

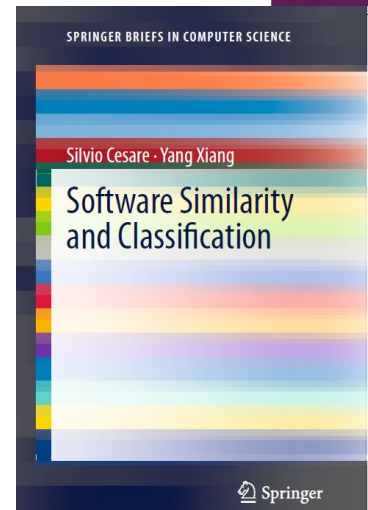
CLONewise

Automatically Detecting Package Clones and Inferring Security Vulnerabilities

Silvio Cesare
Deakin University
<silvio.cesare@gmail.com>

WHO AM I AND WHERE DID THIS TALK COME FROM?

- PhD Candidate at Deakin University, AU.
- Research interests:
 - Malware detection
 - Automated vulnerability detection
- Book author
 - Software similarity and classification, Springer.
 - <http://www.springer.com/computer/security+and+cryptography/book/978-1-4471-2908-0>
- <http://www.FooCodeChu.com>



INTRODUCTION

- ⦿ Developers may “embed” or “clone” code from 3rd party sources
 - Static linking
 - Maintaining a internal copy of a library.
 - Forking a library.
- ⦿ Lots of examples
 - XML parsing → libxml in various programs
 - Image processing → libpng in Firefox
 - Networking → Open SSL in Cisco IOS
 - Compression → zlib everywhere

EMBEDDED IS BAD PRACTICE

- ◉ Linux policies generally disallow (image below).
- ◉ It still happens.
- ◉ Multiple versions of packages now exist.
- ◉ Each copy needs patches from upstream.
- ◉ Copies become insecure over time from unapplied patches.



THE MANUAL APPROACH

- ◉ Scan binaries for version strings.
- ◉ Done in 2005 on mass scale for zlib in Debian Linux.

```
bzlib_private.h:#define BZ_VERSION  "1.0.5, 10-Dec-2007"
```

```
tiffvers.h:#define TIFFLIB_VERSION_STR "LIBTIFF, Version  
3.8.2\nCopyright (c) 1988-1996 Sam Leffler\nCopyright (c)  
1991-1996 Silicon Graphics, Inc."
```

```
png.h:#define PNG_HEADER_VERSION_STRING \  
" libpng version 1.2.27 - April 29, 2008\n"
```

MOTIVATION

- ◉ 10,000 - 20,000 packages in Linux distros.
- ◉ Debian tracks over 420 libraries (see below).
- ◉ Most distros don't track at all.
- ◉ How many vulnerabilities are there?
- ◉ How to automate?

```
67  
68 php-htmlpurifier  
69     - mahara 1.2.5-1 (embed)  
70     - knowledgeroot 0.9.9.5-5 (embed)  
71     - moodle <unfixed> (embed)  
72  
73 peercast  
74     - gnome-peercast <removed> (embed)  
75     [etch] - gnome-peercast <unfixed> (embed)  
76
```

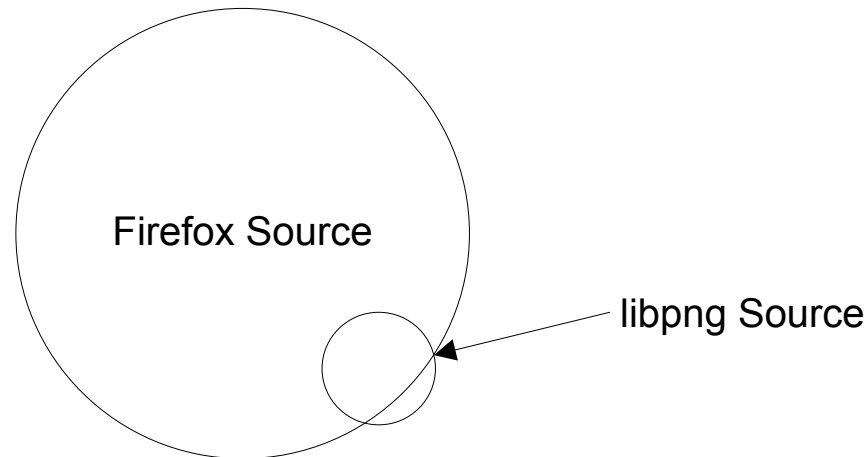
OUTLINE

1. Problem definition and our approach
2. Statistical classification
3. Scaling the analysis
4. Inferring security vulnerabilities
5. Implementation and evaluation
6. Discussion
7. Related work
8. Future work and conclusion

Remember to complete the Black Hat speaker feedback survey.

