Pushing Ports through FreeBSD

Michael Shirk
CharmBUG
9/27/2018

Introduction

- Why am I doing this?
- What you need to build ports
- Walk-through of the process.
- Next Steps

Rant Warning

- Whenever you see the beastie with a hammer, there is a potential for some of my own personal bias to slip in.

- The goal is to minimize this throughout the talk.

- All information not cited in this talk is based on personal experience or opinion (marked with an asterisk *).

Why Am I Doing This?

- Motivation
  - I want something to be easier to install
    - And package up:

      ```
      pkg install -y awesome
      ```
  - There are TONS of applications that are available, but I need this app.
    - I will be talking about a python based application, and when you look at the amount of python ports, it does not seem like a big deal to add another.

What you need to build ports

- FreeBSD System
    - I used a jail with a copy of the ports tree.
    - I also used HardenedBSD, but the ports are the same with extra hardening options.
- Porters Handbook
    - https://www.freebsd.org/doc/en_US.ISO8859-1/books/porters-handbook/
        - Section 3 for quick porting is what I followed
- portlint package

    ```
    pkg install -y portlint
    ```

Writing the MakeFile

- bro-pkg is a python-based application that extends the capabilities of Bro NSM for functionality and security.
  - I tested installing by hand and only found one specific dependency for GitPython >= 2.1.8
- Use a MakeFile from another python port *
  - Made an easy starting point, and selected a port that depended on GitPython

```
www/py-frappe-bench/Makefile:    $
{PYTHON_PKGNAMEPREFIX}GitPython>=1.0.1:devel/py-
gitpython@${PY_FLAVOR} \
```

Writing the MakeFile

- Wait...bro-pkg requires something called btest
  - Written by folks in the Bro community, created as a driver for unit tests.
  - Another python tool
  - I need this port to be added before I can create a bro-pkg port

Writing the MakeFile

```
# $FreeBSD$

PORTNAME=       btest
DISTVERSION=    0.58
PORTREVISION=   1
CATEGORIES=     devel
MASTER_SITES=   CHEESESHOP
PKGNAMEPREFIX=  ${PYTHON_PKGNAMEPREFIX}

MAINTAINER=     mshirk@daemon-security.com
COMMENT=        A simple driver for basic unit tests

LICENSE=        BSD3CLAUSE

NO_ARCH=        yes
USES=           python

.include <bsd.port.mk>
```

Creating the pkg-desc

- Create a description for your port (80 character limit per line)

```
~/btest # cat pkg-descr
The btest is a simple framework for writing unit tests. Freely
borrowing some ideas from other packages, it's main objective is to
provide an easy-to-use, straightforward driver for a suite of
shell-based tests. Each test consists of a set of command lines that
will be executed, and success is determined based on their exit
codes. btest comes with some additional tools that can be used
within such tests to compare output against a previously established
baseline.

WWW: https://github.com/bro/btest/
```

Creating the distinfo file

- Run the following to generate the distinfo based on the installation files (with the MakeFile in the directory)

```
~/btest # make makesum
~/btest # cat distinfo
TIMESTAMP = 1537920428
SHA256 (btest-0.58.tar.gz) =
632f88977e1091cec2edf853826c8b0233f6839a6bf490217e3947d
8ba342402
SIZE (btest-0.58.tar.gz) = 68811
```

Creating the pkg-plist

- Normally this is required with a list of files, but in this case of using Python, the autoportlist option is used in the MakeFile to handle this dynamically by adding this line

```
USE_PYTHON=      autoplist concurrent distutils
```

Final MakeFile

```
# $FreeBSD$

PORTNAME= btest
DISTVERSION= 0.58
PORTREVISION= 1
CATEGORIES= devel
MASTER_SITES= CHEESESHOP
PKGNAMEPREFIX= ${PYTHON_PKGNAMEPREFIX}

MAINTAINER= mshirk@daemon-security.com
COMMENT= Simple driver for basic unit tests

LICENSE= BSD3CLAUSE

USES= python
USE_PYTHON= autoplist concurrent distutils

.include <bsd.port.mk>
```

Check the port config with portlint

- portlint is a tool written in Perl that will check that the files are properly formatted to work in the ports tree.

```
WARN: Makefile: PIE is listed in OPTIONS_DEFINE, but no PORT_OPTIONS:MPIE appears.
WARN: Makefile: RELRO is listed in OPTIONS_DEFINE, but no PORT_OPTIONS:MRELRO appears.
WARN: Makefile: SAFESTACK is listed in OPTIONS_DEFINE, but no PORT_OPTIONS:MSAFESTACK appears.
WARN: Makefile: CFI is listed in OPTIONS_DEFINE, but no PORT_OPTIONS:MCFI appears.
WARN: Makefile: RETPOLINE is listed in OPTIONS_DEFINE, but no PORT_OPTIONS:MRETPOLINE appears.
WARN: Makefile: COMMENT is not supposed to begin with 'A ', 'An ', or 'The '.
WARN: Makefile: extra item placed in the USES/USE_x section, for example, "NO_ARCH".
WARN: Makefile: seems to have unnecessary blank lines at the last part.
0 fatal errors and 8 warnings found.
```

- 6 of these warnings are HardenedBSD specific, the other 2 should be fixed.

