SETTLETOP
SOFTWARE RISK RATINGS REPORT

Customer Codename One
Program Codename One
Date September 10, 2020

Overview

Overall Score

522
Fair

Key Insights

COURSE OF ACTION
- Address 2,104 critical and high DISA STIG and OWASP Top 10 issues
- Address 4,824 blocker, critical and major coding

TECHNICAL HEALTH
- 1 critical and 2 emerging cores found
- Cores contain over 14% of the total files
- Cores contain nearly 32% of the total lines of code

ECONOMIC HEALTH
- 78% of every dollar invested is wasted
- Cost to develop a feature is ~2.5 times as much as that of benchmark projects

Application Description

Name of Project	Codename One
License	GPL v2.0
Distribution Model	Development Tool
Vendor	Codename One LTD
# of Developers	31
Application Age	8 Years
Type of app	Development Tool

Application Structure

Directories	350
Files	1,885
Duplicate Files	561
Classes	3,084
Statements	206,331
Functions	36,242
Duplicate Blocks	3,617

Line Description

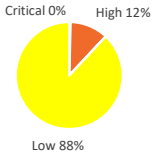
Language	Java
Line	394,821
Comments	116,847
Blank Lines	157,503
Duplicate Lines	121,981
Unique Lines	547,190
Total	669,171

Application Security	Architecture Quality	Code Quality	Economic Health	Open Source Composition
659 Good	363 Poor	373 Poor	450 Fair	767 Good

Application Security

OWASP Top Ten 2017

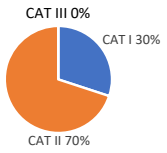
	Count	Issues
Critical	1	2
High	4	60
Medium	0	0
Low	4	457



MICRO FOCUS

Application Security and Development STIG, version 4.9

	STIGs	Critical	High	Medium	Low
CAT I	8	1	624	0	537
CAT II	24	1	1,456	0	6,103
CAT III	0	0	0	0	0
Total	32	2	2,080	0	6,640



Architectural Quality

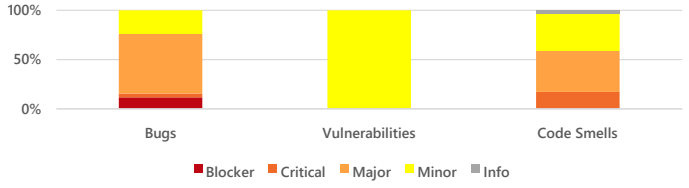
Critical Cores	Emerging Cores	Size of Largest Core	% LOC inside a Core	Files impacted by a change
1	2	176	32%	49%
Good	Fair	Poor	Very Poor	Very Poor

silverthread

Code Quality

Coding Violations

	Blocker	Critical	Major	Minor	Info	Total
Bugs	98	34	509	203	0	844
Vulnerabilities	0	2	0	810	0	812
Code Smells	211	4,479	11,194	10,228	1,034	27,146
Total	309	4,515	11,703	11,241	1,034	28,802



sonarqube

Code Complexity

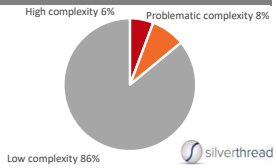
	Average	Total
Cyclomatic complexity	50	94,612
Cognitive complexity	52	97,584

sonarqube

File Complexity

	Total	% of Total
High complexity	108	6%
Problematic complexity	157	8%
Low complexity	1,620	86%

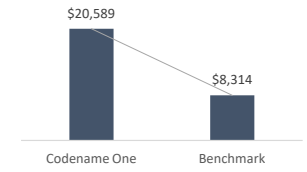
LOCs in Files with varying degrees of complexity



