بسمه‌تعالی

تحلیل نرم‌افزار بازبینی کد
Coverity Code Advisor

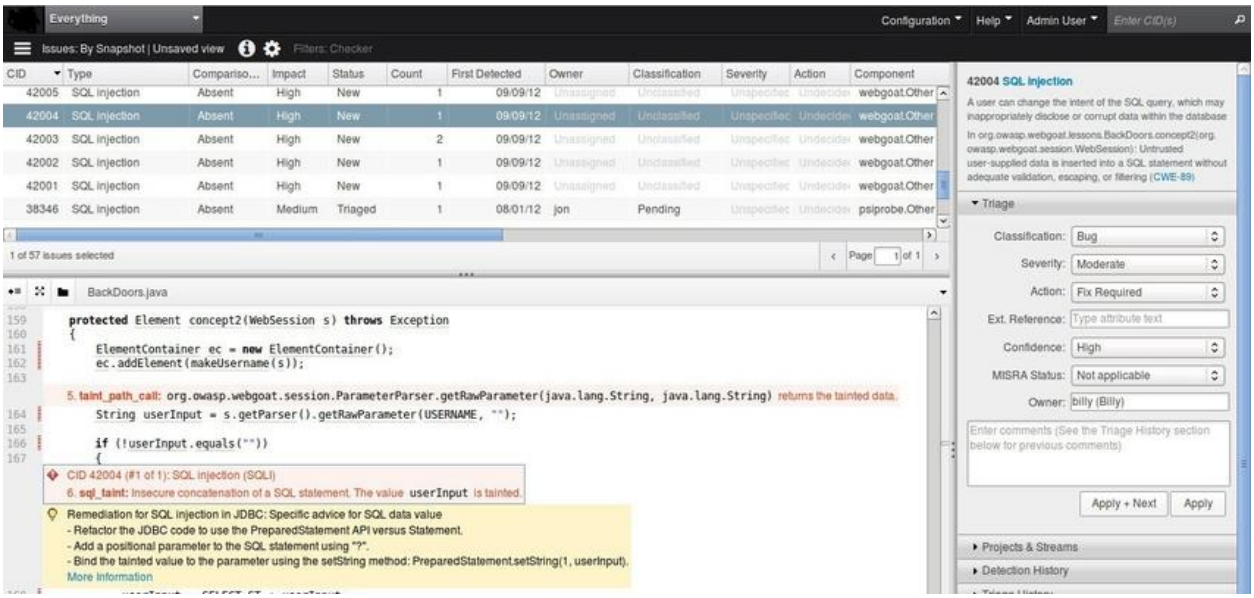
چکیده

برنامه‌ی Coverity Code Advisor، ابزاری قدرتمند برای تحلیل و بازبینی پروژه‌های مختلف نرم‌افزاری است. این برنامه از زبان‌های مختلف برنامه‌نویسی که هر یک کاربردهای گوناگونی دارند پشتیبانی می‌کند. این ابزار آسیب‌پذیری‌های مختلفی را بررسی کرده و تحلیل‌های متنوعی را ارائه می‌دهد.

در کنار پشتیبانی از زبان‌های مختلف برنامه‌سازی، ابزار Coverity از کامپایلرهای مختلف و سیستم‌عامل‌های گوناگون مانند سیستم‌عامل‌های خانواده‌ی ویندوز و لینوکس نیز پشتیبانی می‌کند. همچنین امکان تعریف قواعد و معیارهای جدید برای کشف آسیب‌پذیری‌های خاص نیز وجود دارد.

۱ مقدمه

برنامه Synopsys Static Analysis که با نام Coverity شناخته می‌شود با یافتن مشکلات کیفیتی نرم‌افزار و آسیب‌پذیری‌های مستعد امنیتی باعث کم شدن ریسک به خطر افتادن پروژه‌های نرم‌افزاری و کاهش هزینه‌ها در طول توسعه‌ی نرم‌افزار می‌شود. سازندگان این برنامه، با تجربه‌ی ده ساله‌ی پژوهش در این زمینه و ارائه‌ی روش‌های تشخیص آسیب‌پذیری‌ها محصولی را ساخته‌اند که به حفظ کیفیت و امنیت پروژه‌های نرم‌افزاری کمک می‌کند و بارها در پروژه‌های مختلف استفاده شده است که در بررسی میلیون‌ها خط کد نقش داشته است.