PROBLEM DEFINITION

- ◉ Find package code re-use in sources.
- ◉ Infer vulns caused by out-of-date code .



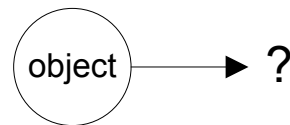
OUR APPROACH

- Consider code re-use detection a binary classification problem:
 - Do packages A and B share code? Yes or no?
- Features for classification:
 - Common filenames
 - Hashes
 - Fuzzy content

STATISTICAL CLASSIFICATION

STATISTICAL CLASSIFICATION

- ◉ Classification assigns classes to objects.
- ◉ Supervised learning.
- ◉ Unsupervised learning.



Class 1 - Spam

Class 2 - Not Spam

FEATURE EXTRACTION

◉ Feature vector

1. N_Filenames_A
2. N_Filenames_Source_A
3. N_Filenames_B
4. N_Filenames_Source_B
5. N_Common_Filenames
6. N_Common_Similar_Filenames
7. N_Common_FilenameHashes
8. N_Common_FilenameHash80
9. N_Common_ExactFilenameHash
10. N_Score_of_Common_Filename
11. N_Score_of_Common_Similar_Filename
12. N_Score_of_Common_FilenameHash
13. N_Score_of_Common_FilenameHash80
14. N_Score_of_Common_ExactFilenameHash80
15. N_Data_Common_Filenames
16. N_Data_Common_Similar_Filenames
17. N_Data_Common_FilenameHashes
18. N_Data_Common_FilenameHash80
19. N_Data_Common_ExactFilenameHash
20. N_Data_Score_of_Common_Filename
21. N_Data_Score_of_Common_Similar_Filename
22. N_Data_Score_of_Common_FilenameHash
23. N_Data_Score_of_Common_FilenameHash80
24. N_Data_Score_of_Common_ExactFilenameHash80
25. N_Common_ExactHash
26. N_Common_DataExactHash

NUMBER OF COMMON FILENAMES

- Source and data.
- Normalize names.

c
cpp
cxx
cc
php
inc
java
py
rb
js
pl
pm
ml
mli
lua

expat-2.0.1/lib	tla-1.3.5+dfsg/src/expat/lib/
amigaconfig.h	
ascii.h	ascii.h
asciitab.h	asciitab.h
expat.dsp	expat.dsp
expat_external.h	expat_external.h
expat.h	expat.h
expat_static.dsp	expat_static.dsp
expatw.dsp	expatw.dsp
expatw_static.dsp	expatw_static.dsp
iasciitab.h	iasciitab.h
internal.h	internal.h
latin1tab.h	latin1tab.h
libexpat.def	libexpat.def
libexpatw.def	libexpatw.def
macconfig.h	macconfig.h
Makefile.MPW	Makefile.MPW
nametab.h	nametab.h
utf8tab.h	utf8tab.h
winconfig.h	winconfig.h
xmlparse.c	xmlparse.c
xmlrole.c	xmlrole.c
xmlrole.h	xmlrole.h
xmltok.c	xmltok.c
xmltok.h	xmltok.h
xmltok_impl.c	xmltok_impl.c
xmltok_impl.h	xmltok_impl.h
xmltok_ns.c	xmltok_ns.c

NUMBER OF SIMILAR FILENAMES

- ◉ Edit distance between filenames.
- ◉ Similarity $\geq 85\%$

$$\textit{similarity}(s, t) = 1 - \frac{\textit{edit_dist}(s, t)}{\max(\textit{len}(s), \textit{len}(t))}$$

