Security of Open Source Web Applications

Maureen Doyle, James Walden
Northern Kentucky University

NKU NORTHERN KENTUCKY UNIVERSITY

Students: Grant Welch, Michael Whelan

Acknowledgements: Dhanuja Kasturiratna

Outline

1. Research Objective
2. Related Work
3. Results
4. Analysis
5. Future Work

Research Objective

Goal: Identify effects of time, size, complexity, and change rate on vulnerability density (VD) of open source web applications.

Research questions:

1. What is the current state of open source web app security?
2. Can size or complexity predict VD?
3. Can churn or deletions predict VD?

Measuring Vulnerabilities

Reported Vulnerabilities in NVD or OSVD

- Coarse-grained time evolution.
- Difficult to correlate with revision.
- Undercounts actual vulnerabilities.

Dynamic Analysis

- Expensive.
- False positives and negatives.
- Requires installation of application.

Static Analysis

- False positives and negatives.
- Static Analysis Vulnerability Density = vulns/kloc.

Code Metrics

Size measure

- Source Lines of Code (SLOC)

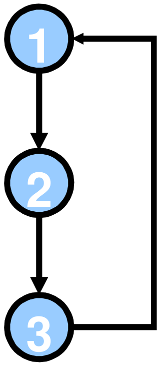
Complexity measures

- Cyclomatic Complexity
- Nesting Complexity
- Maximum, average, total

Change measures

- Churn = lines added + changed
- Lines deleted

1. do loop
2. stmt
3. end loop



CC = E – N + 2 P
= 3 – 3 + 2*1

Related Work

Static Analysis

- Nagappan and Ball, ICSE 2005a
- Coverity Open Source Report 2008
- Fortify Open Source Security Study 2008

Complexity and Change Metrics

- Nagappan and Ball, ICSE 2005b
- Nagappan, Ball, and Zeller, ICSE 2006
- Shin and Williams, QoP 2008

Samples

Selection process

- PHP web applications from freshmeat.net.
- Subversion repository with 100 weeks of revisions.

Revisions

- One revision selected per week for analysis.
- Changes between individual revisions too small.

Range of projects

- 14 projects met selection criteria.
- 5,800 to 388,000 lines of code.
- Removing highest + lowest, range 25-150 kloc.

Achievo

phpMyAdmin

dotProject
Project Management Software

WIKIPEDIA
The Free Encyclopedia

OBM

smarty TM
TEMPLATE ENGINE

ROUND CUBE
WEBMAIL

SquirrelMail
WEBMAIL FOR NUTS

Results

Overall security improvement.

- first week average: 8.88 vulns/kloc
- final week average: 3.30 vulns/kloc

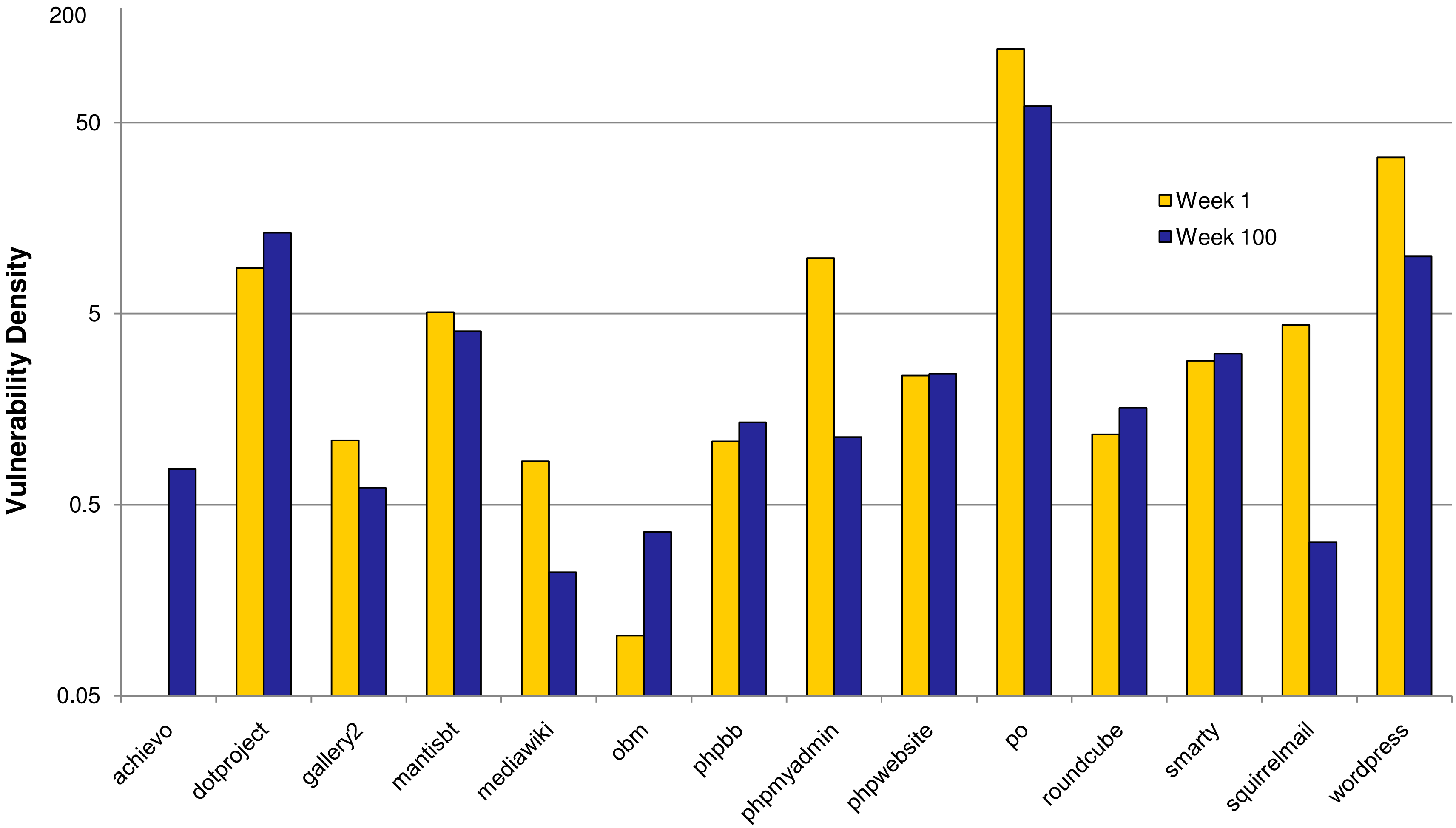
High compared to Coverity's 0.30 SAVD.

