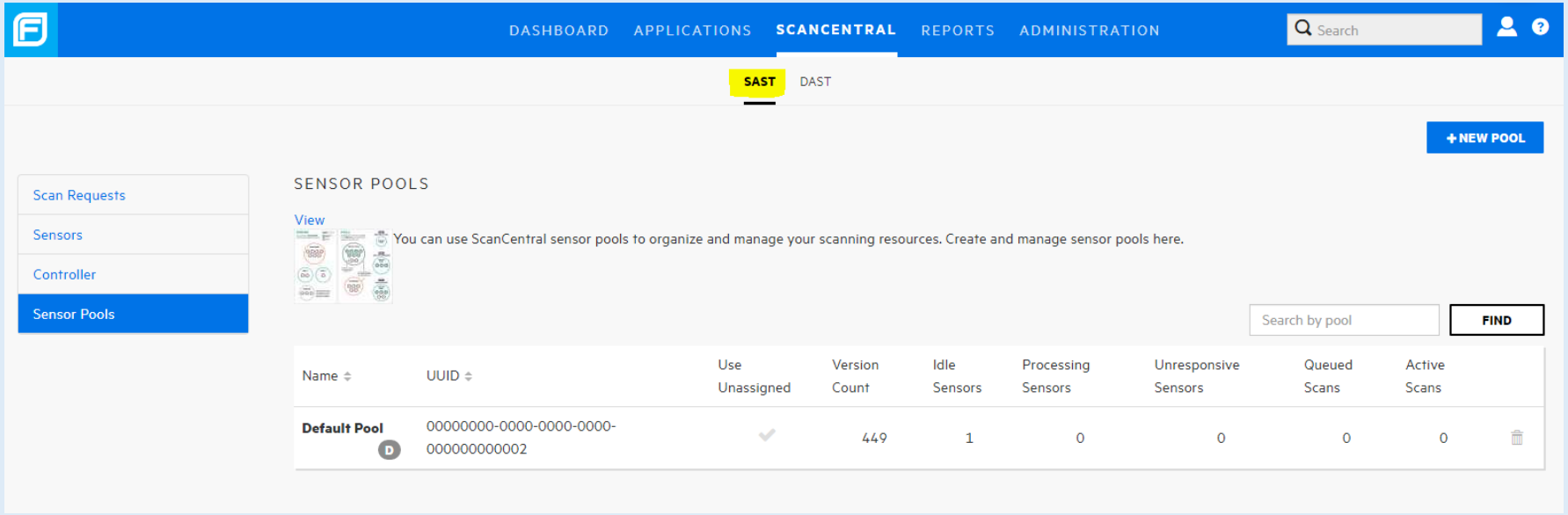
Introduction to ScanCentral SAST configuration and scan analysis

By Dobromir Todorov

- ScanCentral SAST is an automated security tool which utilizes Static Code Analyzer functionalities. In this matter, SCA pinpoints the root cause of security vulnerabilities in the source code, prioritizes the most serious issues, and provides detailed guidance on how to fix them so developers can resolve issues in less time with centralized software security management.

- Static analysis automation with SC SAST can extend the SCA (Static Code Analyzer) capabilities to next level by performing multiple vulnerability scans at once.