NUMBER OF FILES WITH IDENTICAL OR SIMILAR CONTENT

- ◉ Use fuzzy hashing (ssdeep).
- ◉ Number of identical hashes.
- ◉ Number of > 80% similar hashes.
- ◉ Number of > 0% similar hashes.

```
ssdeep,1.0--blocksize:hash:hash,filename  
96:KQhaGCVZGhr83h3bc0ok3892m12wzgnH5w2pw+sxNEI58:FIVkH4x73h39LH+2w+sxaD,"config.h"  
96:MD9fHjsEuddrg31904l8bgx5ROg2MQZHZqpAlycowOsexbHDbk:MJwz/l2PqGqqbr2yk6pVgrwPV,"INSTALL"  
96:EQOJvOl4ab3hhiNFXc4wwcweomr0cNJDBoqXjmAHKX8dEt001nfEhVluX0dDcs:3mzpAsZpprbshfu3oujjdEN  
dp21,"README"
```

SCORING FILENAMES

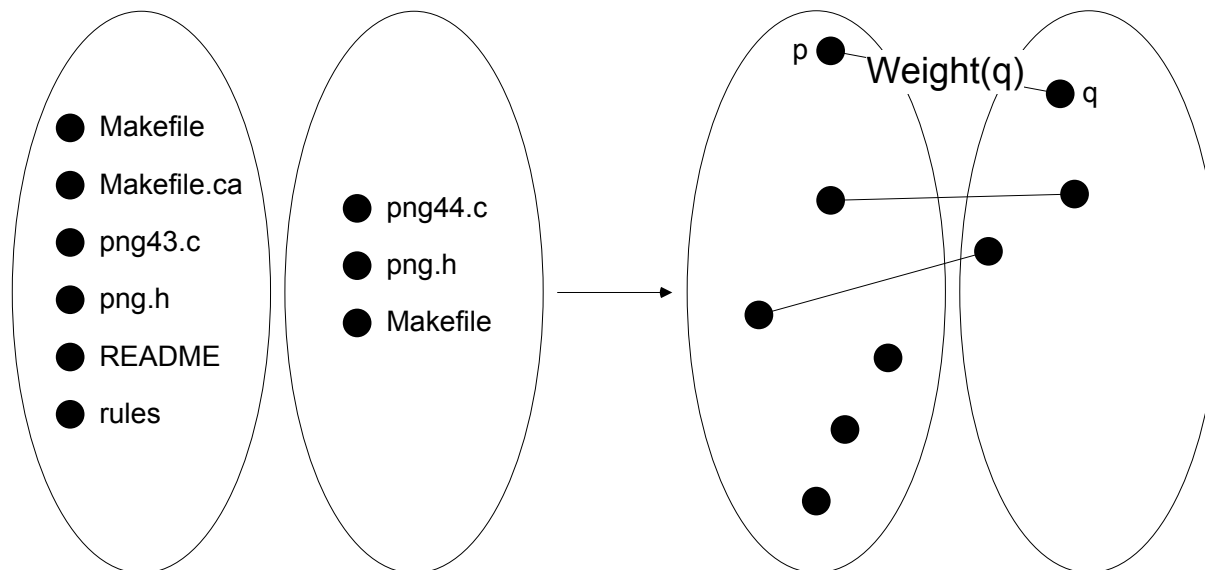
- ◉ README filenames less important.
- ◉ libpng.c more important .
- ◉ Score filenames using ‘inverse document frequency.’

$$idf(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

- ◉ Sum scores of matching filenames.

MATCHING FILENAMES BETWEEN PACKAGES

- ◉ Which similar filenames to match?
- ◉ Each matching has a cost - the filename score.
- ◉ Choose matchings to maximize sum of costs.



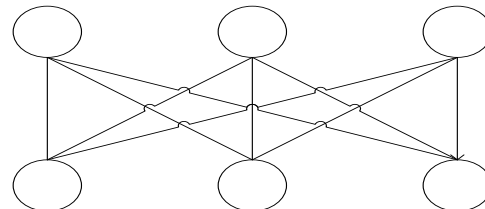
THE ASSIGNMENT PROBLEM

- Given two sets, A and T , of equal size, together with a weight function $C: A \times T \rightarrow \mathbb{R}$. Find a bijection $f: A \rightarrow T$ such that the cost function:

$$\sum_{a \in A} C(a, f(a))$$

is optimal.