- Language differences: C/C++ vs. PHP.
- Vulnerability diffs: buffer overflows vs XSS/SQL.

No correlation with NVD vulnerabilities.

- NVD correlated with freshmeat popularity.

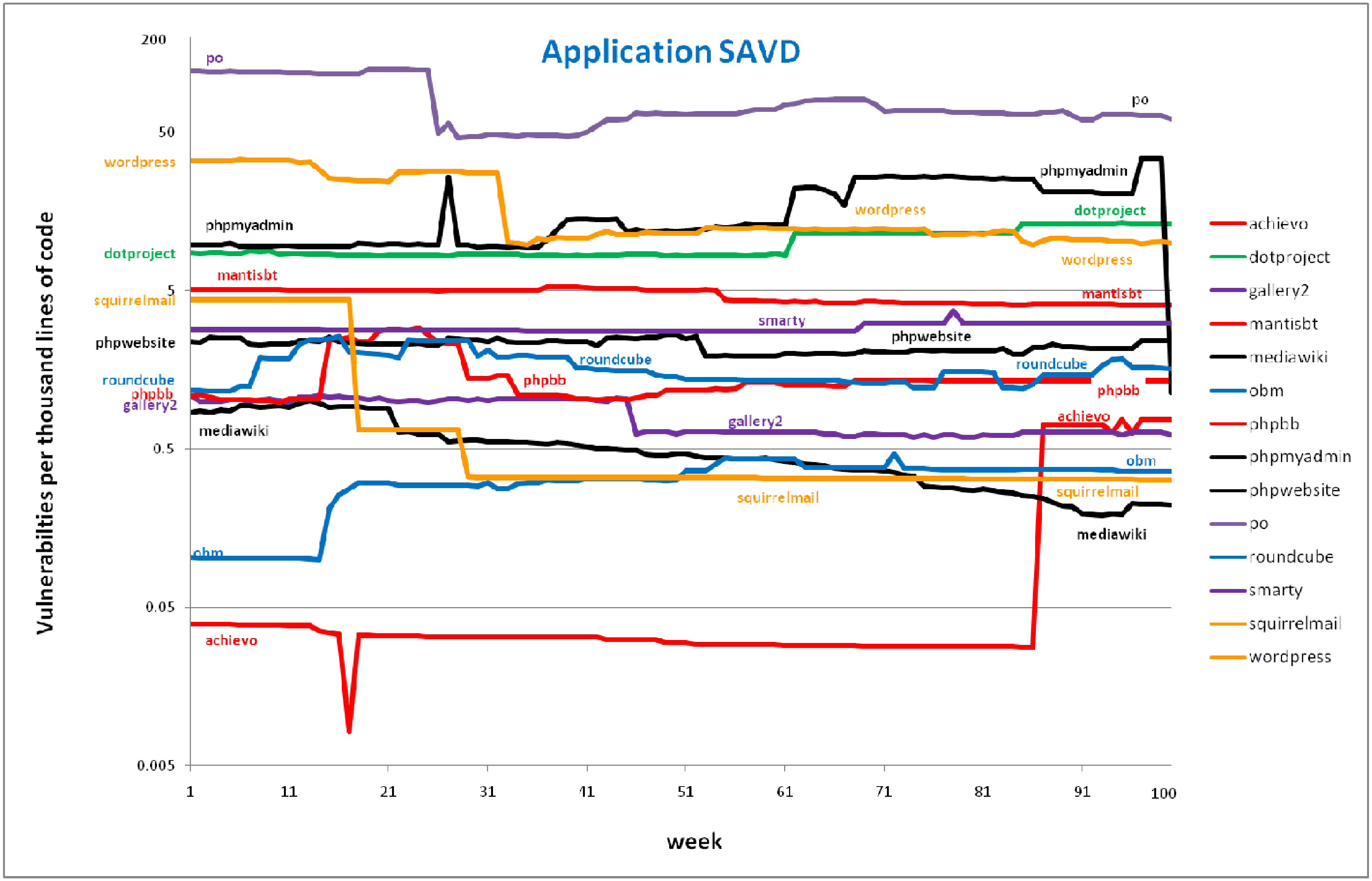
Variation between Web Apps



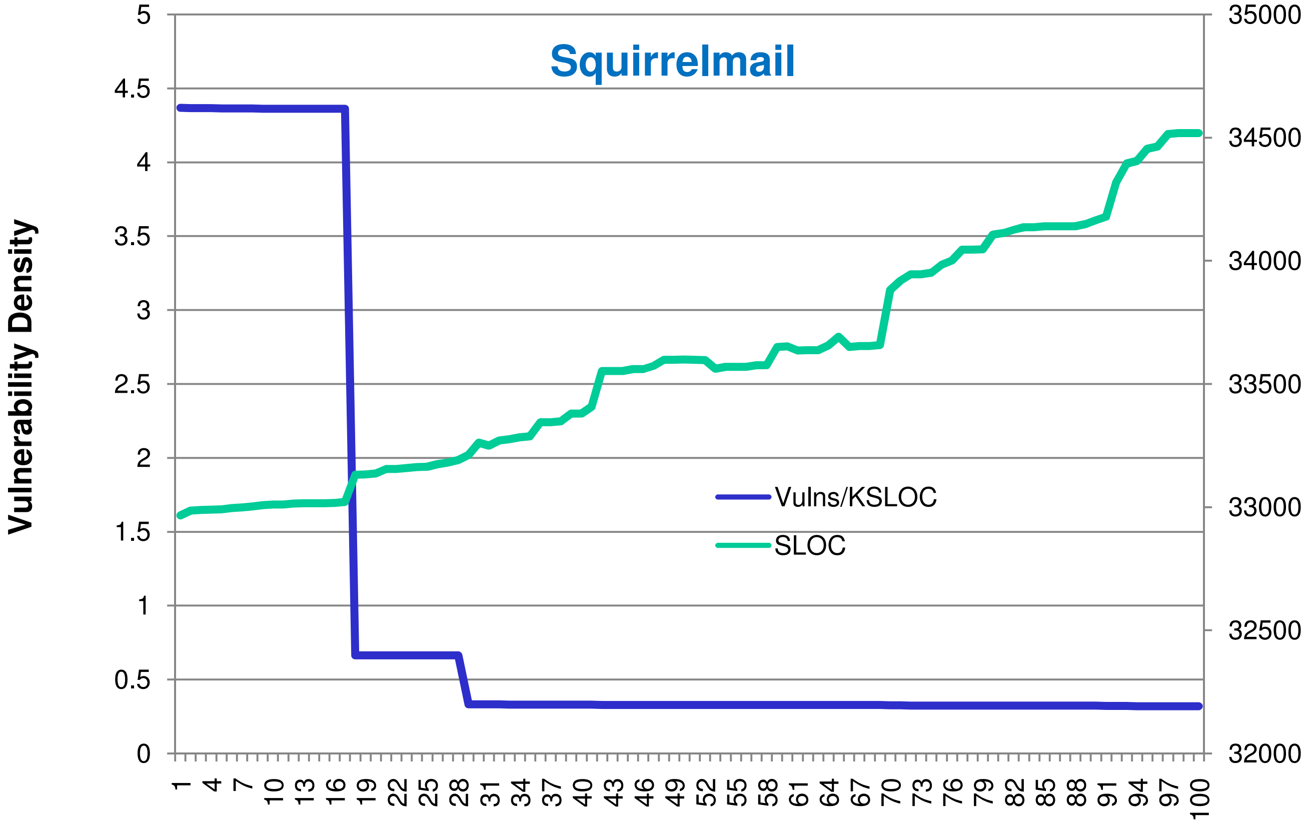
week 1: projects ranged from 0 to 121.4 vulns/kloc