silverthread

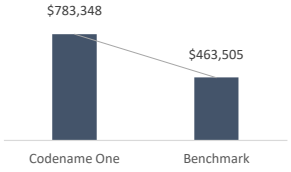
Economic Health

Cost
Cost to produce features



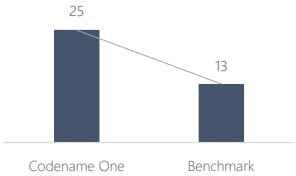
2.5 X delta

Waste
Money wasted per $1m invested



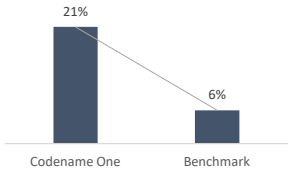
$319,843 + delta

Features
Days to develop feature



1.9 X delta

Risk
% of bug LOC discovered or added per feature LOC



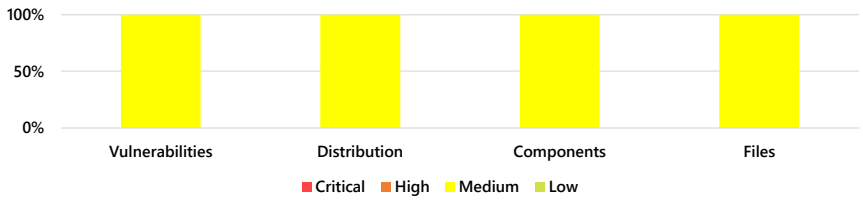
3.5 X delta

silverthread

Open Source Composition

Open Source Security

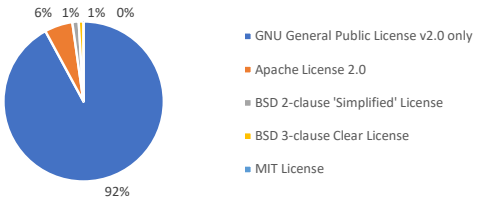
	Critical	High	Medium	Low	Total
Vulnerabilities	0	0	2	0	2
Distribution	0%	0%	100%	0%	100%
Components	0	0	1	0	1
Files	0	0	1	0	1



FOSSID

Top 5 Licenses

License	Components	Matches	% of Total
GNU General Public License v2.0 only	1	2,350	92%
Apache License 2.0	1	144	6%
BSD 2-clause 'Simplified' License	1	34	1%
BSD 3-clause Clear License	1	23	1%
MIT License	1	1	0%
Total	5	2,552	100%



FOSSID

SettleTop SetVIEW Report

User Guide

Preface

If you have questions or comments about using this product, contact SettleTop support by email using the following email address:
support@settletop.com

For More Information

For more information about SettleTop products:
https://www.settletop.com

Introduction

SetVIEW is a software quality scorecard solution that provides a 'high-level' assessment of a software asset by leveraging best of breed 3rd party tools. In doing so, SetVIEW provides a unified report with a holistic view of the quality of an organizations software assets across varying disciplines including, but not limited to, Application Security, Architecture Quality, Code Quality, Open Source Composition and Software Economics.

Key Quality Metrics (KQM) from each vendor tool and discipline are assessed and scored, using a proprietary algorithm, to benchmark and trend the results over time. The report provides an objective measure of risks within a given codebase.

This guide is intended for any individual responsible for monitoring software assets for changes in quality and health issues.

Understanding the Report

Overview

This section provides high level information including the cumulative score, scoring at the discipline level (e.g. Security or Code Quality), Key Insights, and informative application level details, such as number of files, lines of code, application age, etc.

Scoring

The SetVIEW score is calculated using the SettleTop Software Quality Scorecard. Scores will fall between 0 and 1,000, where 1,000 represents an optimal software asset, with no evidence of risk found across any of the analysis tools.

Scoring Values

Rating	Score
Very Poor	0-200
Poor	201-400
Fair	401-600
Good	601-800
Excellent	801-1,000

Overall Score

The 'Overall Score' is a composite of the scores from each of the disciplines, where each metric is assessed both individually, as well as its implications on the software asset as a whole.

Disciplines and Vendor Tools

There are many factors to take into consideration when evaluating the quality of a software asset. For the purposes of the SetVIEW report, focus has been put on vendor tools which provide coverage for following areas or disciplines:

- Application Security
- Architecture Quality
- Code Quality
- Open Source Composition
- Software Economics

Click here for further information regarding which 3rd party vendor tools were leveraged to generate this report.

Application Security

Application Security encompasses measures taken to improve the security of an application often by finding, fixing, and preventing security vulnerabilities.

Architecture Quality

Architecture Quality considers the modularity, cyclicality, and complexity of a software asset to determine potential growing pains and maintainability issues.

Code Quality

Code Quality refers to how well written the code is. Major contributing factors to code quality include the number of bugs, security vulnerabilities and code smells, file/application complexity and code duplication.

Economic Health

Economic Health is the measurement of the cost to maintain or build upon a software asset.