- Known in combinatorial optimisation as ‘the assignment problem.’
- Solved optimally in cubic time.
- Greedy solution is faster.



FEATURE SELECTION

- ◉ Not all features are important.

- ◉ Feature ranking.

1. Feature1
2. Feature2
3. Feature3



1. Feature3 (0.80)
2. Feature1 (0.60)
3. *Feature2 (0.01)*

- ◉ Subset selection.

1. Feature1
2. Feature2
3. Feature3

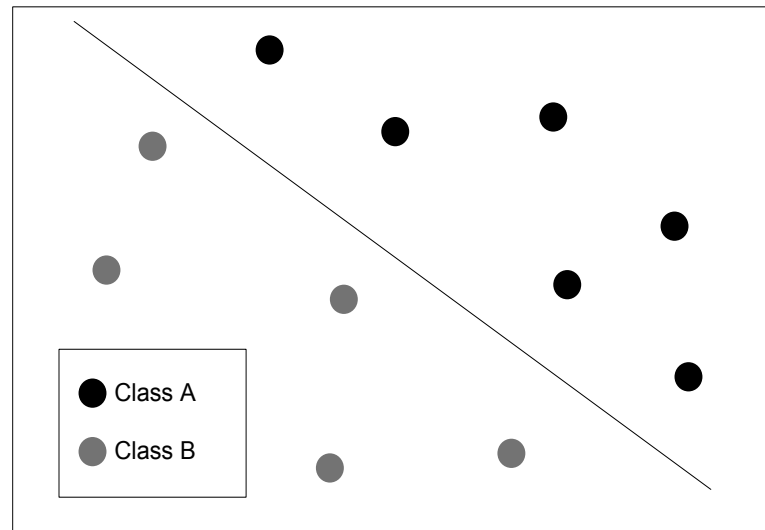


1. Feature1
2. Feature2

- ◉ We chose not to use it.

CLASSIFICATION

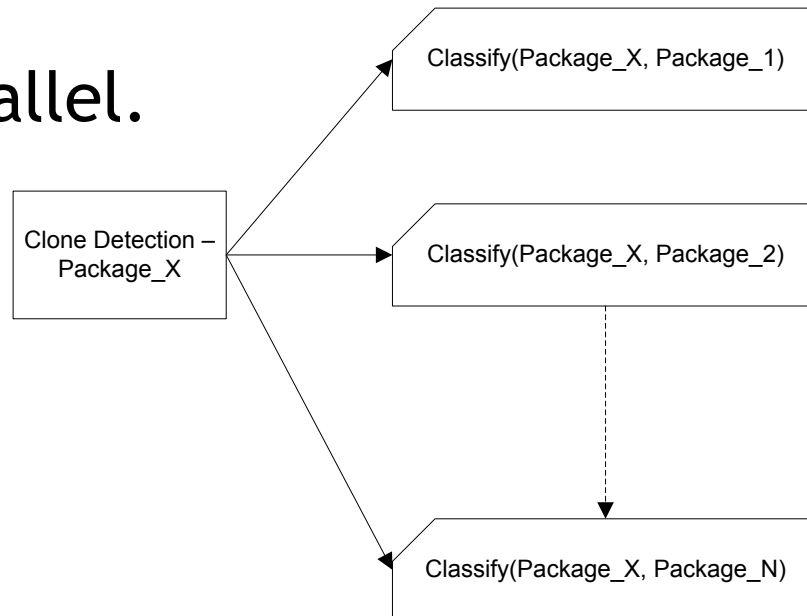
- Consider feature vectors as N-dimensional points.
- Linear classifiers.
- Non linear classifiers.
- Decision trees.



