Continuous Integration of The FreeBSD Project

Li-Wen Hsu <lwhsu@FreeBSD.org>
https://lwhsu.org
Something about Me

- 許立文
- From Taiwan, a.k.a. Formosa
- System Administrator
- Instructor
- Programmer
- Traveler
- Diver
- Snowborder
- Cyclist

https://flightaware.com/live/flight/ANA852
A billion dollars + Linus <= good people with a rigorous engineering process doing BSD

- Karl Lehenbauer, FlightAware
We Need a (more) Higher Quality FreeBSD

- A higher quality Releasee
- Buildable -CURRENT branch
- Reliable -STABLE branch
- Faster development
Test Early, Test Often
Integration Test is Important

We need to do it continuously.

https://media.giphy.com/media/l0MYSpvx4pnsoMNz2/giphy.gif
Continuous Integration (CI)

Grady Booch, 1991

A software engineering practice of merging all developer working copies to a shared mainline several times a day

- To know conflicts earlier
- To know bugs earlier
CI Best Practices

- Maintain a code repository
- Automate the build
- Make the build self-testing
- Everyone commits to the baseline every day
- Every commit (to baseline) should be built
- Keep the build fast
- Test in a clone of the production environment
- Make it easy to get the latest deliverables
- Everyone can see the results of the latest build
- Automate deployment

https://en.wikipedia.org/wiki/Continuous_integration
We Do This from Begining

- Maintain a code repository ✓ (svn.FreeBSD.org)
- Automate the build
- Make the build self-testing
- Everyone commits to the baseline every day
- Every commit (to baseline) should be built
- Keep the build fast
- Test in a clone of the production environment
- Make it easy to get the latest deliverables
- Everyone can see the results of the latest build
- Automate deployment
We are an Open Source OS Project

- Maintain a code repository ✓ (svn.FreeBSD.org)
- Automate the build
- Make the build self-testing
  - Everyone commits to the baseline every day
- Every commit (to baseline) should be built
- Keep the build fast
- Test in a clone of the production environment
- Make it easy to get the latest deliverables
- Everyone can see the results of the latest build
- Automate deployment
What We Need

- Automate the build
- Make the build self-testing
- Every commit (to baseline) should be built
- Keep the build fast
- Make it easy to get the latest deliverables
- Everyone can see the results of the latest build
How We Do

A good CI system provides:

- Automate the build
- Make the build self-testing
- Make it easy to get the latest deliverables
- Everyone can see the results of the latest build

Needs help from everyone:

- Every commit (to baseline) should be built
- Keep the build fast
Automatic Build & Some History
2002~2014, Tinderbox

- By Dag-Erling Smørgrav <des@FreeBSD.org>
- http://tinderbox.freebsd.org
  - Probably started in March 2002
  - Probably stopped in September or October 2014
- Open Sourced
  - svn://svn.freebsd.org/base/user/des/tinderbox
- Well Documented
  - https://wiki.freebsd.org/Tinderbox
<table>
<thead>
<tr>
<th></th>
<th>amd64</th>
<th>arm</th>
<th>armv6 arm</th>
<th>i386</th>
<th>i386 pc98</th>
<th>ia64</th>
<th>mips</th>
<th>mips64 mips</th>
<th>powerpc</th>
<th>powerpc64 powerpc</th>
<th>sparc64</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12:53 UTC</td>
<td>10:19 UTC</td>
<td>13:46 UTC</td>
<td>13:24 UTC</td>
<td>09:13 UTC</td>
<td>n/a</td>
<td>08:54 UTC</td>
<td>07:04 UTC</td>
<td>10:27 UTC</td>
<td>10:23 UTC</td>
<td>07:20 UTC</td>
</tr>
<tr>
<td></td>
<td>13:57 UTC</td>
<td>07:47 UTC</td>
<td></td>
<td>04:59 UTC</td>
<td>08:27 UTC</td>
<td>08:08 UTC</td>
<td>06:14 UTC</td>
<td>n/a</td>
<td>09:15 UTC</td>
<td>09:25 UTC</td>
<td>07:22 UTC</td>
</tr>
<tr>
<td>stable/8</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>n/a</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
</tr>
<tr>
<td></td>
<td>14:01 UTC</td>
<td>14:01 UTC</td>
<td></td>
<td>14:01 UTC</td>
<td>14:01 UTC</td>
<td>14:01 UTC</td>
<td>14:01 UTC</td>
<td>n/a</td>
<td>14:01 UTC</td>
<td>14:01 UTC</td>
<td>14:01 UTC</td>
</tr>
<tr>
<td>stable/7</td>
<td>2014-09-17</td>
<td>n/a</td>
<td>n/a</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>2014-09-17</td>
<td>n/a</td>
<td>2014-09-17</td>
<td>14:00 UTC</td>
<td>14:00 UTC</td>
</tr>
<tr>
<td></td>
<td>14:00 UTC</td>
<td></td>
<td></td>
<td>14:00 UTC</td>
<td>14:00 UTC</td>
<td>14:00 UTC</td>
<td>n/a</td>
<td></td>
<td>14:00 UTC</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2013, jenkins-admin@FreeBSD.org