Open Source Composition

Open source composition takes inventory of: the open source packages (OSP) found within the software asset; any known security vulnerabilities associated with the OSP; the license(s) the OSP is distributed under; and all out of date or expired OSPs.

Tool	Version	Discipline	Description
FossID	20.1.2	Open Source Composition	FossID is a Software Composition Analysis tool that scans your code for open source licenses and vulnerabilities and gives you full transparency and control of your software products and services.
Micro Focus	19.2	Application Security	Micro Focus Fortify is a static analysis tool used to pinpoint root causes of security vulnerabilities in source code.
Silverthread	8.81	Architecture Quality Economic Health	Silverthread provides deep insight into the technical health of a specific system and explore both current and projected economic outcomes.
SonarQube	8.4	Code Quality	SonarQube is a platform for inspection of code quality to perform automatic reviews with static analysis of code to detect bugs, code smells, and security vulnerabilities on 20+ programming languages.

Term	Definition
Blocker	Bug with a high probability to impact the behavior of the application in production: memory leak, unclosed JDBC connection, ... The code MUST be immediately fixed.
Bugs	An issue that represents something wrong in the code. If this has not broken yet, it will, and probably at the worst possible moment. This needs to be fixed. Yesterday.
CAT I	STIG severity code assigned to findings that allow primary security protections to be bypassed, allowing immediate access by unauthorized personnel or unauthorized assumption of super-user privileges. CAT I weaknesses must be corrected before an Authorization to Operate (ATO) is granted.
CAT II	STIG severity code assigned to findings that have a potential to lead to unauthorized system access or activity. CAT II findings can usually be mitigated and will not prevent an Authorization to Operate from being granted.
CAT III	STIG severity code assigned to recommendations that will improve IA posture but are not required for an authorization to operate.
Code Smells	A maintainability-related issue in the code. Leaving it as-is means that at best maintainers will have a harder time than they should making changes to the code. At worst, they'll be so confused by the state of the code that they'll introduce additional errors as they make changes.
Cognitive Complexity	A measure of how difficult a unit of code is to intuitively understand.
Copyleft	Copyleft is a general method for making a program (or other work) free (in the sense of freedom, not “zero price”), and requiring all modified and extended versions of the program to be free as well.
Core	A Core is a cyclical group of files. Theoretically, a Core is any number of files $n \geq 2$ in which the relationship of the system could be mapped as a self-contained loop.
Critical	Either a bug with a low probability to impact the behavior of the application in production or an issue which represents a security flaw: empty catch block, SQL injection, ... The code MUST be immediately reviewed.
Critical Cores	Critical Cores are defined as Cores with 150+ files and have been proven to impose significant negative impacts on the codebase.
Cyclomatic Complexity	A calculation based on the number of paths through the code. Whenever the control flow of a function splits, the complexity counter gets incremented by one. Each function has a minimum complexity of 1. This calculation varies slightly by language because keywords and functionalities do.
DISA	Defense Information Systems Agency (DISA)
Emerging Cores	Emerging Cores are defined as Cores containing 30-150 files and have been proven to cause some financial and developmental impact.
High complexity	Files with a Cyclomatic Complexity of 21 or greater.
Info	Neither a bug nor a quality flaw, just a finding.
KQM	Key Quality Metrics (KQM) are metrics deemed crucial in objectively measuring the quality of a software asset.

Low complexity	Files with a Cyclomatic Complexity between 1-10.
Major	Quality flaw which can highly impact the developer productivity: uncovered piece of code, duplicated blocks, unused parameters, …
Minor	Quality flaw which can slightly impact the developer productivity: lines should not be too long, "switch" statements should have at least 3 cases, …
OWASP Top 10	Open Web Application Security Project (OWASP) Top 10 is a standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications.
Problematic complexity	Files with a Cyclomatic Complexity between 11-20.
STIG	Security Technical Implementation Guides (STIGs) are the configuration standards for DOD [information assurance, or IA] and IA-enabled devices/systems.
Vulnerabilities	A security-related issue which represents a backdoor for attackers.
Weak Copyleft	"Weak copyleft" licenses are generally used for the creation of software libraries, to allow other software to link to the library, and then be redistributed without the legal requirement for the work to be distributed under the library's copyleft license.