









Using Ubuntu VM for Testing

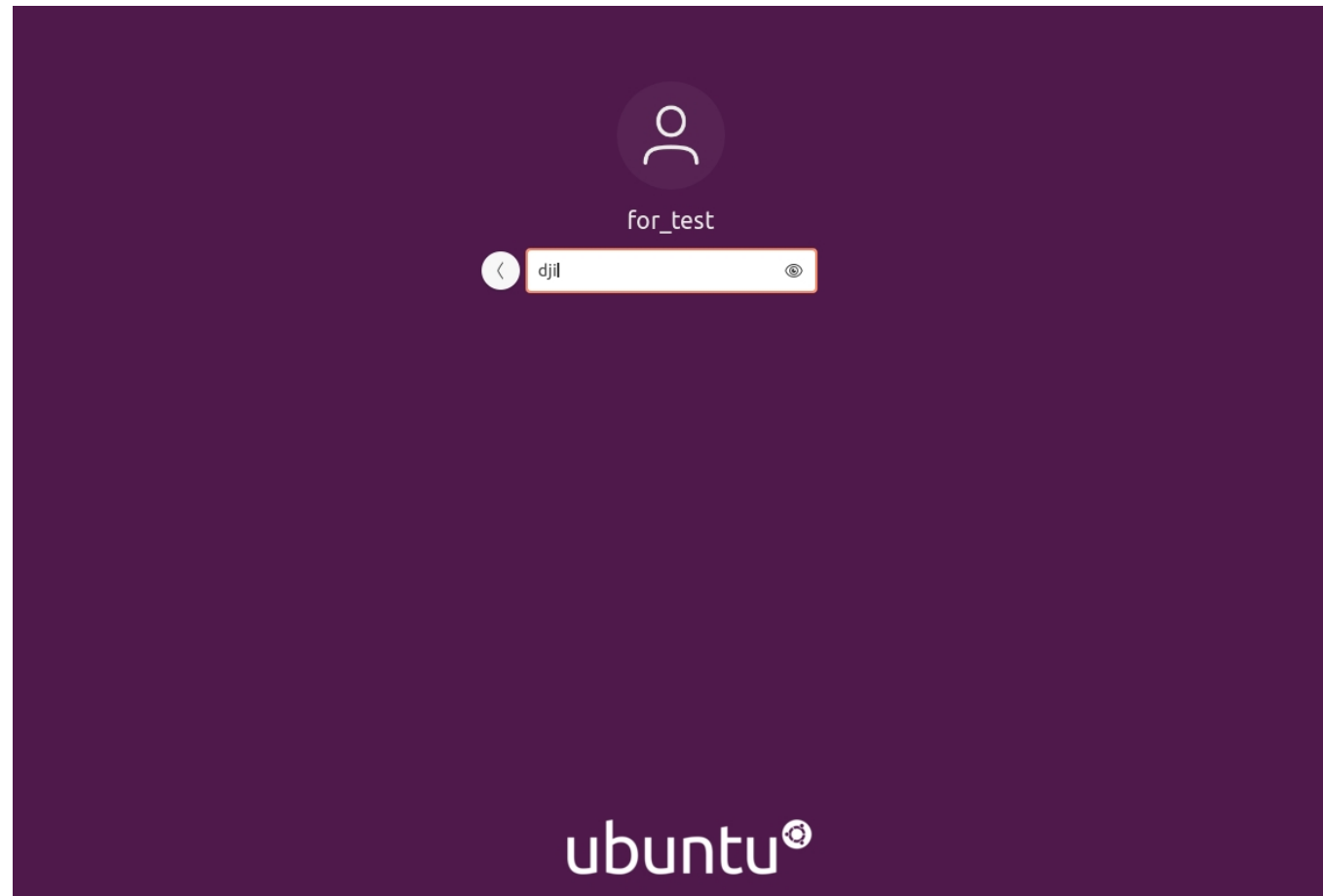
Download Ubuntu .OVF File

- Download link: [for_test_VM](#)
- Please download all of the files and put them into one folder.

