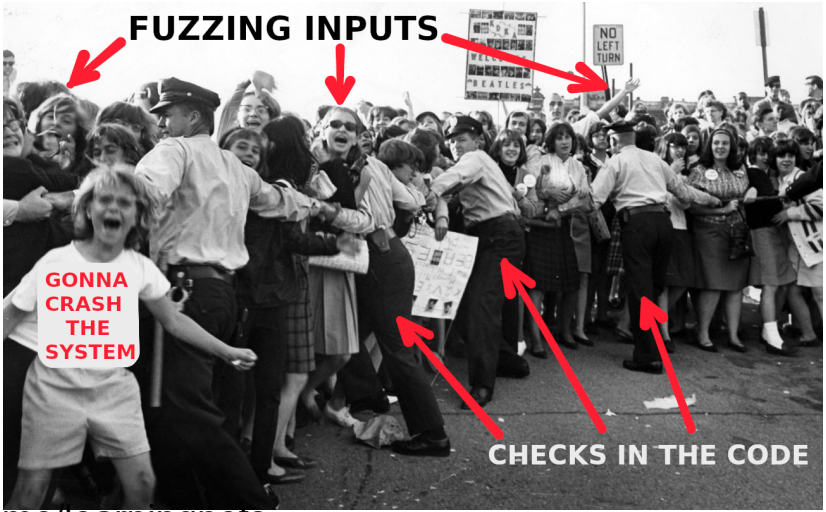


Effusion is the escape of gas molecules through a tiny hole into an evacuated space.

Chemistry, The Central Science, 11th edition



Police use human barricade to hold back Beatles fans (Dale Gleason/The Pittsburgh Press)



Police use fire hose in effort to hold back Beatles fans (Dale Gleason/The Pittsburgh Press)

Fuzzing is...

a great way to delegate

boring software testing to computers

(but you need to have control over it)

Software developer

1. Uses fuzzing to search for bugs
2. Usually interested in all bugs
3. Have access to the source code
4. Enables all available debug features

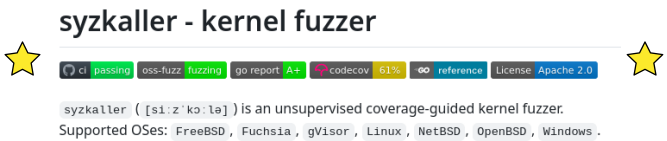
Security researcher

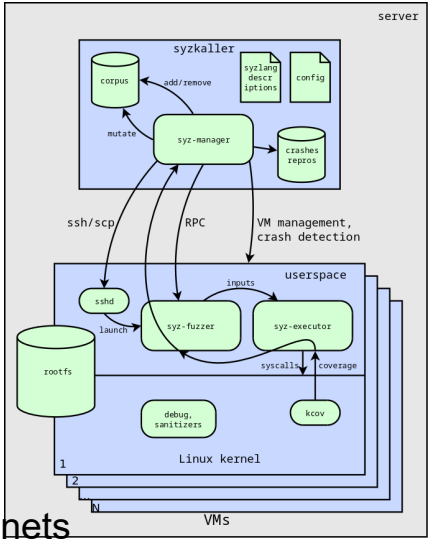
1. Uses fuzzing to discover vulnerabilities
2. Not interested in all bugs
3. Interested in vulnerabilities (bugs reachable via attack surface)
4. May not have access to the source code
5. More interested in bugs with stable reproducers
6. More interested in unique bugs

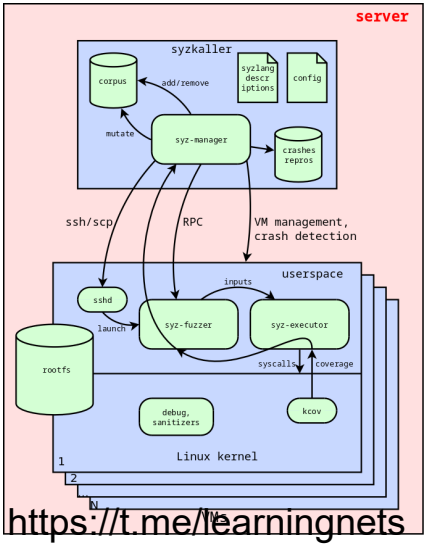
Question

What is special about fuzzing for vulnerability discovery?

- Security researchers usually don't discuss this...
- But today we will!
- As an example, I'll use my favorite kernel fuzzer, **syzkaller**

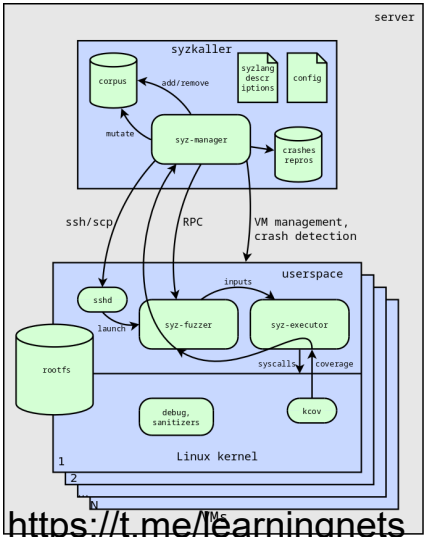






What about a server for syzkaller?