- Founded by Craig Rodrigues in late 2013
  - Current Members:
    - Ahmed Kamal, Baptiste Daroussin (bapt@), Craig Rodrigues (rodrigc@), Li-Wen Hsu (lwhsu@), Steve Wills (swills@)
    - https://www.freebsd.org/administration.html#t-jenkins-admin
  - Alumni:
    - Julio Merino (jmmv@), Steven Kreuzer (skreuzer@), R. Tyler Croy
- First working group was in BSDCan 2014
- Discussions continued in following conferences
- Second working group was in EuroBSDCon 2016
- Third working group was in AsiaBSDCon 2017 (yesterday!)
Jenkins

- An open source automation server written in Java
- Created by Kohsuke Kawaguchi
- https://jenkins.io
- Helps automating the non-human part of the whole software development process
  - Continuous Integration
  - Continuous delivery
- Extended by various of plugins
Created and operated by jenkins-admin@ since the group formed
Independent successor of tinderbox.FreeBSD.org
Periodically builds and tests FreeBSD src and doc trees
  - Branches: head, stable-11, stable-10, stable-9
  - Architectures: amd64, i386, sparc64
  - Others: build with GCC
    - Leverage the cross toolchain
    - amd64 cross build amd64 itself :)
Repository: https://github.com/freebsd/freebsd-ci
https://Jenkins.FreeBSD.org
2016, https://CI.FreeBSD.org

- Started in summer 2016
- Also based on Jenkins
- Experiment for some new ideas and features
- Artifact server introduced
  - Pass artifact between stages in a pipeline
  - Open to the public for doing more tests
- Job template
  - Make it easier to add new CI jobs without knowing jenkins configuration
- Codes are currently in my fork repository, will merge back in the future
  - https://github.com/lwhsu/jjb-freebsd-ci (Jenkins configuration)
  - https://github.com/lwhsu/freebsd-ci (everything else)
head-amd64-* Jobs on CI.FreeBSD.org
Automatic Build on https://CI.FreeBSD.org

- **build jobs**
  - scripts/build/build-world-kernel.sh
  - Generalized for different versions and architectures
  - make buildworld & buildkernel
  - Build distribution files

- **LINT jobs**
  - scripts/build/build-kernel-LINT.sh
  - Build LINT kernel
Self Tests
Kyua

- Testing framework for infrastructure software
- By Julio Merino <jmmv@{NetBSD,FreeBSD}.org>
- Repository
  - https://github.com/jmmv/kyua
- Documentation
  - https://github.com/jmmv/kyua/wiki/Quickstart
  - Julio’s talk & tutorial at AsiaBSDCon 2014
    - https://drive.google.com/drive/folders/0B08g-X1kPkSYNnlEdTB5RjIfbkk
bhyve - The BSD Hypervisor

- http://bhyve.org/
- Built in FreeBSD since 10.0
- On ci.FreeBSD.org
  - *-images jobs build virtual machine images installed by distribution files
  - *-testvm jobs build virtual machine with kyua installed
    - Basically the same with above job, but virtual disk size is smaller
- *-test jobs take testvm image from artifact server and run it
  - For RISC-V architecture we use QEMU
Putting Them Together

- **Create virtual machine image**
  - Pre-install kyua package and test files
  - Customize `/etc/rc.local` script
    - Run tests
    - Generate reports
    - Shutdown virtual machine
  - `scripts/build/build-test_image.sh`