SCALING THE ANALYSIS

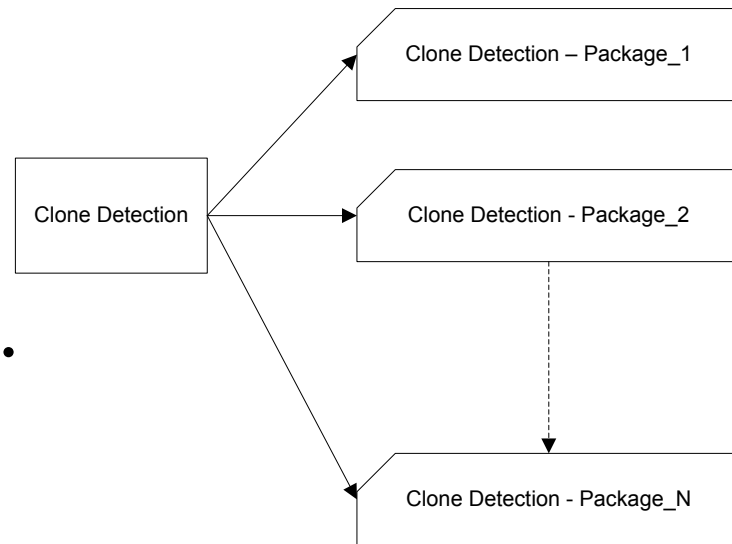
MULTICORE

- ◉ Speedup clone detection on a package.
- ◉ Open MP.
- ◉ Embarrassingly parallel.



CLUSTERING

- ◉ Open MPI.
- ◉ Single job is clone detection on package.
- ◉ Slaves consume jobs.
- ◉ Embarrassingly parallel.



RUNNING THE ANALYSIS

- 4 Node Amazon EC2 Cluster
 - Dual CPU
 - 8 cores per CPU
 - 88 EC2 compute units
 - 60.5G memory per node
- Clone detection on embedded libs known by Debian.
- Store the results for later use.

INFERRING SECURITY VULNERABILITIES

PACKAGE CLONE DETECTION USE-CASE

○ By package

National Cyber-Alert System
Vulnerability Summary for CVE-2010-0205
Original release date: 03/03/2010
Last revised: 11/18/2010
Source: US-CERT/NIST

Overview

The png_decompress_chunk function in pngutil.c in libpng 1.0.x before 1.0.53, 1.2.x before 1.2.43, and 1.4.x before 1.4.1 does not properly handle compressed ancillary-chunk data that has a disproportionately large uncompressed representation, which allows remote attackers to cause a denial of service (memory and CPU consumption, and application hang) via a crafted PNG file, as demonstrated by use of the deflate compression method on data composed of many occurrences of the same character, related to a "decompression bomb" attack.

```
$ Clonewise query-cache libpng
# The following package clones are tracked in the embedded-code-copies
# database. They have not been fixed.
#
libpng CLONED_IN_SOURCE ia32-libs <unfixed>
```

STANDARDIZATION EFFORTS

National Cyber-Alert System

Vulnerability Summary for CVE-2010-0205

Original release date: 03/03/2010

Last revised: 11/18/2010

Source: US-CERT/NIST

Overview

The `png_decompress_chunk` function in `pngutil.c` in `libpng 1.0.x` before 1.0.53, 1.2.x before 1.2.43, and 1.4.x before 1.4.1 does not properly handle compressed ancillary-chunk data that has a disproportionately large uncompressed representation, which allows remote attackers to cause a denial of service (memory and CPU consumption, and application hang) via a crafted PNG file, as demonstrated by use of the deflate compression method on data composed of many occurrences of the same character, related to a "decompression bomb" attack.

Summary: Off-by-one error in the `__opiereadrec` function in `readrec.c` in `libopie` in `OPIE 2.4.1-test1` and earlier, as used on FreeBSD 6.4 through 8.1-PRERELEASE and other platforms, allows remote attackers to cause a denial of service (daemon crash) or possibly execute arbitrary code via a long username, as demonstrated by a long `USER` command to the FreeBSD 8.0 `ftpd`.

Official Common Platform Enumeration (CPE) Dictionary

CPE is a structured naming scheme for information technology systems, software, and packages. Based upon the generic syntax for Uniform Resource Identifiers (URI), CPE includes a formal name format, a method for checking names against a system, and a description format for binding text and tests to a name.

Below is the current official version of the CPE Product Dictionary. The dictionary provides an agreed upon list of official CPE names. The dictionary is provided in XML format and is available to the general public. Please check back frequently as the CPE Product Dictionary will continue to grow to include all past, present and future product releases. The CPE Dictionary is updated nightly when modifications or new names are added. Archived CPE dictionaries are available at <http://static.nvd.nist.gov/feeds/xml/cpe/dictionary/>.