week 100: projects varied from 0.20 to 60.86 vulns/kloc

Variation between Web Apps



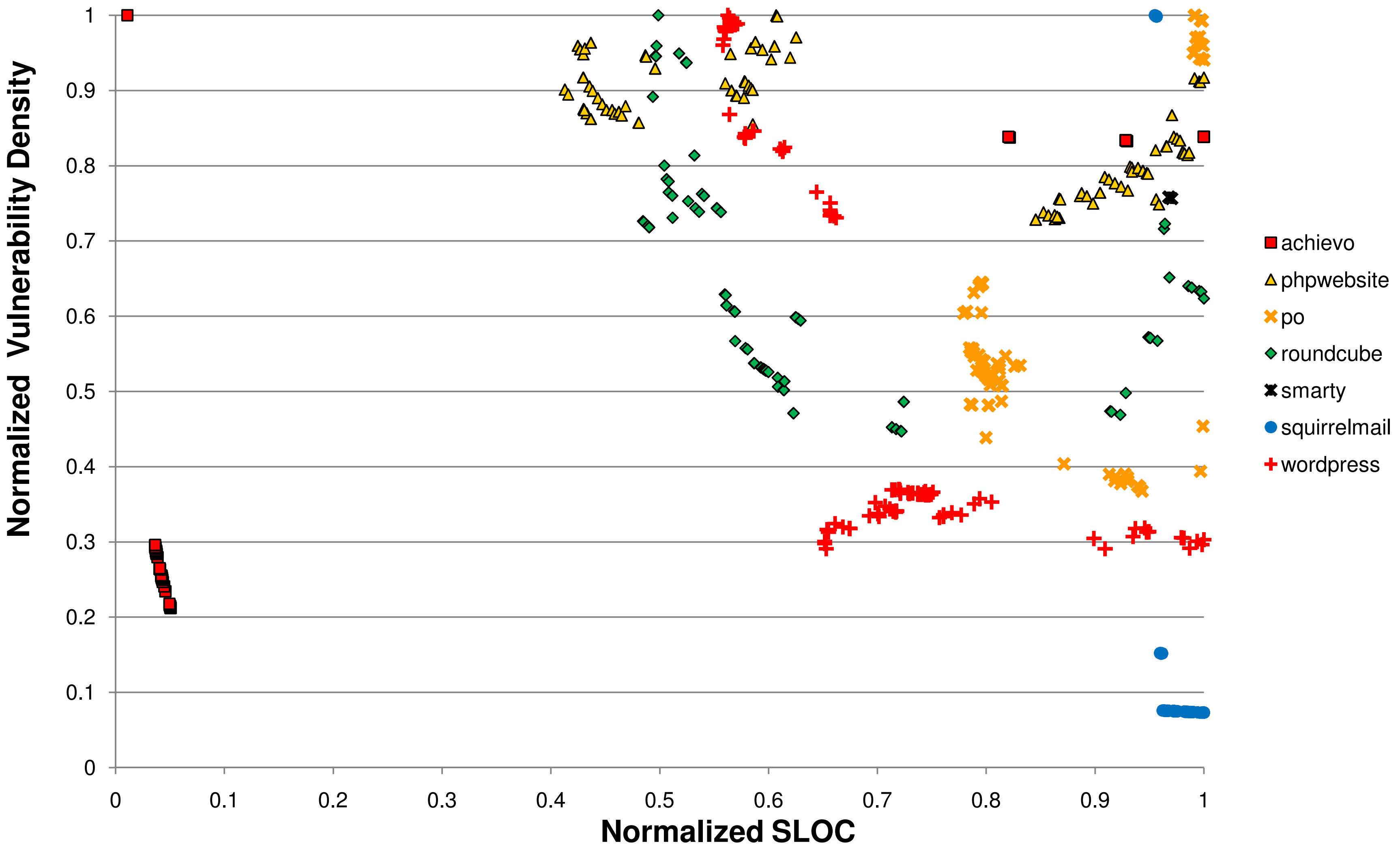
Example: Addressing Security Issues



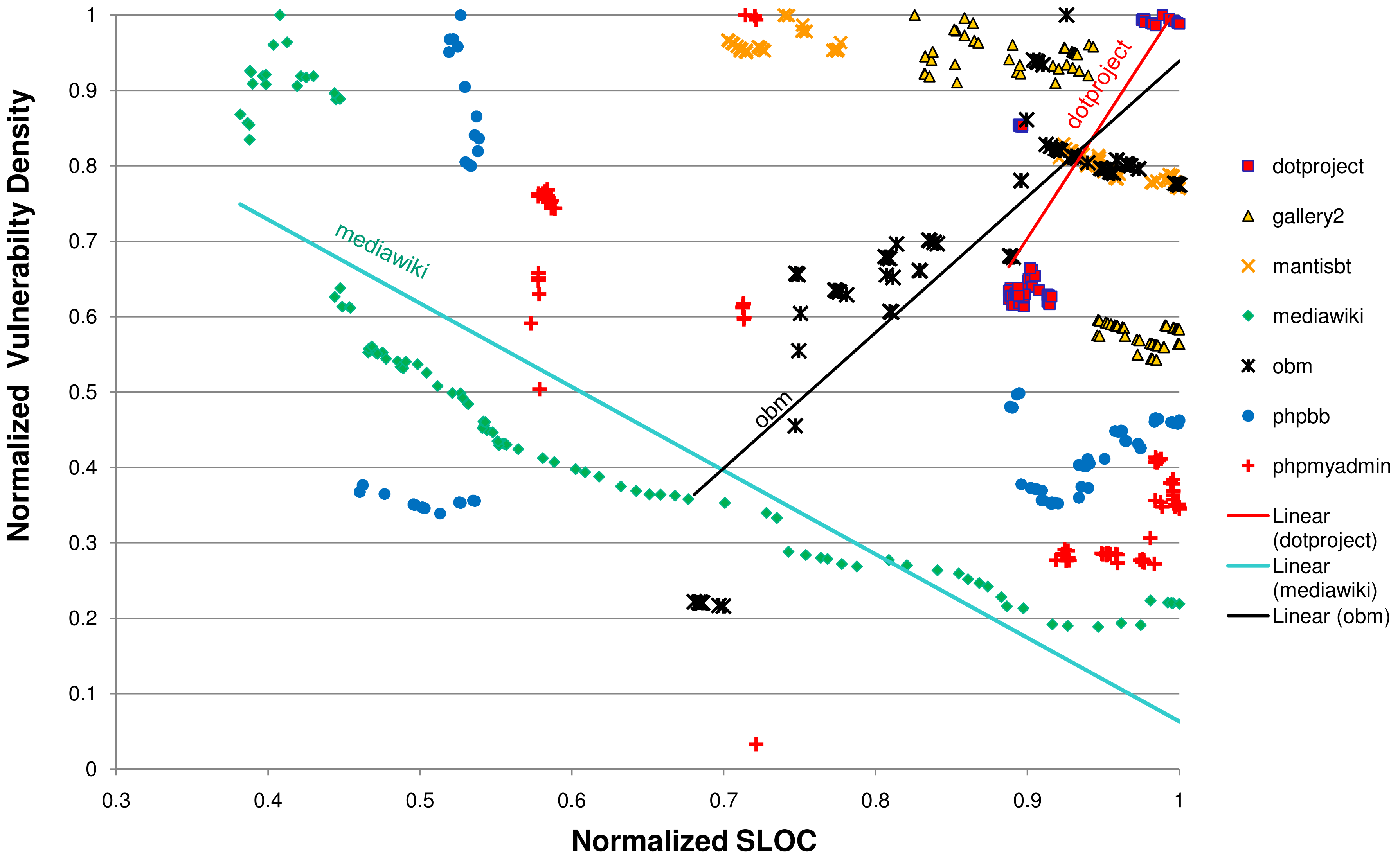
1st drop: New data sanitization and input handling.

2nd drop: Fixed CVE-2006-3174 vulnerabilities.