	 Name ▾
	 for_test_ubuntu-disk1.vmdk
	 for_test_ubuntu.mf
	 for_test_ubuntu.ovf

Open Ubuntu VM

- Please Using Vmware Workstation 16-pro to open the Ubuntu .ovf file.
- If VM is loaded successfully, please login with password: **dji**



Create Dev.img

- Create dev.img for Filesystem mounting before testing.

```
test@ubuntu:~$  
test@ubuntu:~$ cd ~/f2fsj/test_dir/  
test@ubuntu:~/f2fsj/test_dir$  
test@ubuntu:~/f2fsj/test_dir$ ls  
test@ubuntu:~/f2fsj/test_dir$  
test@ubuntu:~/f2fsj/test_dir$ dd if=/dev/zero of=dev.img bs=1M count=16384  
16384+0 records in  
16384+0 records out  
17179869184 bytes (17 GB, 16 GiB) copied, 108.949 s, 158 MB/s  
test@ubuntu:~/f2fsj/test_dir$ ls  
dev.img  
test@ubuntu:~/f2fsj/test_dir$  
test@ubuntu:~/f2fsj/test_dir$
```

Run Setup Script

- Testing environment setting.

```
test@ubuntu:~$ cd f2fsj/filebench/script/
test@ubuntu:~/f2fsj/filebench/script$
test@ubuntu:~/f2fsj/filebench/script$
test@ubuntu:~/f2fsj/filebench/script$ ls
ext4_fb.sh  f2fs_fb.sh  f2fsj.ko  j_f2fs_fb.sh  perf.sh  setup.sh  xfs_fb.sh
test@ubuntu:~/f2fsj/filebench/script$
test@ubuntu:~/f2fsj/filebench/script$ sudo ./setup.sh
[sudo] password for test:
nodev  autofs
nodev  binfmt_misc
->>>>sudo -i to setup randomize_va_space
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
kernel.perf_event_paranoid = -1
vm.dirty_background_ratio = 5
vm.dirty_ratio = 50
vm.dirty_writeback_centisecs = 1500
```

Test Scripts Introduction

- At ~/f2fsj/filebench/script, there are four scripts for f2fsj/f2fs/ext4/xfs testing.

```
test@ubuntu:~/f2fsj/filebench/script$ ls
ext4_fb.sh  f2fs_fb.sh  j_f2fs_fb.sh  perf.sh  setup.sh  xfs_fb.sh
test@ubuntu:~/f2fsj/filebench/script$
test@ubuntu:~/f2fsj/filebench/script$
```

- Benchmarks.

- We use shell scripts to integrate the benchmarks.
- Using -h, then you can see the benchmarks for testing.
- Please note that due to limited size of dev.img, VM does not support big file benchmarks like create_64g/rread_64g/rwrite_64g.

```
test@ubuntu:~/f2fsj/filebench/script$ sudo ./j_f2fs_fb.sh -h
-----
META Only test cases:
./xx_fb.sh create_empty, create empty files
./xx_fb.sh unlink_empty, delete empty files
./xx_fb.sh mkdir, make directory
./xx_fb.sh rmdir, remove directory
./xx_fb.sh readdir, read directory
-----
DATA_And_META test cases:
./xx_fb.sh create_4k, create files with 4kb size
./xx_fb.sh create_32k, create files with 32kb size
./xx_fb.sh read_small, read 4M files(4k for each)
./xx_fb.sh copy_4k, copy files with 4kb size
./xx_fb.sh create_1g, create files with 1GB size
./xx_fb.sh delete_4k, delete files with 4kB size
./xx_fb.sh rread_64g, random read files with 64g size
./xx_fb.sh rwrite_64g, random write files with 64g size
./xx_fb.sh seqread_64g, sequence read files with 64g size
./xx_fb.sh seqwrite_64g, sequence write files with 64g size
./xx_fb.sh create_64g, create a file with 64g size
-----
./xx_fb.sh varmail, realistic workloads for varmail
./xx_fb.sh oltp, realistic workloads for oltp
./xx_fb.sh fileserver, realistic workloads for fileserver
./xx_fb.sh webproxy, realistic workloads for webproxy
./xx_fb.sh webserver, realistic workloads for webserver
-----
./xx_fb.sh test, just test script
-----
test@ubuntu:~/f2fsj/filebench/script$
```