شکل ۱ نمایی از میزکار برنامه‌ی Coverity

۲ عمق و صحت تحلیل‌ها

- نرم‌افزار Coverity بوسیله‌ی یکپارچه بودن با بسیاری از سیستم‌های ساخت[۱]، و تولید بازنمایی‌های قابل اطمینان از کدهای برنامه به درک عمیقی از پروژه و رفتار آن کمک می‌کند.

[۱] Build System

- نرم‌افزار Synopsys Static Analysis تمامی مسیرهای موجود در کد برنامه را پیمایش می‌کند و تمامی خطوط کد را برای کشف نقاط مستعد، مورد بررسی قرار می‌دهد. این برنامه از چندین روش مبتکرانه برای افزایش دقت و عمق تحلیل‌ها استفاده می‌کند.
- با داشتن درک عمیقی از کدها و چارچوب‌های مورد استفاده از آن که با ارائه‌ی تحلیل‌های عمیق و دقیق Coverity فراهم می‌شود، دیگر لازم نیست تا توسعه‌دهندگان زمان و انرژی زیادی را صرف یافتن خطاهای خود کنند و باعث می‌شود که پروژه‌های نرم‌افزای آن‌ها به لحاظ کیفیت و امنیت ارتقا پیدا کند.

۳ سرعت و مقیاس تحلیل‌ها

نرم‌افزار Coverity با قابلیت‌هایی که دارد توانایی تحلیل انواع و اقسام پروژه‌ها در مقیاس‌های مختلف را با سرعت نسبتا زیادی دارد:

- تحلیل‌های موازی به طوری که همزمان به صورت ۱۶هسته‌ای قابلیت اجرای تحلیل‌ها را دارد و کارایی آن نسبت به تحلیل‌های پشت سر هم، ۱۰ برابر بیشتر است.
- در برنامه‌ی Coverity تحلیل‌ها حساس به تغییرات هستند. یعنی هر بار که تغییری در کد برنامه ایجاد می‌شود، کل برنامه مورد تحلیل قرار نمی‌گیرد بلکه تنها محل‌های تغییر و اثرات آن مورد بررسی قرار می‌گیرند. این امر به تسریع تحلیل‌ها کمک زیادی می‌کند. همچنین این امکان وجود دارد که فایل‌ها بدون نیاز به کامپایل شدن هم مورد بررسی قرار گیرند.
- Coverity توانایی بررسی کیفیت کار هزاران توسعه‌دهنده با موقعیت‌های جغرافیایی مختلف را که به صورت توزیع شده عمل می‌کنند را داراست، همچنین توانایی تحلیل پروژه‌هایی با بیش از صد میلیون خط کد را براحتی دارد.

۴ یکپارچگی چرخه‌ی تولید نرم‌افزار[۲]

برنامه‌ی Coverity توانایی همگام‌سازی با بسیاری از سیستم‌ها و ابزارهایی که در طول فرآیند تولید نرم‌افزار استفاده می‌شوند را دارد:

[۲] Software development life cycle (SDLC) integration

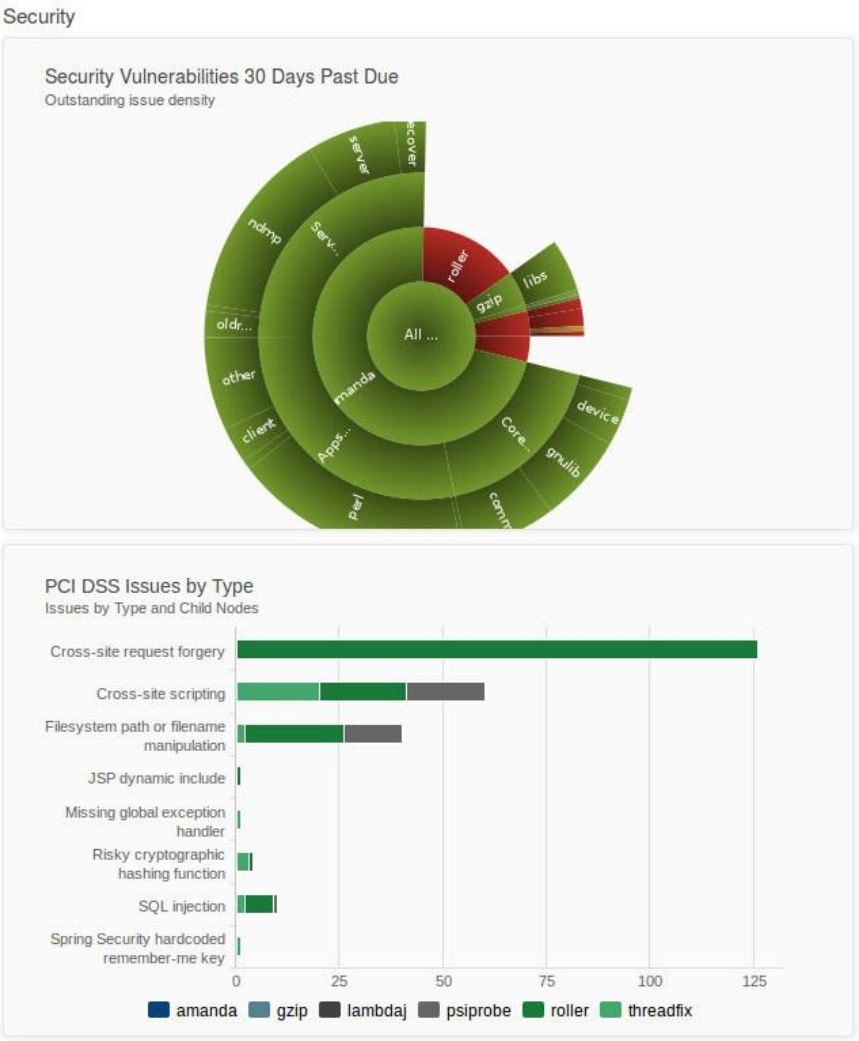
- سیستم‌های کنترل کد پروژه[۳]
- سیستم‌های ساخت و پیوستگی مداوم[۴]
- ردیابی باگ‌های نرم‌افزاری[۵]
- سیستم‌های مدیریت چرخه‌های تولید نرم‌افزار[۶]
- محیط‌های توسعه‌ی یکپارچه[۷]

بستر باز این برنامه امکان افزودن نرم‌افزهای شخص ثالث را برای داشتن تحلیل‌های یکپارچه و مدیریت بهتر خطاها فراهم می‌کند. بدین ترتیب کاربران می‌توانند تمامی خطاها را با یک بازنمایی واحد مشاهده کنند.

۵ مدیریت دقیق و کارآمد ایرادات

- به کمک ویژگی Coverity Connect می‌توان اشکالات مختلف موجود در کد را ردگیری کرد. علاوه بر این راه‌حل دقیق و شفافی نیز برای ایرادات برنامه ارائه می‌شود که اعمال آن‌ها کاملا ساده و به صورت تعاملی می‌باشد. کاربران بدون داشتن تخصص خاصی در حوزه‌ی کدنویسی امن، می‌توانند تغییراتی که توسط Coverity پیشنهاد می‌شود را اعمال و مشکلات برنامه را برطرف سازند.
- Coverity Connect امکان ردگیری ایرادات را با مشخص کردن محل وقوع آن‌ها در کد برنامه فراهم و مسیر دقیقی از محل وقوع ایرادات ترسیم می‌کند.
- این برنامه بسته به نوع مشکل، وظیفه‌ی حل آن را به عهده‌ی یکی از توسعه‌دهندگان موجود در تیم توسعه که برای این کار مناسب‌تر است می‌سپارد. این برنامه توانایی کشف انواع مشکلات امنیتی از قبیل استانداردهای زیر را دارد:
  - ✓ OWASP Top ۱۰
  - ✓ CWE
  - ✓ PCI

[۳] source control management
[۴] continuous integration
[۵] bug tracking
[۶] application life cycle management (ALM)
[۷] integrated development environments (IDEs)



شکل ۲ تحلیل آسیب‌پذیری‌های مختلف در Coverity

۶ توانایی کنترل و کاهش ریسک

با استفاده از قابلیتی که Coverity Policy Manager فراهم می‌کند می‌توانید قواعد جدیدی را برای ارزیابی کیفیت و امنیت پروژه تعریف نمایید. می‌توانید عناصر خاصی را برای بازبینی انتخاب کنید و سپس از میان معیارهای موجود، تعدادی را انتخاب و معیارهای جدیدی را تعریف نمایید. با استفاده از این ویژگی امکان مدیریت و کاهش ریسک ناشی از خطاهای پیچیده نیز وجود دارد.