- **Run tests in virtual machine**
  - Fetch the vm image, boot it
  - Wait for it completes running tests, or kill it when timeout
  - Extract test reports from the vm image
  - `scripts/test/run-tests.sh`
Access to Results
JUnit Format Output

<testcase classname="bin.cat.cat_test" name="align" time="0.018">
<system-out>Executing command [ /bin/sh -c cat -be /usr/tests/bin/cat/d_align.in ]</system-out>
<system-err>Test case metadata
--------------------
allowed_architectures is empty
allowed_platforms is empty
description = Test that cat(1) aligns the output right with options '-be' (PR bin/4841)
has_cleanup = false
is_exclusive = false
required_configs is empty
required_disk_space = 0
required_files is empty
required_memory = 0
required_programs is empty
required_user is empty
timeout = 300

Timing information
--------------------
Start time: 2016-09-23T01:22:37.490230Z
End time:   2016-09-23T01:22:37.508471Z
Duration:   0.018s

Original stderr
---------------
&lt;EMPTY&gt;
</system-err></testcase>
## Test Result
0 failures (-3), 64 skipped (-81)

## All Tests

<table>
<thead>
<tr>
<th>Package</th>
<th>Duration</th>
<th>Fail</th>
<th>(diff)</th>
<th>Skip</th>
<th>(diff)</th>
<th>Pass</th>
<th>(diff)</th>
<th>Total</th>
<th>(diff)</th>
</tr>
</thead>
<tbody>
<tr>
<td>bin.cat</td>
<td>61 ms</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>bin.date</td>
<td>0.89 sec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>39</td>
<td>0</td>
<td>39</td>
<td>0</td>
</tr>
<tr>
<td>bin.dd</td>
<td>0.16 sec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>bin.expr</td>
<td>0.25 sec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>14</td>
<td>0</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>bin.ls</td>
<td>6.8 sec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>35</td>
<td>0</td>
<td>35</td>
<td>0</td>
</tr>
<tr>
<td>bin.mv</td>
<td>0.36 sec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>bin.pax</td>
<td>95 ms</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>bin.pkill</td>
<td>55 sec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>28</td>
<td>0</td>
<td>28</td>
<td>0</td>
</tr>
<tr>
<td>bin.sh.builtins</td>
<td>3.9 sec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>163</td>
<td>0</td>
<td>163</td>
<td>0</td>
</tr>
</tbody>
</table>
Failed Test Case Display

Failed

sys.geom.class.gate.ggate_test.ggatel_file (from (test-report.xml))

Failing for the past 3 builds (Since #1505)
Took 60 ms.

Error Message

work md5 checksum didn't match

Standard Output

Executing command [ ggatel create -u 0 work ]

Standard Error

Test case metadata
--------------
Test Result Trend

Project FreeBSD-head-amd64-test

Recent Changes

Latest Test Result (1 failure / -3)

Upstream Projects

FreeBSD-head-amd64-testvm

Permalinks

- Last build (#1508), 34 min ago
- Last stable build (#1504), 8 hr 22 min ago
- Last successful build (#1507), 3 hr 14 min ago
- Last failed build (#1387), 7 days 7 hr ago
- Last unstable build (#1507), 3 hr 14 min ago
- Last unsuccessful build (#1507), 3 hr 14 min ago
- Last completed build (#1507), 3 hr 14 min ago
Artifacts

- ***.txz**
  - From *-build jobs
  - Distribution files just like what on https://download.FreeBSD.org

- **disk.img.xz**
  - From *-images jobs
  - Virtual machine disk image file, installed from above *.txz files

- **disk-test.img.xz**
  - From *-testvm jobs
  - Virtual machine disk image file, with test script and test case
artifact.ci.FreeBSD.org

- ftpes://artifact.ci.FreeBSD.org
  - Serves files upload from jobs
    - Through Jenkins "Publish Over FTP" Plugin
    - Add FTP over TLS support

- https://artifact.ci.FreeBSD.org
  - Serves files download for downstream jobs in pipeline or external users
  - snapshots/
    - <branch>/<svn_revision>/<target>/<target_arch>
      - head/r303226/amd64/amd64
Open Configurations
Jenkins Job Builder

- By OpenStack project
  - http://docs.openstack.org/infra/jenkins-job-builder/
- Before that, Jenkins job can only be edited on web interface
- Jenkins Job Builder provides YAML -> Jenkins configurations
- Create / update job through Jenkins’ web service
- Make jenkins configurations trackable and reproducible
- https://github.com/lwhsu/jjb-freebsd-ci
Make Creating New Job Easier