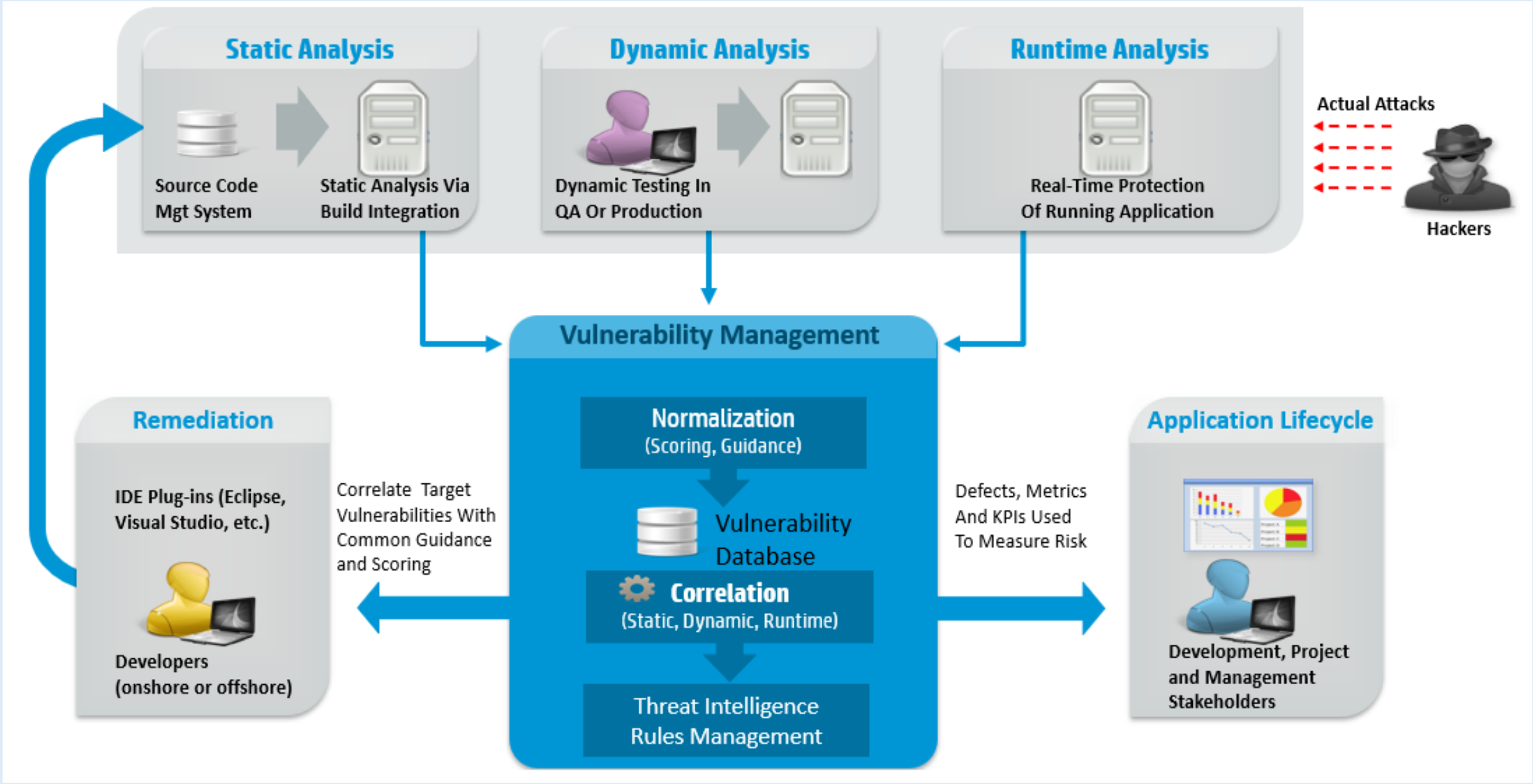
- Static Testing Helps Build Better Code: Static Application Security Testing (SAST) identifies security vulnerabilities during early stages of development when they are least expensive to fix.
- Find Security Issues Early: To process code, Fortify SCA works much like a compiler—which reads source code files and converts them to an intermediate structure enhanced for security analysis. This intermediate format is used to locate security vulnerabilities. The analysis engine, which consists of multiple specialized analyzers, uses secure coding rules to analyze the code base for violations of secure coding practices.
- Manage Results with Fortify Software Security Center (SSC):Fortify Software Security Center (SSC) is a centralized management repository providing visibility to an organization's entire application security program to help resolve security vulnerabilities across the software portfolio. Users can review, audit, prioritize, and manage remediation efforts, track software security testing activities, and measure improvements via the management dashboard and reports to optimize static and dynamic application security test results. Fortify SSC helps to provide an accurate picture and scope of the application security posture across the enterprise. The Fortify SSC server resides in a central location and receives results from different application security testing activities, such as static, dynamic, and real-time analysis.
- Integration Ecosystem Includes:

Integrated Development Environments (IDE): Eclipse, Visual Studio, JetBrains (including IntelliJ)

CI/CD Tools: Jenkins, Bamboo, Visual Studio, Gradle, Make, Azure DevOps, GitHub, GitLab, Maven, MSBuild

Issue Trackers: Bugzilla, Jira, ALM Octane

Open Source Security Management: Sonatype, Snyk, WhiteSource, BlackDuck


Static Analysis
Source Code Mgt System
Static Analysis Via Build Integration
Dynamic Analysis
Dynamic Testing In QA Or Production
Runtime Analysis
Real-Time Protection Of Running Application
Actual Attacks
Hackers
Vulnerability Management
Normalization (Scoring, Guidance)
Vulnerability Database
Correlation (Static, Dynamic, Runtime)
Threat Intelligence Rules Management
Remediation
IDE Plug-ins (Eclipse, Visual Studio, etc.)
Developers (onshore or offshore)
Correlate Target Vulnerabilities With Common Guidance and Scoring
Defects, Metrics And KPIs Used To Measure Risk
Application Lifecycle
Development, Project and Management Stakeholders

The Static Suite

SCA&ScanCentral SAST

- SCA&SC SAST run against applications in development.