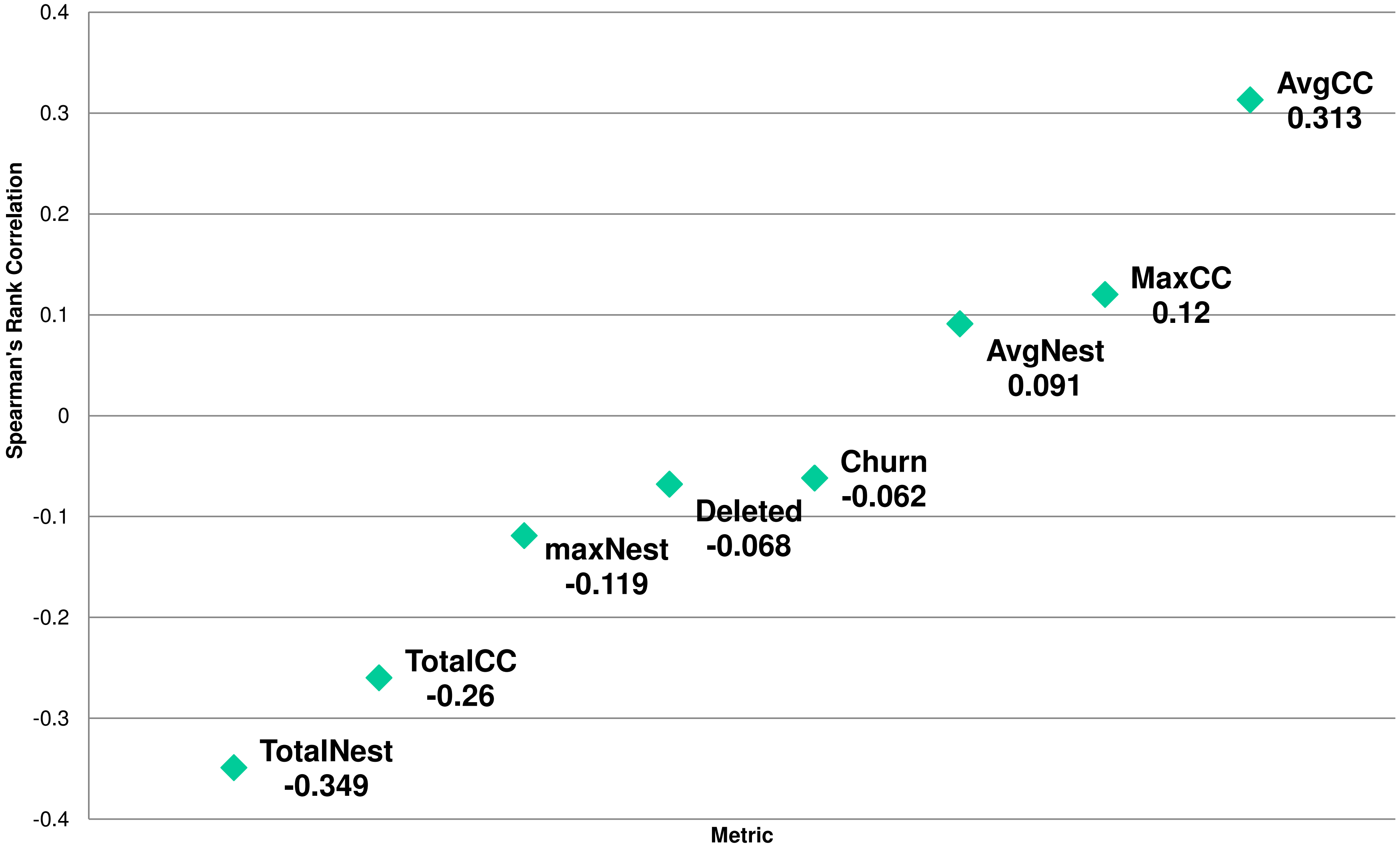
Small Projects (<50K SLOC) SLOC versus SAVD



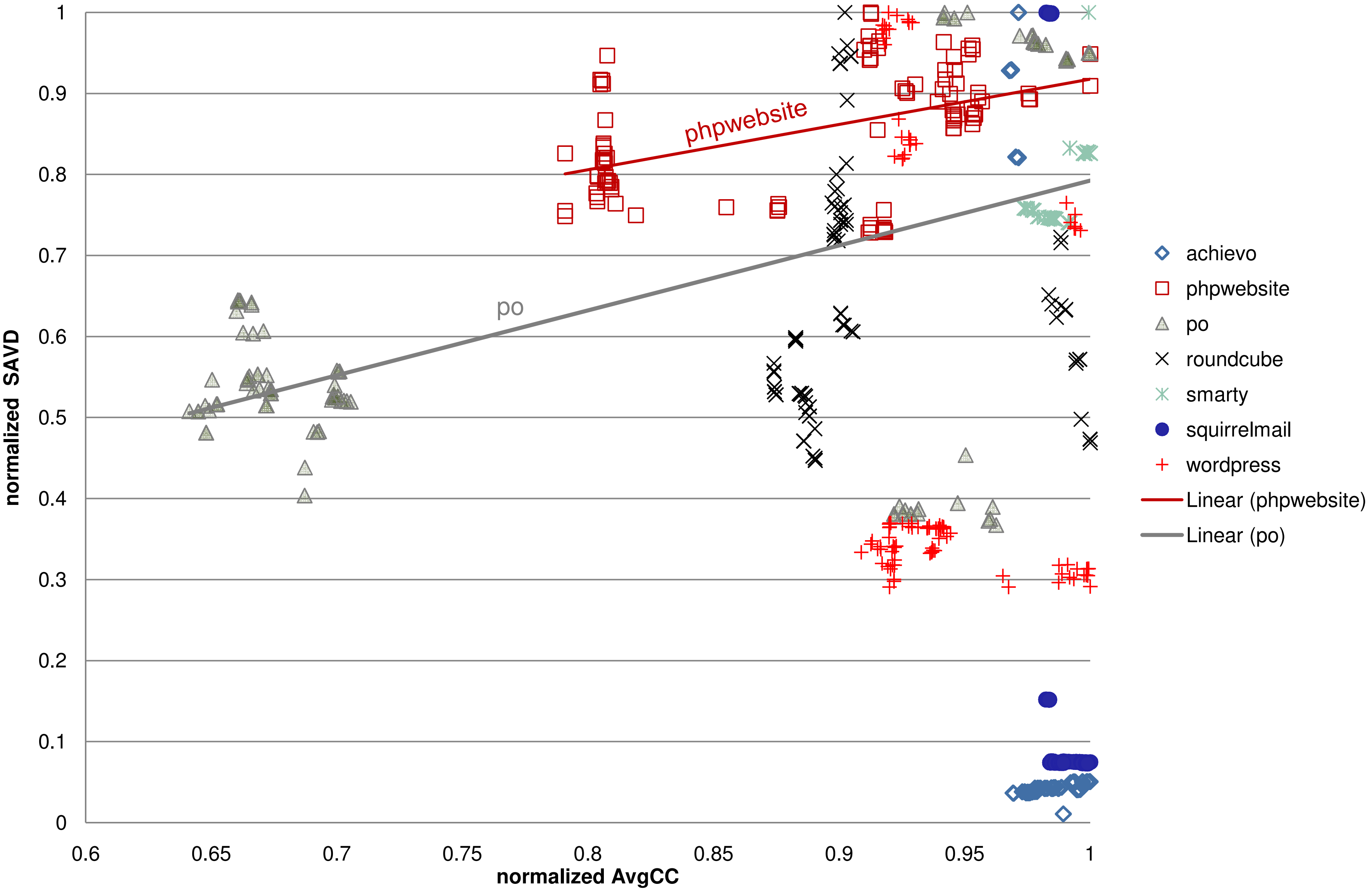
Large Projects (> 50K SLOC)
SLOC versus SAVD (r = 0.27, 0.99 sig)