۷ انطباق با استانداردهای مختلف کشف آسیب‌پذیری

امکان استفاده آسان از کیت توسعه‌ی نرم‌افزار[۸] که توسط قابلیت Coverity Extend فراهم می‌شود باعث می‌گردد تا خطاهای خاص را نیز بتوان کشف کرد. با استفاده از این کیت می‌توان، تحلیل‌کننده‌های جدیدی را تعریف کرد که خطاهای مورد نظر ما را بیابند. برای این کار از یک زبان برنامه‌نویسی خاص‌دامنه[۹] استفاده می‌شودکه به کمک آن می‌توان تحلیل‌گرهای جدیدی را ساخت.

۸ زبان‌ها و چارچوب‌های پشتیبانی شده

همانطور که در جدول ۱ مشخص است، Coverity از طیف وسیعی از زبان‌های برنامه‌نویسی با کاربردهای مختلف پشتیبانی می‌کند. این زبان‌ها در حوزه‌های گوناگون مانند برنامه‌های تحت وب، موبایل، نرم‌افزارهای کامپیوتر شخصی و ... کاربرد دارند. بنابراین از Coverity می‌توان در بسیاری از پروژه‌ها بهره برد.

جدول ۱ زبان‌های پشتیبانی شده توسط Coverity

نام زبان برنامه‌نویسی	توضیحات
C/C++	یک زبان برنامه‌نویسی رایانه‌ای همه‌منظوره،همگردان، سطح میانی، شیءگرا و چندرگه (که از برنامه‌نویسی رویه‌ای، تجرید داده‌ها و برنامه‌نویسی شیءگرا پشتیبانی می‌کند)، عمومی و با قابلیت‌های سطح بالا و سطح پایین می‌باشد. این زبان دارای قابلیت‌های انواع داده ایستا، نوشتار آزاد، چندمدلی، معمولاً زبان ترجمه شده با پشتیبانی از برنامه‌نویسی ساخت‌یافته، برنامه‌نویسی شیءگرا، برنامه‌نویسی جنریک است.
C#	یک زبان برنامه نویسی همگردان، سطح بالا، شیءگرا، ساخت یافته، رویداد محور، تابعی، دستوری و جنریک است که توسط شرکت مایکروسافت در سال ۲۰۰۰ میلادی از خانواده‌ی زبان‌های چارچوب دات‌نت معرفی شد. زبان سی شارپ همچنین از خانواده زبان های برنامه نویسی سی نیز است.
Java	زبان جاوا شبیه به *C++* است اما مدل شیءگرایی آسان‌تری دارد و از قابلیت‌های سطح پایین کمتری پشتیبانی می‌کند. ایده شیء گرایی جاوا از زبان اسمال‌تاک گرفته شده است.
JavaScript	جاوااسکریپت، به اختصار *JS* زبان برنامه‌نویسی سطح بالا، پویا، مبتنی بر شی، وابستگی کم به نوع (*Weakly typed*)، چند رویه و تفسیری است. در کنار *HTML* و *CSS*، جاوااسکریپت یکی از سه هسته صفحات دنیای وب می‌باشد.
PHP	یک زبان برنامه‌نویسیِ شیءگرا است که برای طراحی وب توسعه یافته‌است، اما می‌توان از آن به عنوان یک زبان عمومی نیز استفاده‌کرد.