Run Benchmarks

- Run benchmarks like “Sudo ./j_f2fs_fb.sh create 4k”, then you can see the output like following figure.

```
Info: Trin is enabled
Info: Segments per section = 1
Info: Sections per zone = 1
Info: sector size = 512
Info: total sectors = 33554432 (16384 MB)
Info: zone aligned segment blkaddr: 512
Info: format version with
  "Linux version 5.15.39 (test@ubuntu) (gcc (Ubuntu 9.4.0-1ubuntu1-20.04.2) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #2 SMP Mon Apr 28 19:54:05 PDT 2025"
Info: [/home/test/f2fs/test_dir/dev.ing] Discarding device
Info: Overprovision ratio = 1.570%
Info: Overprovision segments = 260 (GC reserved = 135)
Info: format successful
Format device with f2fs success
umount f2fs: no success
mount f2fs in /j_f2fs_mount_point
/home/test/f2fs/test_dir/dev.ing on /j_f2fs_mount_point type f2fs (rw,relatime,lazytime,background_gc=on,discard,no_heap,user_xattr,inline_xattr,acl,inline_data,inline_dentry,flush_merge,extent_cache,node=adaptive,active_logs=6,alloc_node=reuse,checkpoint_merge,fsync_mode=posix,discard_unit=block)

Filebench Version 1.5-alpha3
0.000: Allocated 1505MB of shared memory
0.016: Populating and pre-allocating filesets
0.938: bigfileset populated: 1000000 files, avg. dir. width = 100, avg. dir. depth = 3.0, 0 leafdirs, 3906.250MB total size
0.938: Removing bigfileset tree (if exists)
0.940: Pre-allocating directories in bigfileset tree
1.244: Pre-allocating files in bigfileset tree
4.021: Waiting for pre-allocation to finish (in case of a parallel pre-allocation)
4.021: Population and pre-allocation of filesets completed
4.021: Starting 1 filecreate instances
5.024: Running...
147.103: Run took 142 seconds...
147.103: Per-Operation Breakdown
Finish      1000000ops    7038ops/s    0.0mb/s    0.0ms/op [0.00ms - 0.72ms]
closefile1  1000000ops    7038ops/s    0.0mb/s    0.0ms/op [0.00ms - 2.28ms]
writefile1  1000000ops    7038ops/s    27.5mb/s    0.1ms/op [0.00ms - 202.26ms]
createfile1 1000000ops    7038ops/s    0.0mb/s    0.0ms/op [0.00ms - 9033.23ms]
147.103: IO Summary: 3000000 ops 21115.056 ops/s 0/7038 rd/wr 27.5mb/s 0.1ms/op
147.103: Shutting down processes
umount f2fs in /j_f2fs_mount_point
rmmod f2fs in /j_f2fs_mount_point
Filebench test create_4k Over
test@ubuntu:~/f2fs/Filebench/scripts$
test@ubuntu:~/f2fs/Filebench/scripts$ sudo ./f2fs_fb.sh create_4k
Begin to filebench test

