Linux Kernel Functional Testing (LKFT)

- Architectures: arm32, arm64, i386, x86_64
- Hardware: X15, DragonBoard 410c, Juno, HiKey, x86_64 servers,
- QEMU: x86* on x86_64 servers, arm* on SynQuacer arm64 hosts
- Linux Branches:
  - LTS: 4.4, 4.9, 4.14, 4.19
  - Latest stable (5.2), mainline, next
- Tests: LTP, libhugetlbfs, perf, v4l2, kvm-unit-tests, s-suite (i/o benchmark), kselftests
- All tests run in all environments on every push for a total of ~25,000 tests per push.
- Authors - Dan Rue and Anders Roxell
### Kselftest Version

<table>
<thead>
<tr>
<th>Linux Version</th>
<th>Kselftest Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>next</td>
<td>next</td>
</tr>
<tr>
<td>mainline</td>
<td>mainline</td>
</tr>
<tr>
<td>4.4.x, 4.9.x, 4.14.x, 4.19.x</td>
<td>v5.2.11 (latest stable release)</td>
</tr>
</tbody>
</table>

### Selected results

<table>
<thead>
<tr>
<th>Board</th>
<th>v5.3-rc7, in-kernel selftest</th>
<th>v4.9.190, v5.1 selftest</th>
<th>v4.9.190, in-kernel selftest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pass</td>
<td>Skip</td>
<td>Fail</td>
</tr>
<tr>
<td>qemu_x86_64</td>
<td>126</td>
<td>17</td>
<td>55</td>
</tr>
<tr>
<td>x86_64</td>
<td>129</td>
<td>13</td>
<td>57</td>
</tr>
<tr>
<td>qemu_i386</td>
<td>115</td>
<td>16</td>
<td>65</td>
</tr>
<tr>
<td>i386</td>
<td>84</td>
<td>13</td>
<td>57</td>
</tr>
<tr>
<td>qemu_arm64</td>
<td>102</td>
<td>15</td>
<td>63</td>
</tr>
<tr>
<td>juno-r2 - arm64</td>
<td>99</td>
<td>14</td>
<td>63</td>
</tr>
<tr>
<td>db410c - arm64</td>
<td>100</td>
<td>15</td>
<td>65</td>
</tr>
<tr>
<td>hikey - arm64</td>
<td>100</td>
<td>15</td>
<td>63</td>
</tr>
<tr>
<td>qemu_arm</td>
<td>94</td>
<td>14</td>
<td>70</td>
</tr>
<tr>
<td>x15 - arm</td>
<td>97</td>
<td>15</td>
<td>68</td>
</tr>
<tr>
<td>Total</td>
<td>1046</td>
<td>147</td>
<td>626</td>
</tr>
</tbody>
</table>
Kselftest use-cases ...

- Kernel and Kselftest rev matched
  - Linux kernel

- Kselftest from latest stable
  - Stable release

Install and run Kselftest on a target
Discussion agenda

- Kselftest use-case discussion.
  - Run latest stable kselftest on stable kernels
    - Proving to be problematic. Balance coverage vs. dealing with test bugs in failing to skip when they should.
    - See large number of skips – this doesn’t necessarily offer coverage. Hard to measure how much more coverage we get from this option.
      - Some areas such as networking and bpf are very active and often the reason for increased test failures.
      - All but bpf support latest stable kselftest on stable kernels. This is an exception and not a rule.
Results from stable release

• 5.3.rc8
  - qemu_x86_64 (198 total: 125p, 17skip, 21fail, 35 known fails) – 63% 8.5%skip **11%fail, 18%known fails**
  - qemu_arm64 (180 total: 102p, 15skip, 30fail, 33known fails) - 57%p, 8%skip, **17%, 18%known fails**
Results from stable release

- **5.2.14-rc1**
  - qemu_x86_64 (178 total: 129p, 12skip, 7fail, 30known fails) - 72%p, 6.7%skip, **3.9% fail, 16%known fails**
  - qemu_arm64 (168 total: 111p, 17skip, 8fail, 32known fails) - 66%p, 10%skip, **4.7% fail, 19%known fails**

- **4.19.72-rc1**
  - qemu_x86_64 (181 total: 117p, 16skip, 2fail, 46known fails) – 65%p, 8%skip, **1%fail, 25%known fails**
  - qemu_arm64 (168 total: 94p, 24skip, 1fail, 49known fails) – 55%p, 14%skip, **0%fail, 29%known fails**
Results from stable release

• 4.14.143-rc1
  – qemu_x86_64 (178 total: 90p, 19skip, 2fail, 67known fails) – 50%p, 10%skip, 1%fail, 37%known fails
  – qemu_arm64 (166 total: 72p, 24skip, 1fail, 69known fails) – 43%p, 14%skip, 0.6%fail, 41%known fails

• 4.9.192-rc1
  – qemu_x86_64 (179 total: 85p, 29skip, 3fail, 62known fails) – 47%p, 16%skip, 0.01%fail, 35%known fails
  – qemu_arm64 (165 total: 66p, 27skip, 1fail, 71known fails) – 40%p, 16%skip, 0%fail, 43%known fails
Next steps ...

- Improve process – find test bugs early before they make it into a release. *linux-next* reports during merge window.
- Need help and support from individual test maintainers to reduce overhead on CI ring admins in dealing with breakages.
- Reduce known fails and fails numbers.
Next steps ...

• Since my talk calling out ARM coverage as weak, several ARM tests are added. Still more work is needed.

• Continue to increase coverage. Driver coverage is weak.

• Improve framework. Clearly identify dependencies on other utilities and recommend install options. e.g x86 test 32-bit library dependency checks.

• Kbuild integration? – easier if kselftest gets moved to root level. Being under tools is making it difficult to add support. (patch flow to stable could become harder with the move) – Discussed and developers don’t want this feature.