As of December 2009, The National Vulnerability Database is now accepting contributions to the Official CPE Dictionary. Organizations interested in submitting CPE Names should contact the NVD CPE team at cpe_dictionary@nist.gov for help with the processing of their submission.

The CPE Dictionary hosted and maintained at NIST may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright in the United States. Attribution would, however, be appreciated by NIST.

DEBIAN SECURITY TRACKING

[/\[secure-testing\]](#)

Index of /



Files shown: **1**
Directory revision: [19695](#) (of [19695](#))
Sticky Revision:

File ^	Rev.	Age	Author	Last log entry
bin/	19639	6 days	geissert	The Bugnum field shouldn't contain the hash character
check-external/	18956	2 months	geissert	Add script that is meant to be executed by a cronjob
data/	19695	2 hours	joeyh	automatic update
doc/	19273	7 weeks	pabs	Fix or drop links that changed or broke with the aliioth transition. Thanks to P...
hardening/	19669	3 days	jmm	chrony hardened
lib/	18465	4 months	pabs	Fix a crash when running the update script with a newer apt_pkg
org/	19600	13 days	luciano	luciano frontdesk
stamps/	1934	6 years	fw	Add list parser written in Python. "make check" runs a syntax check (no SQLite ...
website/	19273	7 weeks	pabs	Fix or drop links that changed or broke with the aliioth transition. Thanks to P...
Makefile	18790	3 months	fw	Makefile: remove one more missing -old- dependency

[/\[secure-testing\]](#) / [data](#) / [CVE](#) / [list](#)

Contents of /data/CVE/list

[Parent Directory](#) | [Revision Log](#)

Revision [19695](#) - ([show annotations](#)) ([download](#))
Sun Jul 8 21:14:29 2012 UTC (2 hours, 34 minutes ago) by joeyh
File size: 7142120 byte(s)

automatic update

1	CVE-2012-3863 [asterisk: Possible resource leak on uncompleted
2	- asterisk <unfixed>
3	CVE-2012-3847 (slssvc.exe in Invensys Wonderware SuiteLink in
4	NOT-FOR-US: Windows utility
5	CVE-2012-3846 (Cross-site scripting (XSS) vulnerability in inc
6	NOT-FOR-US: php-pastebin not in Debian

Contents of /data/CPE/list

[Parent Directory](#) | [Revision Log](#)

Revision [18936](#) - ([show annotations](#)) ([download](#))

Fri Apr 13 12:05:16 2012 UTC (2 months, 3 weeks ago) by pere

File size: 55047 byte(s)

Update list of CPE entries and aliases based on CVE ids for 2011 and 2012.

```
1 a2ps;cpe:/a:gnu:a2ps
2 abc2ps;cpe:/a:abc2ps:abc2ps
3 abcm2ps;cpe:/a:jef_moine:abcm2ps
4 abcmidi;cpe:/a:abcmidi:abcmidi
5 abiword;cpe:/a:abisource:community_abiword
```

AUTOMATED VULNERABILITY INFERENCE

1. Take CVE, match CPE name to Debian package.
2. Parse CVE summary and extract vuln filename.
3. Find clones of package with similar filename.
4. Trim dynamically linked clones.
5. Is vuln affected clone already being tracked?

EXAMPLE

○ By CVE

National Cyber-Alert System

Vulnerability Summary for CVE-2010-0205

Original release date: 03/03/2010

Last revised: 11/18/2010

Source: US-CERT/NIST

Overview

The png_decompress_chunk function in pngutil.c in libpng 1.0.x before 1.0.53, 1.2.x before 1.2.43, and 1.4.x before 1.4.1 does not properly handle compressed ancillary-chunk data that has a disproportionately large uncompressed representation, which allows remote attackers to cause a denial of service (memory and CPU consumption, and application hang) via a crafted PNG file, as demonstrated by use of the deflate compression method on data composed of many occurrences of the same character, related to a "decompression bomb" attack.

```
$ Clonewise find-bugs CVE-2010-0205
# SUMMARY: The png_decompress_chunk function in pngutil.c in libpng
# 1.0.53, 1.2.x before 1.2.43, and 1.4.x before 1.4.1 does not prop
# compressed ancillary-chunk data that has a disproportionately lar
# representation, which allows remote attackers to cause a denial o
# (memory and CPU consumption, and application hang) via a crafted
# demonstrated by use of the deflate compression method on data com
# occurrences of the same character, related to a "decompression bo
#
# CVE-2010-0205 relates to a vulnerability in package libpng.
# The following source filenames are likely responsible:
#     pngutil.c
#
# The following package clones are tracked in the embedded-code-cop
# database. They have not been fixed.
#
libpng CLONED_IN_SOURCE ia32-libs <unfixed> CVE-2010-0205
```