- Not everyone knows (and has to know) how to configure Jenkins
- Most of the build jobs just have different build scripts
- Create a "job template"
  - Environment setup
    - Check out latest source code
    - Setup needed version of FreeBSD
    - Install required packages
  - Execute specified build script for that job
  - Environment cleanup
- job:
  
  name: FreeBSD-head-scan_build
  
  scm:
  - FreeBSD-src-head

  builders:
  - checkout-scripts
  - setup-jail
  - execute-in-jail

  publishers:
  - clean-jail

  wrappers:
  - timestamps
Build-in-jail Script Sets

- Simplify clean environment setup and job execution
  - Three phase: setup.sh, execute.sh, clean.sh
  - sudo installed by default
- A typical job configuration
  - build.sh
    - Main build script, which is executed by execute.sh in the jail
  - jail.conf
    - Jail environment configuration: version, arch, etc.
  - pkg-list
    - Package needs to be installed (from pkg.FreeBSD.org)
Adding a New Job

- Submitters bring their own build scripts and confs:
  - jail.conf, pkg-list, build.sh
  - Other configurations (ex: src.conf, make.conf)
- Jenkins-admin merges scripts to FreeBSD-ci/jobs/<job name>/
- Jenkins-admin creates a new entry jenkins job builder, push it
- Done! Our restless Jenkins executes it around the clock
Example: FreeBSD-head-aarch64-build

- **jail.conf**
  ```
  TARGET=amd64
  TARGET_ARCH=amd64
  WITH_32BIT=0
  OSRELEASE=11.0-RELEASE
  ```

- **pkg-list**
  ```
  aarch64-binutils
  ```

- **build.sh**
  ```
  #!/bin/sh
  env \n
  JFLAG=${{BUILDER_JFLAG}} \n
  TARGET=arm64 \n
  TARGET_ARCH=aarch64 \n
  sh -x freebsd-ci/scripts/build/build-world-kernel.sh
  ```
Quarantine Mode

- Used to build with patch from external
- QUARANTINE=1 in jail.conf
- Using jail to create an isolated environment
  - Remove internet connection after required packages installed
- Limitation job resource from jenkins job setup
  - Limit execution time
    wrappers:
      - timeout:
        timeout: 90
        fail: true
        type: absolute
Other Integrations
Notifications

- **When to notify**
  - Build failed
  - Build unstable (compile successfully, but some test cases failed)
  - Build back to stable

- **How to notify**
  - Email
  - IRC

- **What to notify**
  - Changes since last build (who & what)
  - Tail of the console output
  - What are the test cases failed
Email Notification