[۸] Software Development Kit
[۹] domain-specific functional programming language

Python	یک زبان برنامه‌نویسی همه منظوره، همگردان/مُفَسّر، سطح بالا، شیءگرا، اِسکریپتی و متن باز است
ASP.NET	یک چارچوب کاربردی *Web* است که توسط شرکت مایکروسافت عرضه گردیده تا برنامه نویسان بتوانند برای ساخت سایت‌های *Web* و برنامه‌های *Web* پویا وسرویس‌های *Web* پویا وسرویس‌های *Web XML* از آن استفاده کنند.
Objective-C	یک زبان شی‌گرا است که با اضافه کردن مفاهیم ارسال پیام از زبان اسمال‌تاک به زبان سی ایجاد شده. در حال حاضر استفادهٔ اصلی آن در محیط‌های *Mac OS X* و *iPhone OS* است.
JSP	صفحات جاواسرور(جی‌اس‌پی)، یک فناوری از سکوی جاواست که به توسعه‌دهندگان نرم‌افزارِ سرورها کمک می‌نماید تا صفحات پویا مبتنی بر اچ‌تی‌ام‌ال و ایکس‌ام‌ال یا اسناد دیگری را ایجاد نمایند.
Node.js	جی‌اس امکان استفاده از جاوااسکریپت برای نوشتن اسکریپت‌های سمت سرور را فراهم می‌کند تا بدین صورت بتوان با آن صفحات وب پویا را قبل از فرستادن آن به مرورگر کاربر تولید کرد.
Ruby	یک زبان برنامه‌نویسی انعطاف‌پذیر، پویا و شیءگرا است. روبی ویژگی‌های نگارشی پرل و شی گرایی اسمال‌تاک را با هم در خود دارد.
Android	یک سیستم‌عامل همراه است که گوگل برای تلفن‌های همراه و تبلت‌ها عرضه می‌کند و با همکاری ده‌ها شرکت بر روی دستگاه‌های مبتنی بر اندروید قرار می‌دهد. اندروید بر پایه‌ی هسته لینوکس ساخته شده‌است و در بین سیستم‌عامل‌های همراه بیشترین استفاده را دارد.
Swift	سویفت یک زبان برنامه‌نویسی چند شیوه‌ای و از نوع کامپایلری است که برای توسعه‌ی *iOS*، *macOS*، *watchOS* و *tvOS* توسط شرکت اپل ساخته‌شده است.
Fortran	زبان برنامه‌نویسی مفسری است (ایستای کامپایل شده). زبان برنامه‌نویسی فورترن زبانی ساده و محاسباتی است و پروژه‌های بسیاری از رشته‌های فنی مهندسی به کمک این زبان نوشته و اجرا شده‌است.
Scala	یک زبان برنامه‌نویسی شیءگرا و تابعی است. اسکالا تلفیق زبان‌های شی‌گرا همچون روبی و جاوا با زبان‌های تابعی همچون *Haskell* و *Erlang* است.
VB.NET	ویژوال بیسیک دات نت یکی از زبان‌های معرفی شده به منظور نوشتن برنامه‌های مبتنی و با استفاده از چهارچوب دات نت است. دستورهای این زبان مشابه بیسیک ,*Q basic* است.
iOS	یک سیستم عامل همراه ساخته شرکت اپل است که در ابتدا برای آیفون و آی‌پاد تاچ توسعه داده می‌شد، از آن زمان به بعد برای استفاده در سایر دستگاه‌های شرکت اَپل مانند آی‌پد و اَپل تی‌وی گسترش یافت. شرکت اَپل مجوز استفاده از آی اُ اِس برای نصب بر روی سخت‌افزارهای شخص ثالث را نمی‌دهد.

۹ پلتفرم‌های پشتیبانی شده

ابزار Coverity تقریبا از انواع مختلف سیستم‌های عامل، پشتیبانی می‌کند. که انعطاف‌پذیری استفاده از این ابزار را افزایش می‌دهد. در جدول ۲، سیستم‌عامل‌هایی که توسط Coverity پشتیبانی می‌شود، همراه با توضیح مختصری آمده است:

جدول ۲ سیستم‌عامل‌های پشتیبانی شده توسط Coverity

نام	نشان	توضیحات
Windows	Windows	خانواده‌ای از سیستم‌عامل‌هایی است که شرکت مایکروسافت آن را برای رایانه‌های شخصی (*PC*)، تلفن‌های هوشمند و رایانه‌های لوحی تولید کرده‌است. این سیستم‌عامل، نسخه‌های متعددی دارد که از سال ۱۹۸۵ تاکنون به بازار عرضه شده‌اند.
AIX	AIX	نام دسته‌ای از سیستم‌عامل‌های یونیکسی انحصاری است که توسط شرکت آی‌بی‌ام وبرای چندین معماری سخت‌افزاری مختلف توسعه‌داده شده و به فروش می‌رسند.
Linux		یک سیستم‌عامل شبه یونیکس است که بخش عمده‌ی آن سازگار با استاندارد پازیکس است. لینوکس از سخت‌افزارهای مختلفی پشتیبانی می‌کند از جمله انواع مختلف تلفن همراه، تبلت، مسیریاب، و کنسول بازی تا رایانه‌های رومیزی، رایانه‌های بزرگ و ابررایانه‌ها.
HP-UX	hp UX	یک سیستم‌عامل انحصاری است که توسط شرکت هیولت-پاکارد توسعه می‌یابد. این سیستم‌عامل مبتنی بر سیستم پنج یونیکس است و از جمله چهار سیستم‌عامل انحصاری است که از اوپن‌گروپ گواهی سازگاری با استاندارد *UNIX* ۰۳ دریافت کرده است و قانونا می‌تواند یک «یونیکس» نامیده شود.
Mac OS X	macOS	این سیستم‌عامل به صورت انحصاری بر روی کامپیوترهای مکینتاش و تمام مک‌هایی که از سال ۲۰۰۲ عرضه شده‌اند اجرا می‌شود. مک او اس بعد از مایکروسافت ویندوز، دومین سیستم عامل مورد استفاده در جهان در زمینه دسکتاپ (رایانه شخصی) است.

NetBSD		نت‌بی‌اس‌دی یک سیستم عامل شبه یونیکس و آزاد است که از بی‌اس‌دی یونیکس مشتق شده است. در بین سیستم‌عامل‌های خانواده بی‌اس‌دی، نت‌بی‌اس‌دی قدیمی‌ترین محسوب می‌شود و هنوز هم به طور فعالانه‌ای در حال توسعه است.
Solaris		نام گونه‌ای از سیستم‌عامل یونیکس است که در ابتدا توسط سان مایکروسیستمز تولید می‌شد. این سیستم در سال ۱۹۹۳ جایگزین سیستم قدیمی‌تر شرکت سان که سان‌اواس نام داشت شد.
FreeBSD		یک سیستم‌عامل همه‌منظوره و شبه یونیکس است که پروژه‌ی *FreeBSD* آن را توسعه می‌دهد. *FreeBSD* یکی از نخستین سیستم‌عامل‌های متن‌باز است و یکراست از بی‌اس‌دی یونیکس مشتق شده است.

۱۰ سیستم‌های مدیریت کد پشتیبانی شده

استفاده از سیستم‌های مدیریت نسخه یا کد برنامه، در پروژه‌های کوچک و بزرگ نرم‌افزاری اجتناب‌ناپذیر است، بنابراین Coverity از سیستم‌های معروف مدیریت نسخه پشتیبانی می‌کند و توانایی همگام‌سازی با هریک از آن‌ها را فراهم می‌نماید، سیستم‌هایی که توسط Coverity پشتیبانی می‌شود در جدول زیر آمده است:

جدول ۳ سیستم‌های مدیریت نسخه و کد پشتیبانی شده توسط Coverity

نام	نشان
Accurev	
Apache Subversion (SVN)	
Git	
Mercurial (Hg)	

	*Perforce Helix*
	*Team Foundation Server SCM*

۱۱ کامپایلرهای پشتیبانی شده

Coverity علاوه بر پشتیبانی از زبان‌های مختلف برنامه‌نویسی، از کامپایلرها و مفسرهای گوناگون آن زبان‌ها نیز پشتیبانی می‌کند. کامپایلرهایی که توسط این ابزار پشتیبانی می‌شوند به قرار زیر است:

- ARM C/C++
- Borland C++
- CEVA-XC۴۵۰۰
- Clang
- Cosmic C
- Freescale CodeWarrior
- GNU GCC/G++
- Green Hills C/C++/EC++
- HI-TECH PICC
- HP aCC
- IAR C/C++
- IBM AIX
- IBM XLC
- Intel C++
- JDK for Mac OS X
- Keil compilers
- Marvell MSA
- QNX C/C++
- Renesas C/C++
- SNC C/C++
- SNC GNU C/C++
- Sony ORBIS SDK
- Sony PS۴
- STMicroelectronics GNU C/C++
- STMicroelectronics ST Micro C/C++
- Sun (Oracle) CC
- Sun/Oracle JDK
- Synopsys MetaWare C and C++
- TASKING for ARM Cortex
- TI Code Composer
- Visual Studio
- VisualDSP++
- Wind River C/C++
- OpenJDK
- MPLAB XC۸

۱۲ آسیب‌پذیری‌هایی که مورد بررسی قرار می‌گیرند

آسیب‌پذیری‌هایی که توسط Coverity مورد بررسی قرار می‌گیرد در ذیل آمده است، بررسی این موارد به توسعه‌ی پروژه‌های امن و باکیفیت کمک می‌نماید.

- خطاهای استفاده از رابط برنامه‌نویسی کاربردی[۱۰]
- خطاهای قواعد قراردادی برنامه‌نویسی[۱۱]
- مشکلات سیستم ساخت[۱۲]
- سرریزی بافر[۱۳]
- نقص سلسله‌مراتبی کلاس‌ها[۱۴]
- مشکلات نگه‌داری کد[۱۵]
- خطر دسترسی همزمان به داده‌ها[۱۶]
- خطای کنترل جریان[۱۷]
- تزریق اسکریپت از طریق وبگاه[۱۸]
- جعل درخواست‌های میان‌وبگاهی[۱۹]
- بن‌بست[۲۰]
- مشکلات مدیریت خطا[۲۱]
- عبارت‌های نادرست[۲۲]
- مشکلات مدیریت اعداد صحیح[۲۳]
- دستکاری مسیرها[۲۴]
- ناکارآمدی عملکرد[۲۵]
- هنگ کردن برنامه[۲۶]
- شرایط رقابتی[۲۷]
- فقدان منابع[۲۸]
- نقض قواعد[۲۹]
- نقض قواعد استاندارد قراردادی[۳۰]
- تنظیمات امنیتی نادرست[۳۱]
- تزریق به پایگاه داده[۳۲]
- اعضای مقداردهی نشده[۳۳]
- سرریزی اعداد صحیح[۳۴]
- دستکاری ناامن اطلاعات[۳۵]
- خرابی حافظه[۳۶]
- دسترسی غیرمجاز به حافظه[۳۷]
- اشاره‌گر تهی[۳۸]
- شواهد کدنویسی غیرانعطاف‌پذیر[۳۹]

---

[۱۰] API usage errors
[۱۱] Best practice coding errors
[۱۲] Build system issues
[۱۳] Buffer overflows
[۱۴] Class hierarchy inconsistencies
[۱۵] Code maintainability issues
[۱۶] Concurrent data access violations
[۱۷] Control flow issues
[۱۸] Cross-site scripting (XSS)
[۱۹] Cross-site request forgery (CSRF)
[۲۰] Deadlock
[۲۱] Error handling issues
[۲۲] Incorrect expression
[۲۳] Integer handling issues

۱۳ مراجع

[۱] https://www.synopsys.com/software-integrity/resources/datasheets/coverity.html

[۲] https://www.synopsys.com/software-integrity/security-testing/static-analysis-sast.html

[۳]https://marketplace.visualstudio.com/items?itemName=CoverityBySynopsys.CoveritySoftwareTesting

[۴] https://continuousassurance.org/category/static-analysis-tools/

[۵] https://www.owasp.org/index.php/Source_Code_Analysis_Tools

---

[۲۴] Path manipulation
[۲۵] Performance inefficiencies
[۲۶] Program hang
[۲۷] Race conditions
[۲۸] Resource leaks
[۲۹] Rule violations
[۳۰] Security best practices violations
[۳۱] Security misconfigurations
[۳۲] SQL injection
[۳۳] Uninitialized members
[۳۴] Integer overflows
[۳۵] Insecure data handling
[۳۶] Memory—corruptions
[۳۷] Memory—illegal accesses
[۳۸] Null pointer dereferences
[۳۹] Hard-coded credentials