USENIX WOOT'26 Artifact Appendix: FuzzBT: Holistic-State-Guided Fuzzing for Bluetooth Host Stack in Kernels

Sungwoo Kim, *Purdue University*
Hui Peng, *Google, Inc.*
Imtiaz Karim, *The University of Texas at Dallas*
Ruoyu Wu, *Purdue University*
Jianliang Wu, *Simon Fraser University*
Elisa Bertino, *Purdue University*
Mathias Payer, *EPFL*
Dave (Jing) Tian, *Purdue University*

A Artifact Appendix

A.1 Abstract

This document describes the artifact of the FuzzBT paper and claims the available and functional badges.

A.2 Major Claims

(C1) Artifact available: FuzzBT's source code is available at Zenodo (https://doi.org/10.5281/zenodo.20076507).
(C2) Artifact functional: FuzzBT works for the most recent mainline Linux (v7.1-rc1) as of the writing of the description.

A.3 Evaluation Workflow

A.3.1 Supported environments

- x86_64 machine with KVM enabled.
- 32GB of RAM.
- 100GB of free disk space.
- Docker Engine, version 29.4.1.

A.3.2 Experimental setup

The evaluation setup will build many components from scratch, including the Linux kernel, Debian image, QEMU, and Syzkaller.

0. Download and decompress the most recent artifact from Zenodo:
https://doi.org/10.5281/zenodo.20076507

1. Set up a Docker container

```
cd workspace
docker build --build-arg UID=$(id -u)
 ↪  --build-arg GID=$(id -g) -t kbuild .
```

2. Start the container.

```
docker run -it --rm -v $(pwd):/workspace
 ↪  kbuild
```

3. (inside the container) Build a BTSCov and Linux kernel

```
cd /workspace/configs
bash build.sh
cd /workspace/btscov
mkdir build && cd build
cmake -DLLVM_DIR=$(llvm-config --cmakedir) ..
make -j
cd /workspace
git clone -b v7.1-rc1 --depth=1
 ↪  git://git.kernel.org/pub/scm/linux/kernel/
 ↪  git/torvalds/linux.git
cd linux
git apply --whitespace=fix
 ↪  ../../patches/0001-introduce-btscov.patch
cp ../patches/.config .
make LLVM=1 KCFLAGS+=" -w" -j
```

4. (inside the container) Build a Syzkaller

```
cd /workspace
git clone https://github.com/google/syzkaller
cd syzkaller
git checkout
 ↪  efb3e894dd7e30475c92970989c58e5aad3d5e7b
git apply --whitespace=fix
 ↪  ../patches/syzkaller.patch
make -j
```

5. (inside the container) Build a QEMU

```
cd /workspace
git clone --depth=1 -b v10.1.5
 ↪  https://gitlab.com/qemu-project/qemu.git
cd qemu
./configure --target-list="x86_64-softmmu"
 ↪  --enable-tools
ninja -C build
```

6. (**outside** the container) Prepare a linux image

```
# pwd should be <artifact>/workspace
mkdir image
cd image
../syzkaller/tools/create-image.sh -s 8192
```

7. (**outside** the container) Run the image

```
# pwd should be
↪ <artifact>/workspace/image
../qemu/build/qemu-system-x86_64 \
  -L ../qemu/pc-bios \
        -m 2G \
        -smp 2 \
        -kernel ../linux/arch/x86/boot/bzImage
        ↪ \
        -append "console=ttyS0 root=/dev/sda
        ↪ earlyprintk=serial net.ifnames=0" \
        -drive file=./trixie.img,format=raw \
        -net
        ↪ user,host=10.0.2.10,hostfwd=tcp:127.0.0.1:10021-:22
        ↪ \
        -net nic,model=e1000 \
        -enable-kvm \
        -nographic
```

In step 8, we need to copy artifacts into the VM.

8. (**outside** the container) Setup image

```
# pwd should be
↪ <artifact>/workspace/configs
scp -o StrictHostKeyChecking=no -i
↪ ../image/trixie.id_rsa -P 10021 -r ./btprog
↪ root@localhost:/root/
ssh -o StrictHostKeyChecking=no -p 10021 -i
↪ ../image/trixie.id_rsa  root@localhost
apt install bluetooth
systemctl enable bluetooth
shutdown now
```

9. (**outside** the container) Copy configs

```
# pwd should be <artifact>
cp workspace/corpus/btssym.db
↪ workspace/syzkaller
cp workspace/minimum.json
↪ workspace/syzkaller/minimum.json
cp workspace/fuzzbt.json
↪ workspace/syzkaller/fuzzbt.json
```

A.3.3 Run evaluation

1. Minimum working example, 1 VM with 1 CPU and 2GB RAM:

outside the container, run:

```
cd workspace/syzkaller
./bin/syz-manager -config ./minimum.json
```

2. Run the evaluation with 4VMs with 4 CPUs and 8GB RAM for each:

outside the container, run:

```
cd workspace/syzkaller
./bin/syz-manager -config fuzzbt.json
```

It will serve the dashboard on `http://0.0.0.0:56720`.