- Double-edged sword
- If nobody reads, it's totally no use
- Email-ext plugin

freebsd-current mailing list, July 2016

- Jenkins build is still unstable: FreeBSD_ HEAD #500  jenkins-admin at FreeBSD.org
- Jenkins build is still unstable: FreeBSD_ HEAD #501  jenkins-admin at FreeBSD.org
- bhyve with lagg failsver doesn't work on wifi  Randy Westlund
- Jenkins build is still unstable: FreeBSD_ HEAD #502  jenkins-admin at FreeBSD.org
- FreeBSD_ HEAD #386 - Build #3721 - Failure  jenkins-admin at FreeBSD.org
- Build failed in Jenkins: FreeBSD_ HEAD #503  jenkins-admin at FreeBSD.org
- AR8327 Etherswitch led control patch  Dan Nelson
- FreeBSD_ HEAD #386 - Build #3722 - Still Failing  jenkins-admin at FreeBSD.org
- EARLY_AP_STARTUP hangs during boot  Gary Jennejohn
- FreeBSD_ HEAD #386 - Build #3723 - Fixed  jenkins-admin at FreeBSD.org
- Jenkins build is still unstable: FreeBSD_ HEAD #504  jenkins-admin at FreeBSD.org
- Jenkins build is still unstable: FreeBSD_ HEAD #505  jenkins-admin at FreeBSD.org
- Jenkins build is still unstable: FreeBSD_ HEAD #506  jenkins-admin at FreeBSD.org
- AR8327 Etherswitch led control patch  Dan Nelson
- EARLY_AP_STARTUP hangs during boot  John Baldwin
- Jenkins build is still unstable: FreeBSD_ HEAD #507  jenkins-admin at FreeBSD.org
- FreeBSD_ HEAD #508  jenkins-admin at FreeBSD.org
- resolvconf needs @RESTARTCMD@ to be replaced after r303062  Guy Yar
- Jenkins build is still unstable: FreeBSD_ HEAD #509  jenkins-admin at FreeBSD.org
- resolvconf needs @RESTARTCMD@ to be replaced after r303062  Pedro Giffuni
- resolvconf needs @RESTARTCMD@ to be replaced after r303062  Glen Barber
- resolvconf needs @RESTARTCMD@ to be replaced after r303062  Pedro Giffuni
- Jenkins build is still unstable: FreeBSD_ HEAD #509  jenkins-admin at FreeBSD.org
- ACPI errors when booting laptop  Ben Woods
- Jenkins build is still unstable: FreeBSD_ HEAD #510  jenkins-admin at FreeBSD.org
- EFI boot: can we make loader.efi work as BOOT(x64, a64, arm, ia32).efi?  Xin Li
- EFI boot: can we make loader.efi work as BOOT(x64, a64, arm, ia32).efi?  Nathan Whitehorn
- Jenkins build is back to stable : FreeBSD_ HEAD #511  jenkins-admin at FreeBSD.org
- EARLY_AP_STARTUP hangs during boot  Gary Jennejohn
- [PATCH] randomized delay in locking primitives. take 2  Mateusz Guzik
- [PATCH] randomized delay in locking primitives. take 2  Konstantin Belousov
- Jenkins build is back to normal : FreeBSD_ HEAD _sparc64 #185  jenkins-admin at FreeBSD.org
Reduce False Alert

- Be more careful before sending
- Pre-send script
  - Filter out the internal errors of the jenkins cluster
Notification Contents

- Use email-ext plugin to customize mail body
Template for Failed/Unstable Build Jobs

$PROJECT_NAME - Build #$BUILD_NUMBER - $BUILD_STATUS:

Build information: ${BUILD_URL}
Full change log: ${BUILD_URL}changes
Full build log: ${BUILD_URL}console

Change summaries:

${CHANGES, format="%r by %a:\n%m\n"}

The end of the build log:

${BUILD_LOG}
Example: Build Failed Mail

FreeBSD_HEAD_i386 - Build #3987 - Failure:

Build information: https://jenkins.FreeBSD.org/job/FreeBSD_HEAD_i386/3987/
Full change log: https://jenkins.FreeBSD.org/job/FreeBSD_HEAD_i386/3987/changes
Full build log: https://jenkins.FreeBSD.org/job/FreeBSD_HEAD_i386/3987/console

Change summaries:

306148 by jpaetzel:
Update oce driver to 11.0.50.0

Submitted by: Venkat Duvvuru <venkatkumar.duvvuru at broadcom.com>

The end of the build log:

[...truncated 190386 lines...]

--- oce_if.o ---
/usr/src/sys/modules/oce/.../dev/oce/oce_if.c:45:10: fatal error: 'oce_user.h' file not found
#include "oce_user.h"

1 error generated.
*** [oce_if.o] Error code 1

bmake[4]: stopped in /usr/src/sys/modules/oce

Template for Failed/Unstable Test Jobs

$PROJECT_NAME - Build #$BUILD_NUMBER - $BUILD_STATUS:

Build information: ${BUILD_URL}
Full change log: ${BUILD_URL}changes
Full build log: ${BUILD_URL}console

Change summaries:

${CHANGES, format="%r by %a:\n%n%m\n"}

The failed test cases:

${FAILED_TESTS,showStack=false,showMessage=true, maxTests=100}
Example: Test Failed Mail

FreeBSDHEAD-tests - Build #1746 - Still Unstable:

Build information: https://jenkins.FreeBSD.org/job/FreeBSDHEAD-tests/1746/
Full change log: https://jenkins.FreeBSD.org/job/FreeBSDHEAD-tests/1746/changes
Full build log: https://jenkins.FreeBSD.org/job/FreeBSDHEAD-tests/1746/console