F2FS-tools: mkfs.f2fs Ver: 1.11.0 (2018-07-10)

Info: Disable heap-based policy
Info: Debug level = 0
Info: Label =
Info: Trin is enabled
Info: Segments per section = 1
Info: Sections per zone = 1
Info: sector size = 512
Info: total sectors = 33554432 (16384 MB)
Info: zone aligned segment blkaddr: 512
Info: format version with
  "Linux version 5.15.39 (test@ubuntu) (gcc (Ubuntu 9.4.0-1ubuntu1-20.04.2) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #2 SMP Mon Apr 28 19:54:05 PDT 2025"
Info: [/home/test/f2fs/test_dir/dev.ing] Discarding device
Info: Overprovision ratio = 1.570%
Info: Overprovision segments = 260 (GC reserved = 135)
Info: format successful
Format device with f2fs success
umount f2fs: no success
mount f2fs in /f2fs_mount_point
/home/test/f2fs/test_dir/dev.ing on /f2fs_mount_point type f2fs (rw,relatime,lazytime,background_gc=on,discard,no_heap,user_xattr,inline_xattr,acl,inline_data,inline_dentry,flush_merge,extent_cache,node=adaptive,active_logs=6,alloc_node=reuse,checkpoint_merge,fsync_mode=posix,discard_unit=block)
Tue 29 Apr 2025 10:45:20 PM PDT
Filebench Version 1.5-alpha3
0.000: Allocated 1505MB of shared memory
0.012: Populating and pre-allocating filesets
1.230: bigfileset populated: 1000000 files, avg. dir. width = 100, avg. dir. depth = 3.0, 0 leafdirs, 3906.250MB total size
1.230: Removing bigfileset tree (if exists)
1.373: Pre-allocating directories in bigfileset tree
1.714: Pre-allocating files in bigfileset tree
5.265: Waiting for pre-allocation to finish (in case of a parallel pre-allocation)
5.265: Population and pre-allocation of filesets completed
5.265: Starting 1 filecreate instances
9.268: Running...
233.857: Run took 227 seconds...
233.859: Per-Operation Breakdown
Finish      1000000ops    4394ops/s    0.0mb/s    0.0ms/op [0.00ms - 0.14ms]
closefile1  1000000ops    4394ops/s    0.0mb/s    0.0ms/op [0.00ms - 1.71ms]
writefile1  1000000ops    4394ops/s    17.2mb/s    0.0ms/op [0.00ms - 3096.80ms]
createfile1 1000000ops    4394ops/s    0.0mb/s    0.2ms/op [0.01ms - 4486.42ms]
233.860: IO Summary: 3000000 ops 13101.696 ops/s 0/4394 rd/wr 17.2mb/s 0.2ms/op
233.860: Shutting down processes
Begin to umount, Tue 29 Apr 2025 10:49:15 PM PDT
umount f2fs in /f2fs_mount_point
Filebench test create_4k Over
Tue 29 Apr 2025 10:49:18 PM PDT
test@ubuntu:~/f2fs/Filebench/scripts$
```