Audit Workbench (AWB)

- Visual interface for analysis of software vulnerabilities.

Secure Coding Plugins

- Integrated vulnerability detection into Integrated Development Environments (IDEs).

SSC Server

- Management for multiple audit projects from a single centralized console.

Fortify ScanCentral SAST 21.2 Requirements

- Fortify ScanCentral SAST Application Server/Controller

Micro Focus Fortify ScanCentral SAST supports Apache Tomcat version 9.x for **Java 11**. Fortify recommendsthat you install the Fortify ScanCentral SAST Controller on a high-end 64-bit processor running at 2 GHz with at least 8 GB of RAM.

Controller Platforms and Architectures

The Fortify ScanCentral SAST Controller supports the platforms and architectures listed in the following table.

Operating System	Versions
Windows	Server 2016 Server 2019
Linux	Red Hat Enterprise Linux 7.x, 8 SUSE Linux Enterprise Server 12, 15

- Fortify ScanCentral SAST Client and Sensor Hardware Requirements

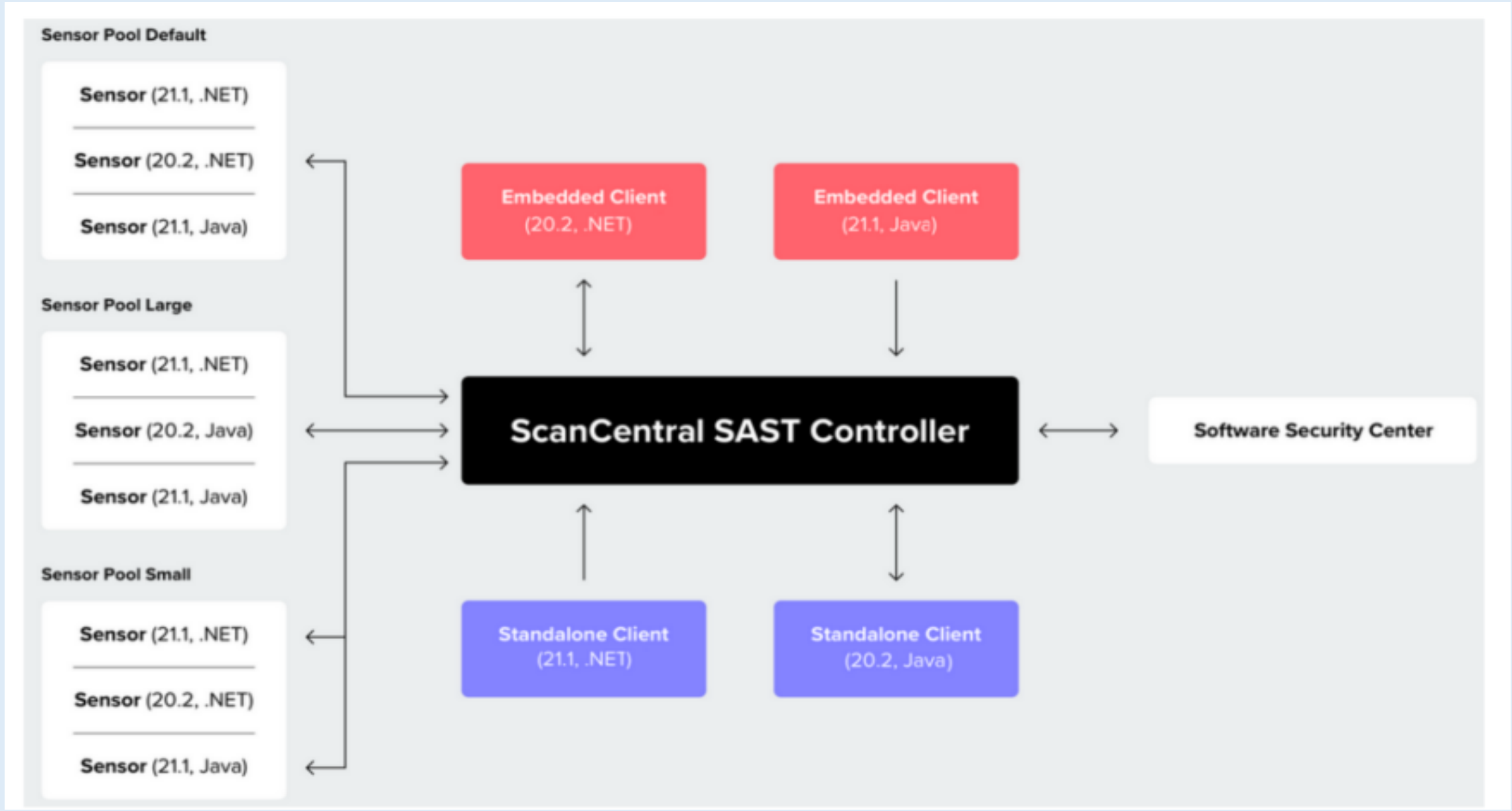
Fortify ScanCentral SAST clients and sensors run on any machine that supports Micro Focus Fortify Static Code Analyzer.