Change summaries:
No changes

The failed test cases:
2 tests failed.
FAILED: sys.mqueue.mqueue_test.mqtest3

Error Message:
atf-check failed; see the output of the test for details

FAILED: sys.mqueue.mqueue_test.mqtest4

Error Message:
atf-check failed; see the output of the test for details

Integration with Code Review System

- Phabricator Differential Plugin
  - From Uber
    - https://github.com/uber/phabricator-jenkins-plugin
    - Good readme.md, just follow it
  - Supports only git (for now, patch for svn is preparing)
    - https://github.com/lwhsu/phabricator-jenkins-plugin/tree/scm
    - Basically provide option to select the main SCM
    - Conditionally issue SCM-specified options to arc
Patch Number Attached to each Build
Build #17 (Sep 3, 2016 2:24:56 PM)

Revision: 49365
No changes.

Started by remote host 2001:1900:2254:206a::16:75

This was a build of D7749 by unknown (unknown)

Igor fix

Reviewers: bcr, wblock

Reviewed By: bcr, wblock

Subscribers: wblock, bcr

Differential Revision: https://reviews.freebsd.org/D7749
Checkstyle: 66.114 warnings from one analysis.

- 1 fixed warning
Patch Build Status

On Phabricator:

- **Repository:** FreeBSD src repository
- **Lint:** Lint OK
- **Unit:** No Unit Test Coverage
- **Build Status:**
  - Buildable 5112
  - Build 5205: CI src build
  - Jenkins
  - Build 5204: arc lint + arc unit

- **Build Status:**
  - Buildable 5172
  - Build 5294: CI src build
  - Jenkins
Clang scan-build

- Static analysis tool provided by clang
- Clang Scan-Build Plugin
- bsd.clang-analyze.mk
  - Imported from NetBSD

```
make analyze \
CLANG_ANALYZE_OUTPUT_DIR=clangScanBuildReports \
CLANG_ANALYZE_OUTPUT=html
```
Clang scan-build Bug Report

<table>
<thead>
<tr>
<th>Group</th>
<th>Type</th>
<th>File</th>
<th>Path Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security</td>
<td>Potential insecure memory buffer bounds restriction in call 'strcpy'</td>
<td>/workspace/src/usr.sbin/ctm/ctm_rmail/ctm_rmail.c</td>
<td>1</td>
<td>Call to function 'strcpy' is insecure as it does not provide bounded memory. Replace unbounded copy functions with analogous functions that support length arguments such as 'strncpy'. CWE-119</td>
</tr>
<tr>
<td>Dead store</td>
<td>Dead assignment</td>
<td>/workspace/src/sys/modules/sym/.../.../dev/sym/sym_shipd.c</td>
<td>1</td>
<td>Value stored to 'fl' is never read</td>
</tr>
<tr>
<td>Dead store</td>
<td>Dead assignment</td>
<td>/workspace/src/sys/modules/sound/sound/.../.../dev/sound/pcm/mixer.c</td>
<td>1</td>
<td>Value stored to 'ret' is never read</td>
</tr>
<tr>
<td>Security</td>
<td>Potential insecure memory buffer bounds restriction in call 'strcpy'</td>
<td>/workspace/src/usr.bin/less/.../.../contrib/less/filename.c</td>
<td>1</td>
<td>Call to function 'strcpy' is insecure as it does not provide bounded memory. Replace unbounded copy functions with analogous functions that support length arguments such as 'strncpy'. CWE-119</td>
</tr>
<tr>
<td>Logic error</td>
<td>Dereference of null pointer</td>
<td>/workspace/src/lib/nurses/panelw/.../.../contrib/nurses/panel/p_bottom.c</td>
<td>4</td>
<td>Access to field '_begy' results in a dereference of a null pointer (loaded from field 'win')</td>
</tr>
<tr>
<td>Dead store</td>
<td>Dead assignment</td>
<td>/workspace/src/sys/modules/drm/radeon/.../.../dev/drm/r600_cp.c</td>
<td>1</td>
<td>Value stored to 'num_backends' is never read</td>
</tr>
</tbody>
</table>
```c
/* Recheck that ".." entry in the vdp directory points * to the inode we looked up before vdp lock was * dropped. */
error = ufs_lookup_ino(pdp, NULL, cnp, &ino);
if (error) {
  vput(tdp);
  return (error);
}
if (ino1 != ino) {
  vput(tdp);
  goto restart;
}
*vpp = tdp;
```