Run through testing the port

- Process is based on the porters handbook to ensure the port builds and can be turned into a package (See section 3.4)

- Run the following from the directory with your port files

```
make -DBATCH stage
make check-orphans
make package
make install
make deinstall
make package (as user if necessary)
```

Did it work? Great

- To assist with including the port in the ports tree, create a diff in the patch format

- Because I was working from a github tree, I used git:

```
git diff HEAD output

patch –p1 –V none < ~/py-btest.patch
```

- When using git, I changed the "/dev/null" lines to match the filenames.

Example patchdiff

```
diff --git a/devel/py-btest/Makefile
b/devel/py-btest/Makefile
new file mode 100644
index 000000000000..880a4311a50c
--- a/devel/py-btest/Makefile
+++ b/devel/py-btest/Makefile
@@ -0,0 +1,20 @@
+# $FreeBSD$
+
+PORTNAME= btest
+DISTVERSION=    0.58
+PORTREVISION=   1
+CATEGORIES=  devel
+MASTER_SITES=   CHEESESHOP
+PKGNAMEPREFIX=   ${PYTHON_PKGNAMEPREFIX}
```

Moving onto bro-pkg

- Now I have the btest port, and I can just do the same thing for bro-pkg

- I glossed over how the python code is downloaded using the CHEESESHOP macro

- Makes it easier to make python ports

```
.if !defined(IGNORE_MASTER_SITE_CHEESESHOP)
MASTER_SITE_CHEESESHOP+= \

https://files.pythonhosted.org/packages/%SUBDIR%/ \
        https://pypi.org/packages/%SUBDIR%/
  .endif
```

But I can't do the same thing

- What is this mess?

```
https://files.pythonhosted.org/packages/7e/10/84c49dcf0170
04acfa7a453627155b7201e3102bb0c0a563b0a26f9e0ab2/bro_pkg-
1.4.2-py2.py3-none-any.whl
```

But I can't do the same thing

- The way in which this file is stored will not allow it to work correctly with the easy macros

- There is a tarfile on github from the authors, and the ports building has nice macros that work with github.

bro-pkg MakeFile

```
# $FreeBSD$

PORTNAME=       bro-pkg
DISTVERSION=    1.4.2
PORTREVISION=   1
CATEGORIES=     security
PKGNAMEPREFIX=  ${PYTHON_PKGNAMEPREFIX}

MAINTAINER=     mshirk@daemon-security.com
COMMENT=        Bro NSM package manager

LICENSE=        BSD3CLAUSE

RUN_DEPENDS=    ${PYTHON_PKGNAMEPREFIX}GitPython>=2.1.8:devel/py-gitpython@$
{PY_FLAVOR}

USES=           python
USE_PYTHON=     semantic_version configparser btest autoplist concurrent distutils

USE_GITHUB=     yes
GH_TUPLE=       bro:package-manager:cc21207

NO_ARCH=        yes

.include <bsd.port.mk>
```

USE_PYTHON

- That is not how that works, only for specific macros that assist with building the python port.

- You need to add in specific RUN_DEPENDS lines for the python modules you need to make it work.

- This includes the btest port I need to run bro-pkg.

bro-pkg MakeFile updated

```
# $FreeBSD$

PORTNAME=       bro-pkg
DISTVERSION=    1.4.2
PORTREVISION=   1
CATEGORIES=     security
PKGNAMEPREFIX=  ${PYTHON_PKGNAMEPREFIX}

MAINTAINER=     mshirk@daemon-security.com
COMMENT=        Bro NSM package manager

LICENSE=        BSD3CLAUSE

RUN_DEPENDS=    ${PYTHON_PKGNAMEPREFIX}GitPython>=2.1.8:devel/py-gitpython@$
{PY_FLAVOR} \
                ${PYTHON_PKGNAMEPREFIX}semantic_version>=0:devel/py-
semantic_version@${PY_FLAVOR} \
                ${PYTHON_PKGNAMEPREFIX}configparser>=0:devel/py-configparser@$
{PY_FLAVOR} \
                ${PYTHON_PKGNAMEPREFIX}btest>=0:devel/py-btest@${PY_FLAVOR}

USES=           python
USE_PYTHON=     autoplist concurrent distutils

USE_GITHUB=     yes
GH_TUPLE=       bro:package-manager:cc21207

NO_ARCH=        yes

.include <bsd.port.mk>
```

My ports install correctly

- After running through the same checks as btest, I verified bro-pkg installed correctly onto my system

- The port will install all of the necessary dependencies and should work.

