Bleep bloop! A robot workshop.

Aaron Conole <aconole@redhat.com>
December 6, 2018
My presentation is a lot like yours, only more interesting ‘cause it involves robots

The talk

- What is the 0-day robot?
- How can I avoid being taunted by a machine?
- I want to mess with the robot, how do I do that?
- Community feedback

Don’t forget

Tweet about things here with tag #OVSCon
Hey ovs-dev, wanna test all patches?

0-day means testing before day 1
- Yes... I know... 0-day sometimes means something more ominous
- In this case, it’s before day 1 of the patch landing
- Almost 1000 "series" processed since June 2018

Simple machines that talk to other machines
- Idea is straight-forward, we have some mechanical tasks
- No real thought required - just do some basic stuff
- Email everyone on failure
- Informative, not blocking
Oh, I get it— make the robot do all the work.

All these patches, like tears in a rain
- Developers come up with a bunch of patches
- They send them via email to the mailing list
- Patchwork gets those emails and builds up series information
- The robot monitors patchwork and starts processing
95473936754455451545245455478764

Patchwork
Open vSwitch

Show patches with: State = Action Required | Archived = No | 186 patches

EXTERMINATE! EXTERMINATE!

Open the doors
How can I be so bad at everything I try and still be so great?

Directive 4

1. Check out master (or, for certain notation, the appropriate branch)
2. Apply the next patch in the series
3. Run checkpatch
4. Push to https://github.com/ovsrobot/ovs
   `./configure && make`
5. More patches? Goto 2
   `make distcheck` - we’ll get to it
Now tragedy! Ha ha ha ha, that’s funny.

If any of those fail... bleep bloop!

* Previous message: [ews-dev] [PATCH] MAINTAINERS. Move Ethan to emeritus status.

* Messages sorted by: [date] [thread] [subject] [author]

### Build
reading sources... [96%] tutorials/servlet
reading sources... [96%] tutorials/xm individual
reading sources... [96%] tutorials/jsp
reading sources... [96%] tutorials/xdoclet
reading sources... [96%] tutorials/ewi
reading sources... [96%] tutorials/ews-advanced
reading sources... [96%] tutorials/ews-advanced

### Warning, treated as error:
MAINTAINERS: rhist: ERMA: Error processing content block for the "list-table" directive: uniform two-level bullet list expected, but row 2 does not contain the same number of items as row 1 (3 vs 2).

```
.. list-table:: EWS Emeritus Maintainers
   :header-rows: 1
   :align: left
   :widths: 3

   * - Name             
     * - Ethan J Jackson
     * - ej at eecs.berkeley.edu

make[2]: *** [doc-check] Error 1
make[2]: Leaving directory `/usr/lib/jenkins/jobs/apsnem_build_frew_jw/Workspace'
make[1]: *** [all] reuested Error 1
make[1]: Leaving directory `/usr/lib/jenkins/jobs/apsnem_build_frew_jw/Workspace'
makem: *** [all] Error 2
```

Please check this out. If you feel there has been an error, please email <EMAIL> at byhat.org

Thanks,
6-day Robot

* Previous message: [ews-dev] [PATCH] MAINTAINERS. Move Ethan to emeritus status.
Shut up and get to the point!

The logic is undeniable, prevention of bleep bloops is easy:

1. Run checkpatch (Install it as git hook! Then it just happens!)
2. Run a `make distcheck` for your series
3. Make sure that each patch builds individually (helps with bisecting)
4. Make sure each patch will apply to the branch cleanly
5. If you use github, push to a branch, and monitor the travis output.
Now, no one can say I don’t own a checkpatch hook

Attractive! Nice software

```bash
12:28:40 aconole ~$ cat git/ovs/.git/hooks/pre-commit
#!/bin/sh
if git rev-parse --verify HEAD 2>/dev/null then
    git diff-index -p --cached HEAD
else
    :
fi | utilities/checkpatch.py -S
```
Here I am with a brain the size of a planet

Check your travis branch status

- go to https://patchwork.ozlabs.org/project/openvswitch/list/
- Look up the series ID for your patches
- Find them on travis https://travis-ci.org/ovsrobot/ovs/branches

Better monitoring / reporting could be a welcome improvement

- Right now, the branch lookup is unwieldy
- Probably need some effort to do cleanups in this area
- Remember, the will accomplishes everything
Test my shiny, metal patch

I could see your patch move
Are you familiar with the old robot saying, DOES NOT COMPUTE?

Why doesn’t make distcheck get reported?

- Some of the tests in the make-check testsuite are unreliable
- Specifically, the most recently failing one I’ve seen is "ovn – ensure one gw controller restart in HA doesn’t bounce the master"
- Could be more… could be an artifact of the system
Yeah, well I’m gonna go build my own CI, with blackjack

Okay, maybe you want a robot of your very own
Who wouldn’t?

Or you want to be as lazy as you can be
- By letting a robot on your laptop test your patches
- Because again, who wouldn’t?

Either way, the door opened and you got in
- With https://github.com/orgcandman/pw-ci
- You too can launch a robot
Give you a fresh start. All we ask is your cooperation...

Let’s call him clamps

1. Clone the repo (https://github.com/orgcandman/pw-ci)
2. cd pw-ci/3rd-party/openvswitch/
3. Run ./build.sh
4. Set up jenkinsrc and pwmonrc files
5. Import the config (this is the main job)
6. Optionally, set up another jenkins job to run pwmon and cisubmit
We launch at six bells
Now, all I have to do is spell check it

patchwork reporting

- Patchwork supports 'check' reporting
- These are basically, test name, test status, link to extra data
- Could be useful for having an all-in-one spot to push CI status.
- 'Easy' to report (can be done with curl)
I better use my special robot vision to see what’s inside

A view of checks

[<V3,5/5> arch/powerpc/mm/hugetlb: NestMMU

Message ID: 20181205030931.12037-6-aneesh.kumar@linux.ibm.com
State: New
Headers: show
Series: NestMMU pte upgrade workaround for mprotect
Related: show

Checks

<table>
<thead>
<tr>
<th>Context</th>
<th>Check</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>snowpatch_ozlabs/checkpatch</td>
<td>warning</td>
<td>total: 0 errors, 0 warnings; 4 checks, 74 lines checked</td>
</tr>
<tr>
<td>snowpatch_ozlabs/build-pmac32</td>
<td>success</td>
<td>build succeeded &amp; removed 0 sparse warning(s)</td>
</tr>
</tbody>
</table>
That was the other guy. My name’s boiler

Partnering with other groups

- UNH interoperability lab currently runs tests for DPDK
- Working with Jeremy Plsek to incorporate some NFV and ovs-dpdk testing
- UNH IOL currently does performance testing and is building a public dashboard
I need your nics, your machines, and your patches

**Send hardware!**
- Mellanox is the first up to ship some HW that we’ll incorporate
- Other famous hardware makers who will remain unnamed have said …
- Anyone else? Takers?

**Send patches**
- Want to enable lots of the check-suites: check-dpdk, check-userspace, check-kmod
- Most need some kind of privileged resource.
- Maybe you have a fix?
Questions?