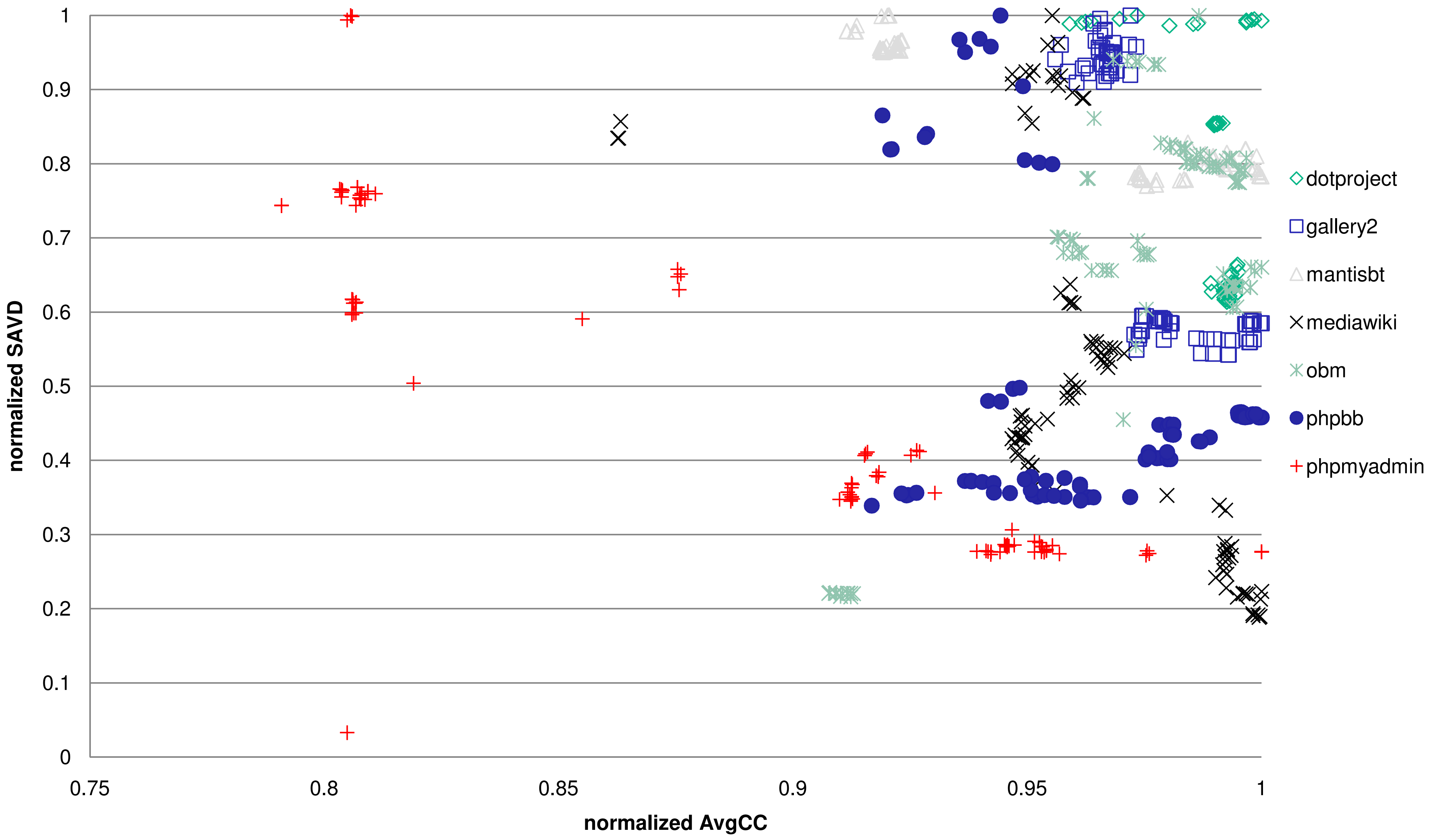
Metric Analysis
Spearman's Correlation, r, to SAVD



Small Projects (<50K SLOC)
AvgCC vs SAVD



Large Project (>50K SLOC)
AvgCC vs SAVD



Conclusions

No single metric is predictive for SAVD.

- Similar to Naggapan and Ball's results for defects of five different Windows projects.

Complexity is an indicator for SAVD.

- Supports Shin's finding of weak correlations of CC and NC with vulns in Mozilla JSE.

Churn is not an indicator for SAVD.

- Different from Naggapan and Ball's results for pre-release defect density in W2k3.

Future Work

Analyzing vulnerability type information

- 14 different types of vulnerabilities
- 5 severity levels

Why does app security vary so much?