IMPLEMENTATION AND EVALUATION

IMPLEMENTATION

- ◉ 3,500 Lines of C++ and shell scripts.

- ◉ Open Source

<http://www.github.com/silviocesare/Clonewise>



github
SOCIAL CODING



FILENAME AS FEATURES

- ◉ Ubuntu Linux
- ◉ 3,077,063 unique filenames.
- ◉ Follows inverse power law distribution.
- ◉ R square value of regression analysis 0.928.



ESTABLISHING GROUND TRUTH

- ◉ Debian Linux embedded-code-copies.txt.
 - Not really machine readable.
 - Cull entries which we can't match to packages.
 - 761 labelled positives.
- ◉ Negatives any packages not in positives
 - 475780 generated labelled negatives.

PACKAGE CLONE DETECTION

- ◉ Identified 34 previously unknown clones in Debian.
 - Lots more to do.
- ◉ Statistical classification
 - Random Forest gave best accuracy.
 - Increasing the decision threshold reduces FPs.
 - Predict 3 FPs in 10,000 classifications.
 - More likely an upper limit.

ACCURACY OF STATISTICAL CLASSIFICATION

Classifier	TP/FN	FP/TN	TP Rate	FP Rate
Naïve Bayes	439/322	484/56296	57.69%	0.85%
Multilayer Perceptron	204/557	48/56732	26.81%	0.08%
C4.5	523/238	86/56694	68.73%	0.15%
Random Forest	533/228	60/56720	70.04%	0.11%
Random Forest (0.8)	446/315	15/56765	58.61%	0.03%

EFFICIENCY OF CLONE DETECTION

- ⦿ 4 hours on an Amazon HPC cluster.
- ⦿ MPI_Scatter to do static job assignment was inefficient.
 - Better to consume from a work queue.
- ⦿ Need to use multicore to balance load.

VULNERABILITIES DETECTED

Package	Embedded Package	Package	Embedded Package
OpenSceneGraph	lib3ds	boson	lib3ds
mrpt-opengl	lib3ds	libopenscenegraph7	lib3ds
mingw32-OpenSceneGraph	lib3ds	libfreeimage	libpng
libtlen	expat	libfreeimage	libtiff
centerim	expat	libfreeimage	openexr
mcabber	expat	r-base-core	libbz2
udunits2	expat	r-base-core-ra	libbz2
libnodeupdown-backend-ganglia	expat	lsb-rpm	libbz2
libwmf	gd	criticalmass	libcurl
kadu	mimetex	albert	expat
cgit	git	mcabber	expat
tkimg	libpng	centerim	expat
tkimg	libtiff	wengophone	gaim
ser	php-Smarty	libpam-opie	libopie
pgpoolAdmin	php-Smarty	pysol-sound-server	libmikod
sepostgresql	postgresql	gnome-xcf-thumbnailer	xcftool
		plt-scheme	libgd

DISCUSSION,
RELATED WORK,
FUTURE WORK AND
CONCLUSION

PRACTICAL CONSEQUENCES

- ◉ Write access to Debian's security tracker.
- ◉ Red Hat embedded code copies wiki created.
- ◉ Debian plan to integrate Clonewise into infrastructure.

```
Revision 15537 - (view) (download) (annotate) - (select for diffs)  
Modified Fri Oct 29 04:31:39 2010 UTC (20 months ago) by silvio-guest  
File length: 57330 byte(s)  
Diff to previous 15535  
  
gnome-xcf-thumbnailer embeds xcf-tools and has an outstanding cve.
```

```
Revision 15535 - (view) (download) (annotate) - (select for diffs)  
Modified Thu Oct 28 02:38:42 2010 UTC (20 months ago) by silvio-guest  
File length: 57277 byte(s)  
Diff to previous 15532  
  
libpng is embedded in  
doxygen  
gdal  
libtk-img  
htmldoc  
libf1tk1.1  
syslinux-common (this appears unfixable)  
texlive-bin  
vice  
VisualBoyAdvance  
  
This list is still incomplete. All the packages above link the s
```


REFERENCING CVEs IN ADVISORIES

- Red Hat reference CVEs of embedded libs.
- Not every vendor does.
- It would be nice if CVE supported this.