1. It needs hardware virtualization
2. Unusual characteristics for a server:
   - Number of CPU cores is crucial
   - RAM $\approx$ 4GB * (CPU_N / 2)
   - No huge hard drive needed unless tracing or snapshots are used
3. It can run on:
   - Dedicated server (it needs to be customized; otherwise, you'll overpay)
   - VPS with nested virtualization (not many options)

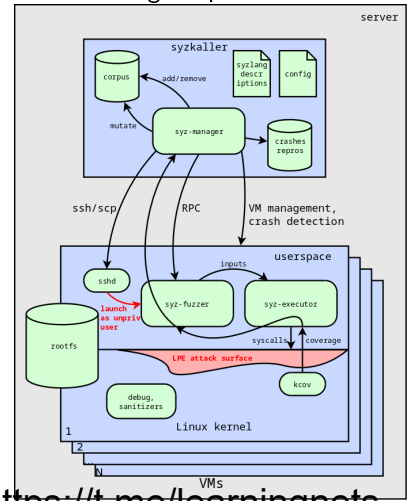


Adapt to vuln discovery

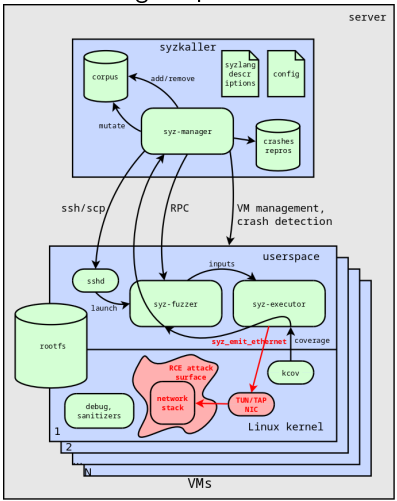
1. Not interested in all bugs, interested in vulnerabilities (bugs reachable via attack surface)
2. More interested in bugs with stable reproducers
3. More interested in unique bugs

1) Not Interested in All Bugs, Interested in Vulnerabilities

Fuzzing for potential LPE

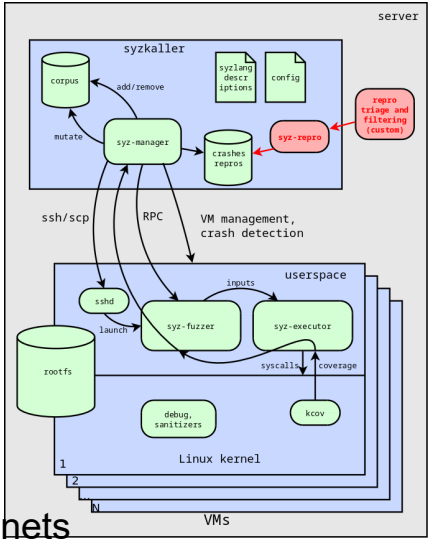


Fuzzing for potential RCE

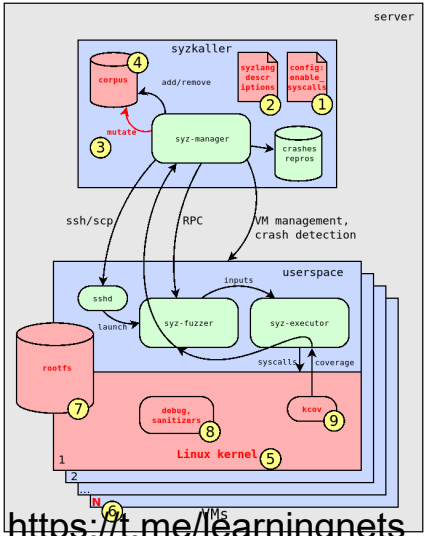


See: https://xairy.io/articles/syzkaller-external-network

2) More Interested in Bugs with Stable Reproducers



3) More Interested in Unique Bugs



Ideas on how to make a fuzzer find unique bugs

1. Limit enabled syscalls to make fuzzing go deeper
2. Write new syzlang descriptions
3. Change the mutation of fuzzing inputs (i.e. integrate symbolic execution)
4. Start fuzzing from the crafted corpus
5. Modify the Linux kernel (for example, my CVE-2021-26708)
6. Use more computing power than competitors
7. Modify the rootfs of fuzzing VMs (for example, my CVE-2017-2636)
8. Improve the Linux kernel bug detectors
9. Customize kcov or use cover_filter for directed fuzzing

- It's a wonderful research instrument
- It can be used not only for vuln discovery

  (for example, it's how I discovered msg_msg heap spraying)
- It's an everyday practice
- For unique findings, your fuzzing setup should be unique as well
- You need to be brave: you are risking your efforts and computing power *
- And that's why it's so exciting when you eventually find success!


Thank you!
Questions?
alex.popov@linux.com
a13xp0p0v
positive technologies
https://t.me/learningnets