- Now I am ready to submit a bug report to get both ports added

Proper Bug Report

- I watch numerous emails go by on how to submit bugs to FreeBSD

- I have submitted normal bug reports for issues in the base system.

- I took the latest Python NEW PORT bug report that was closed as completed and used that as a guide*
  - (Assuming there is a nice how-to somewhere)

- Key item is that I first put in a bug for btest, followed by bro-pkg, and made bro-pkg dependent on the btest bug being closed.

Response

- Received the following feedback on my btest bug

```
+PORTREVISION=   1
You're not revising a port, without bumping
version. New ports don't need this; Only time
PORTREVISION is needed is when updating a port
without updating port version
```

Revised btest MakeFile

```
# $FreeBSD$

PORTNAME= btest
DISTVERSION= 0.58
CATEGORIES= devel
MASTER_SITES= CHEESESHOP
PKGNAMEPREFIX= ${PYTHON_PKGNAMEPREFIX}

MAINTAINER= mshirk@daemon-security.com
COMMENT= Simple driver for basic unit tests

LICENSE= BSD3CLAUSE

USES= python
USE_PYTHON= autoplist concurrent distutils

.include <bsd.port.mk>
```

Response

- Received the following feedback on my bro-pkg bug

```
+PORTREVISION= 1
Remove
```

```
BSD3CLAUSE and first line of RUN_DEPENDS is not aligned
the same, nor is python in USES=
```

```
GH_TUPLE
Remove :cc21207 and add DISTVERSIONPREFIX=    v
under PORTNAME
https://www.freebsd.org/doc/en_US.ISO8859-1/books/
porters-handbook/makefile-distfiles.html#makefile-
master_sites-github-ex3
```

bro-pkg MakeFile updated

```
# $FreeBSD$

PORTNAME=       bro-pkg
DISTVERSIONPREFIX=      v
DISTVERSION=    1.4.2
CATEGORIES=     security
PKGNAMEPREFIX= ${PYTHON_PKGNAMEPREFIX}

MAINTAINER=     mshirk@daemon-security.com
COMMENT=        Bro NSM package manager

LICENSE=        BSD3CLAUSE

RUN_DEPENDS=    ${PYTHON_PKGNAMEPREFIX}GitPython>=2.1.8:devel/py-gitpython@$
{PY_FLAVOR} \
                ${PYTHON_PKGNAMEPREFIX}semantic_version>=0:devel/py-
semantic_version@${PY_FLAVOR} \
                ${PYTHON_PKGNAMEPREFIX}configparser>=0:devel/py-configparser@$
{PY_FLAVOR} \
                ${PYTHON_PKGNAMEPREFIX}btest>=0:devel/py-btest@${PY_FLAVOR}

USES=           python
USE_PYTHON=     autoplist concurrent distutils

USE_GITHUB=     yes
GH_ACCOUNT=     bro
GH_PROJECT=     package-manager

NO_ARCH=        yes

.include <bsd.port.mk>
```

Lessons Learned

- Using another port as a guide did not get me into too much trouble

- Using the porters handbook and portlint ensures things are in generally the correct format before submitting

- The bugs are still open at this time, with the goal of them eventually making their way into the ports tree for FreeBSD
  - And then down to HardenedBSD, DragonFlyBSD

Next Steps

- Next FreeBSD ports I am going to work on:
  - lb or load balancing application that is a part of the netmap framework
    - I have asked for this to be included in base, but for now will live with a port
  - I am also going to be working on a METAPORT to install all of the necessary NSM tools on FreeBSD
  - suricata-update – another python based script that should be easy to port
- OpenBSD Ports I need to create
  - Same two ports (btest, bro-pkg)
  - Pulledpork (perl script for Snort rule management)
  - suricata-update – if it does not exist already.

Another port example

- What if I want to make a port package that just copies specific files into place, without anything else?

  https://svnweb.freebsd.org/ports/head/x11-themes/lumina-themes/distinfo?view=markup&pathrev=453596

- Used by Lumina desktop to just copy desktop wallpapers down to a users system.

- Thanks to JT for sharing this example.

Questions?