CVE-ID	
CVE-2011-3026 (under review)	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
Integer overflow in libpng, as used in Google Chrome before 17.0.963.56, allows remote attackers to cause a denial of service or possibly have unspecified other impact via unknown vectors that trigger an integer truncation.	
References	



Critical: xulrunner security update

Advisory: RHSA-2012:0143-1

Type: Security Advisory

Severity: Critical

Issued on: 2012-02-16

Last updated on: 2012-02-16

Affected Products:

- RHEL Desktop Workstation (v. 5 client)
- Red Hat Enterprise Linux (v. 5 server)
- Red Hat Enterprise Linux Desktop (v. 5 client)
- Red Hat Enterprise Linux Desktop (v. 6)
- Red Hat Enterprise Linux HPC Node (v. 6)
- Red Hat Enterprise Linux Server (v. 6)
- Red Hat Enterprise Linux Server AUS (v. 6.2)
- Red Hat Enterprise Linux Server EUS (v. 6.2.z)
- Red Hat Enterprise Linux Workstation (v. 6)

CVEs (cve.mitre.org): [CVE-2011-3026](#)

EMBEDDED CODE COPIES VERSUS CODE REUSE

- Clonewise detects code reuse.
- If zlib embedded in packages X and Y:
 - Clonewise detects clones between all X, Y, and zlib.
- What we really want to know is:
 - X is not cloned in Y.
 - Zlib is cloned in X and Y.
- Mitigation
 - Clone detection on known embedded libraries.

RELATED WORK

◉ Debian Linux zlib audit in 2005

◉ Plagiarism detection

- Attribute counting
- Structure-based

◉ Code clone detection

- Tokenization
- Abstract syntax trees

Halstead complexity measures

From Wikipedia, the free encyclopedia

Halstead complexity measures are *software metrics* introduced by Maurice Howard Halstead in 1977 for software development. Halstead makes the observation that metrics of the software should reflect the effort but be independent of their execution on a specific platform. These metrics are therefore computed statically. Halstead's goal was to identify measurable properties of software, and the relations between them. They are (like the volume, mass, and pressure of a gas) and the relationships between them (such as the gas laws).

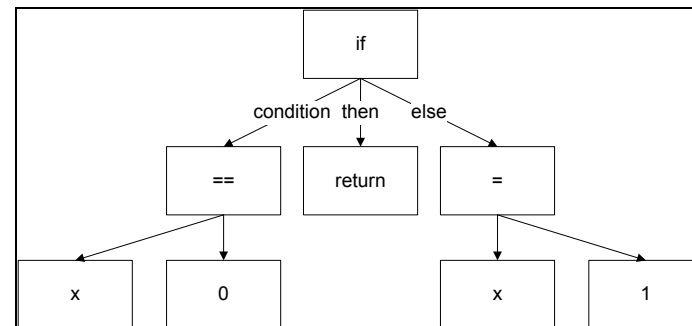
Contents [hide]

- 1 Calculation
- 2 References
- 3 See also
- 4 External links

Calculation

For a given problem, Let: First we need to compute the following numbers, given the program:

- η_1 = the number of distinct operators
- η_2 = the number of distinct operands
- N_1 = the total number of operators
- N_2 = the total number of operands



FUTURE WORK

- ◉ Source repositories
 - Sourceforge
 - Github
- ◉ Other OSs - BSD etc
- ◉ Integration into build/packaging systems?
- ◉ Integration into Debian Linux infrastructure.



CONCLUSION

- ◉ Vendors have 10,000+ packages.
- ◉ How to audit for clones?
- ◉ Clonewise can provide a solution.
- ◉ And help improve security.
- ◉ <http://www.FooCodeChu.com>



Remember to complete the Black Hat speaker feedback survey.