- Assuming 'error' is 0
- Taking false branch
- Taking false branch
- Dereference of null pointer (loaded from variable \texttt{vpp})
Igor + Checkstyle

- textproc/igor
  - By Warren Block <wblock@FreeBSD.org>
  - Automated proofreader for text files, man pages, and DocBook SGML source files

- Checkstyle
  - A tool to help programmers write Java code that adheres to a coding standard
  - Output an XML can be parsed by Jenkins Checkstyle Plugin

We added a new option (-X) to igor to output checkstyle XML
Checkstyle Result

CheckStyle Result

Warnings Trend

<table>
<thead>
<tr>
<th></th>
<th>All Warnings</th>
<th>New Warnings</th>
<th>Fixed Warnings</th>
</tr>
</thead>
<tbody>
<tr>
<td>66116</td>
<td></td>
<td>45</td>
<td>45</td>
</tr>
</tbody>
</table>

Summary

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>High Priority</th>
<th>Normal Priority</th>
<th>Low Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>66116</td>
<td></td>
<td>0</td>
<td>66116</td>
<td>0</td>
</tr>
</tbody>
</table>

Details

<table>
<thead>
<tr>
<th>Source Folder</th>
<th>Total</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>doc/en_US_ISO8859-1/articles/bsd-char</td>
<td>113</td>
<td></td>
</tr>
<tr>
<td>doc/en_US_ISO8859-1/articles/building-products</td>
<td>641</td>
<td></td>
</tr>
<tr>
<td>doc/en_US_ISO8859-1/articles/contributing</td>
<td>144</td>
<td></td>
</tr>
<tr>
<td>doc/en_US_ISO8859-1/articles/contributors</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>doc/en_US_ISO8859-1/articles/cups</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>doc/en_US_ISO8859-1/articles/explaining-bsd</td>
<td>225</td>
<td></td>
</tr>
<tr>
<td>doc/en_US_ISO8859-1/articles/fonts</td>
<td>91</td>
<td></td>
</tr>
</tbody>
</table>
Checkstyle Result Trend

- Compare with previous build

- Overall Trend

Checkstyle: 66,117 warnings from one analysis.

- 7 new warnings
- 16 fixed warnings
## Compiler Warnings

### LLVM/Clang Warnings

#### Warnings Trend

<table>
<thead>
<tr>
<th>All Warnings</th>
<th>New Warnings</th>
<th>Fixed Warnings</th>
</tr>
</thead>
<tbody>
<tr>
<td>853</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

#### Summary

<table>
<thead>
<tr>
<th>Total</th>
<th>High Priority</th>
<th>Normal Priority</th>
<th>Low Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>853</td>
<td>2</td>
<td>851</td>
<td>0</td>
</tr>
</tbody>
</table>

#### Details

<table>
<thead>
<tr>
<th>Source Folder</th>
<th>Total</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>:</code></td>
<td>116</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>1</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>1</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>1</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>1</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>1</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>30</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>25</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>17</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>14</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
<tr>
<td><code>usr/obj/…</code></td>
<td>17</td>
<td><img src="example" alt="Distribution" />/</td>
</tr>
</tbody>
</table>
Compiler Warnings Trend

LLVM/Clang Warnings Trend
Build Status Badge

- https://wiki.freebsd.org/arm64:

- https://<job url>/badge/icon
- https://<job url>/lastCompletedBuild/badge/icon
Other Jenkins Plugins Used

- **SCM Sync configuration**
  - Use post-commit hook to notify administrators

- **SafeRestart**
  - Schedule restart jenkins when all current jobs complete
  - Save and resume the build queue

- **Green balls**
  - People like green
  - Everything is fine
Epilogue
Summary

- Continuous Integration does help development of the FreeBSD project
- Hope these experience and codes can also help downstream projects
Future Work

● Have you to work on testing FreeBSD!
● Check for reproducible build
● More tests, even some are not enabled by default
  ○ Dtrace
● Build for project branches
  ○ Make testing feature branches easier
● CI for ports
  ○ Collaboration with "redports"
  ○ Automatically "exp-run"
● Performance tests
● Work with other projects
Acknowledgement

- jenkins-admin@FreeBSD.org
- The FreeBSD Foundation
- clusteradmin@FreeBSD.org
- phabric-admin@FreeBSD.org
- People on -testing, -current, -stable mailing lists