- Analyze security processes for each app.

How do we validate SAVD measurement?

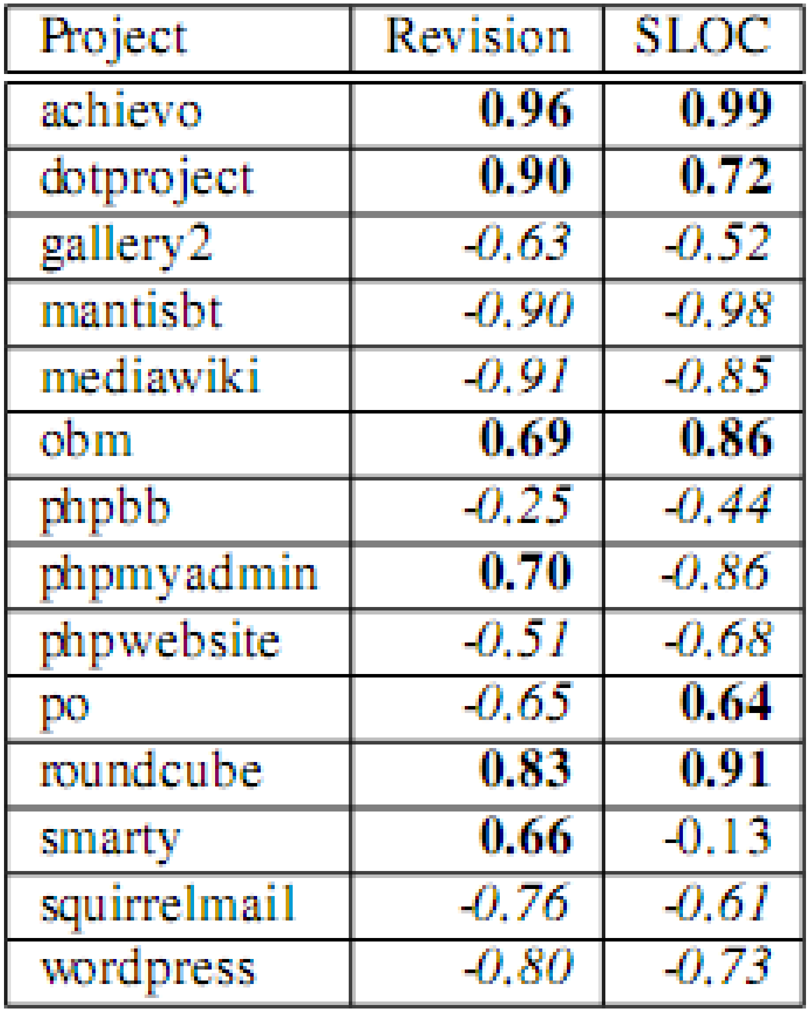
- NVD vulnerability count correlates with popularity.

Java web applications

- How does Java SAVD compare with PHP SAVD?
- How do trends compare between Java and PHP?
- More software metrics available for Java.