Code Compile

- Also, we provide compile script to build f2fsj.
- Please goto ~/f2fsj/f2fsj; Then using “./script/build.sh f2fsj” to build.

```
test@ubuntu:~/f2fsj/test_dir$ cd ../f2fsj/
test@ubuntu:~/f2fsj/f2fsj$
test@ubuntu:~/f2fsj/f2fsj$
test@ubuntu:~/f2fsj/f2fsj$ ls
acl.c  checkpoint.c  debug.c      f2fs.h  gc.h      inode.c  j_checkpoint.c  j_epoch_commit.c  j_epoch_process.c  j_journal_file.h  j_log.h      j_recovery.c  Makefile  node.h  segment.c  super.c  xattr.c
acl.h  compress.c    dir.c        file.c  hash.c    iostat.c  j_checkpoint.h  j_epoch_commit.h  j_epoch_process.h  j_log_basic.h    j_log_operate.c  j_recovery.h  namel.c  recovery.c  segment.h  sysfs.c  xattr.h
Makefile  data.c      extent_cache.c  gc.c    inline.c  iostat.h  j_epoch.c      j_epoch.h        j_journal_file.c  j_log_content.h  j_log_operate.h  Kconfig    node.c  setup.h    shrinker.c  verity.c
test@ubuntu:~/f2fsj/f2fsj$
test@ubuntu:~/f2fsj/f2fsj$ ./script/build.sh f2fsj
make -C /lib/modules/5.15.39/build M=/home/test/f2fsj/f2fsj modules
make[1]: Entering directory '/home/test/change_kernel/linux-5.15.39'
CC [M] /home/test/f2fsj/f2fsj/dir.o
CC [M] /home/test/f2fsj/f2fsj/file.o
CC [M] /home/test/f2fsj/f2fsj/inode.o
CC [M] /home/test/f2fsj/f2fsj/namel.o
CC [M] /home/test/f2fsj/f2fsj/hash.o
CC [M] /home/test/f2fsj/f2fsj/super.o
CC [M] /home/test/f2fsj/f2fsj/inline.o
CC [M] /home/test/f2fsj/f2fsj/checkpoint.o
CC [M] /home/test/f2fsj/f2fsj/gc.o
CC [M] /home/test/f2fsj/f2fsj/data.o
CC [M] /home/test/f2fsj/f2fsj/node.o
CC [M] /home/test/f2fsj/f2fsj/segment.o
CC [M] /home/test/f2fsj/f2fsj/recovery.o
CC [M] /home/test/f2fsj/f2fsj/shrinker.o
CC [M] /home/test/f2fsj/f2fsj/extent_cache.o
CC [M] /home/test/f2fsj/f2fsj/sysfs.o
CC [M] /home/test/f2fsj/f2fsj/j_log_operate.o
CC [M] /home/test/f2fsj/f2fsj/j_epoch_commit.o
CC [M] /home/test/f2fsj/f2fsj/j_checkpoint.o
CC [M] /home/test/f2fsj/f2fsj/j_epoch.o
CC [M] /home/test/f2fsj/f2fsj/j_journal_file.o
CC [M] /home/test/f2fsj/f2fsj/j_recovery.o
CC [M] /home/test/f2fsj/f2fsj/j_epoch_process.o
CC [M] /home/test/f2fsj/f2fsj/debug.o
CC [M] /home/test/f2fsj/f2fsj/xattr.o
CC [M] /home/test/f2fsj/f2fsj/acl.o
CC [M] /home/test/f2fsj/f2fsj/verity.o
CC [M] /home/test/f2fsj/f2fsj/compress.o
LD [M] /home/test/f2fsj/f2fsj/f2fsj.o
MODPOST /home/test/f2fsj/f2fsj/Module.symvers
CC [M] /home/test/f2fsj/f2fsj/f2fsj.mod.o
LD [M] /home/test/f2fsj/f2fsj/f2fsj.ko
make[1]: Leaving directory '/home/test/change_kernel/linux-5.15.39'
make -C /lib/modules/5.15.39/build M=/home/test/f2fsj/f2fsj modules_install INSTALL_MOD_PATH=/home/test/f2fsj/f2fsj/build
make[1]: Entering directory '/home/test/change_kernel/linux-5.15.39'
arch/x86/Makefile:142: CONFIG_X86_X32 enabled but no binutils support
INSTALL /home/test/f2fsj/f2fsj/build/lib/modules/5.15.39/extra/f2fsj.ko
SIGN /home/test/f2fsj/f2fsj/build/lib/modules/5.15.39/extra/f2fsj.ko
DEPMOD /home/test/f2fsj/f2fsj/build/lib/modules/5.15.39
Warning: modules_install: missing 'system.map' file. Skipping depmod.
make[1]: Leaving directory '/home/test/change_kernel/linux-5.15.39'
no file compile success
[sudo] password for test:
make -C /lib/modules/5.15.39/build M=/home/test/f2fsj/f2fsj clean
make[1]: Entering directory '/home/test/change_kernel/linux-5.15.39'
CLEAN /home/test/f2fsj/f2fsj/Module.symvers
make[1]: Leaving directory '/home/test/change_kernel/linux-5.15.39'
rm -f *-~ /home/test/f2fsj/f2fsj/*-ur-safe
make clean over
*****
+      make f2fsj success      +
+                               +
*****
Wed 30 Apr 2025 06:24:18 PM PDT
```


Problems that may be encountered

- Please use Vmware Workstation 16 pro to load VM because other versions may lead to unsuccessful import.
- The hardware configuration of the virtual machine is 8 CPUs and 8GB of memory. You can adjust it according to your actual situation.
- The running results in the screenshot are for reference only, as the results may vary depending on different hardware configurations.
- In order to adapt to the environment described in the article, we have replaced the kernel of Ubuntu 20.04.3 with version 5.15.39, which is not from the Ubuntu distribution. Considering the uncertainties in the environment transplantation, if you encounter any problems during use, please feel free to contact us!