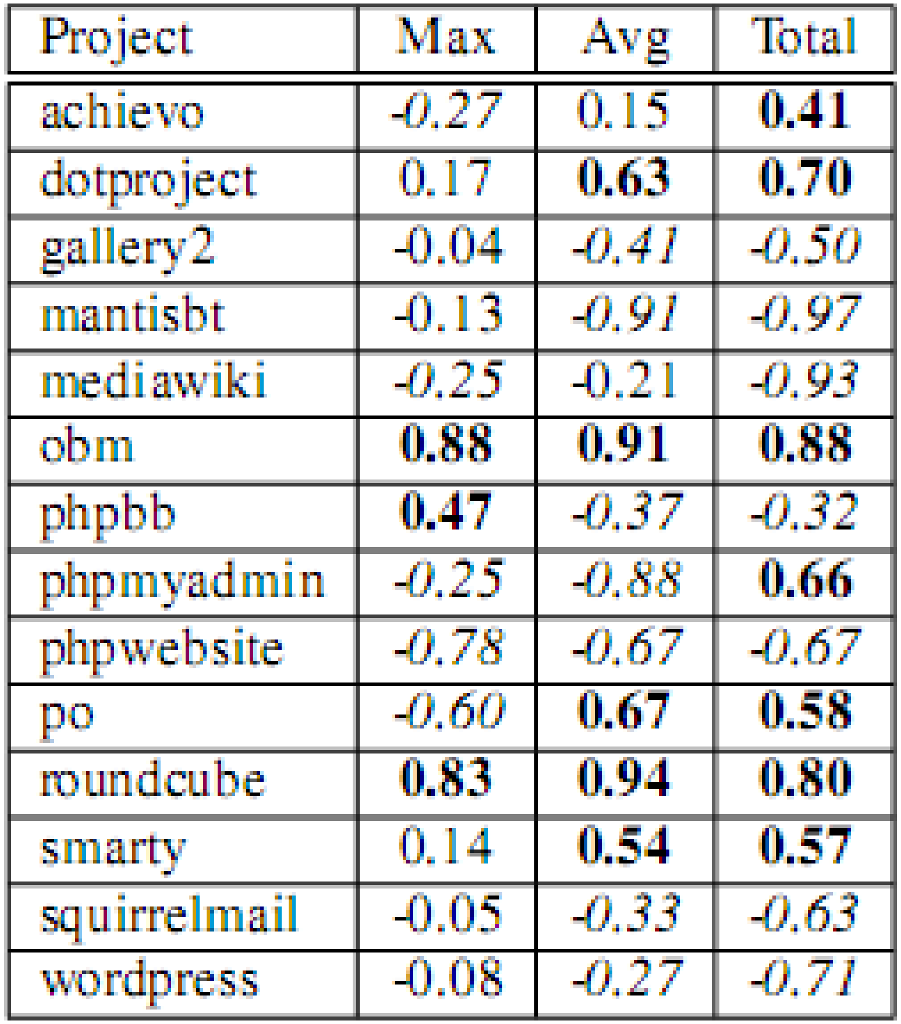
Extra Slides

SAVD vs Time and Size



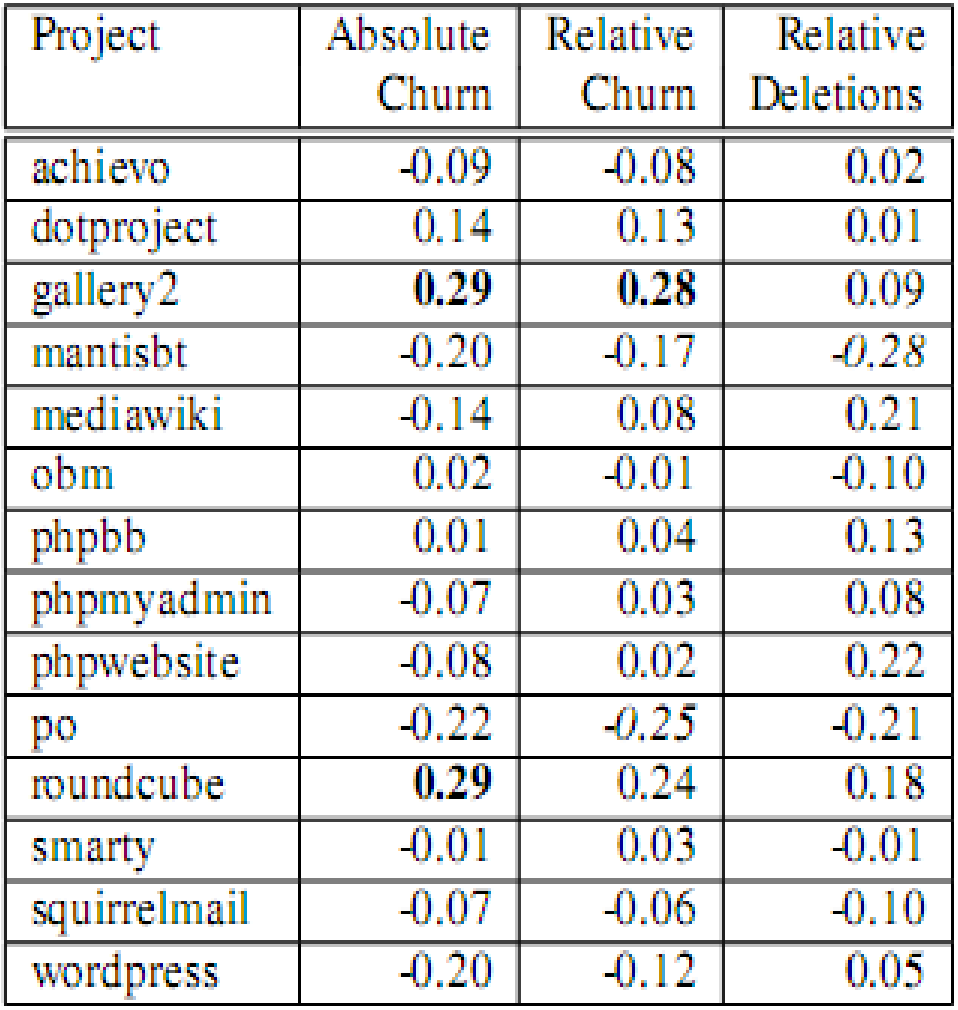
Project	Revision	SLOC
achievo	**0.96**	**0.99**
dotproject	**0.90**	**0.72**
gallery2	*-0.63*	*-0.52*
mantisbt	*-0.90*	*-0.98*
mediawiki	*-0.91*	*-0.85*
obm	**0.69**	**0.86**
phpbb	*-0.25*	*-0.44*
phpmyadmin	**0.70**	*-0.86*
phpwebsite	*-0.51*	*-0.68*
po	*-0.65*	**0.64**
roundcube	**0.83**	**0.91**
smarty	**0.66**	-0.13
squirrelmail	*-0.76*	*-0.61*
wordpress	*-0.80*	*-0.73*

SAVD vs. Nesting



Project	Max	Avg	Total
achievo	*-0.27*	0.15	**0.41**
dotproject	0.17	**0.63**	**0.70**
gallery2	-0.04	*-0.41*	*-0.50*
mantisbt	-0.13	*-0.91*	*-0.97*
mediawiki	*-0.25*	-0.21	*-0.93*
obm	**0.88**	**0.91**	**0.88**
phpbb	**0.47**	*-0.37*	*-0.32*
phpmyadmin	*-0.25*	*-0.88*	**0.66**
phpwebsite	*-0.78*	*-0.67*	*-0.67*
po	*-0.60*	**0.67**	**0.58**
roundcube	**0.83**	**0.94**	**0.80**
smarty	0.14	**0.54**	**0.57**
squirrelmail	-0.05	*-0.33*	*-0.63*
wordpress	-0.08	*-0.27*	*-0.71*

SAVD vs. Churn



Project	Absolute Churn	Relative Churn	Relative Deletions
achievo	-0.09	-0.08	0.02
dotproject	0.14	0.13	0.01
gallery2	**0.29**	**0.28**	0.09
mantisbt	-0.20	-0.17	*-0.28*
mediawiki	-0.14	0.08	0.21
obm	0.02	-0.01	-0.10
phpbb	0.01	0.04	0.13
phpmyadmin	-0.07	0.03	0.08
phpwebsite	-0.08	0.02	0.22
po	-0.22	*-0.25*	-0.21
roundcube	**0.29**	0.24	0.18
smarty	-0.01	0.03	-0.01
squirrelmail	-0.07	-0.06	-0.10
wordpress	-0.20	-0.12	0.05