- Supported programming languages/build tools:

https://www.microfocus.com/documentation/fortify-static-code-analyzer-and-tools/2120/Fortify_Sys_Reqs_21.2.0.pdf#page=30

ScanCentral SAST 21.2 install files

- Fortify_ScanCentral_Controller_21.2.3.zip
- Fortify_SCA_and_Apps_21.2.3_Windows.zip
- Fortify_SSC_Server_21.2.3.zip


Sensor Pool Default
Sensor (21.1, .NET)
Sensor (20.2, .NET)
Sensor (21.1, Java)
Sensor Pool Large
Sensor (21.1, .NET)
Sensor (20.2, Java)
Sensor (21.1, Java)
Sensor Pool Small
Sensor (21.1, .NET)
Sensor (20.2, .NET)
Sensor (21.1, Java)
Embedded Client (20.2, .NET)
Embedded Client (21.1, Java)
ScanCentral SAST Controller
Software Security Center
Standalone Client (21.1, .NET)
Standalone Client (20.2, Java)

New functionalities in ScanCentral:

- Start Maintenance mode

After the Controller is placed in maintenance mode, sensors complete the scans they are currently running, but accept no new scans. After the Controller is back up and running, the sensors again become available: https://www.microfocus.com/documentation/fortify-software-security-center/2120/SC_SAST_Guide_21.2.0.pdf#page=40

- Graceful Shutdown and Timer Support

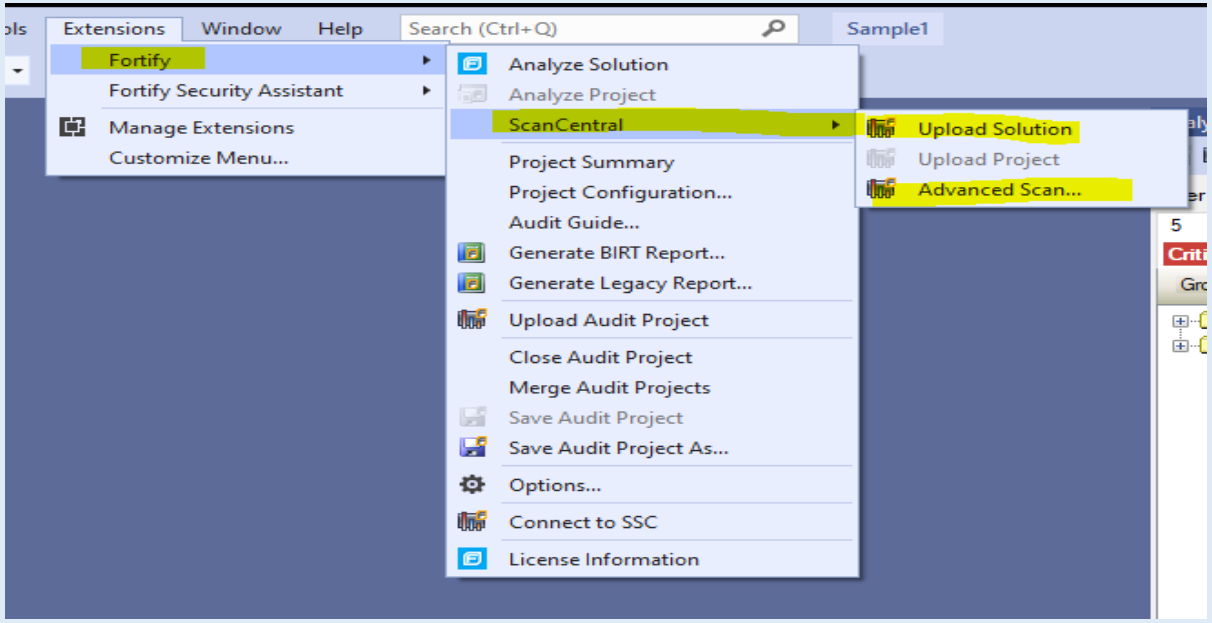
When shutting down Fortify ScanCentral SAST, the controller allows currently running scans to complete while keeping other scans from starting. Once the controller is running again, it will run the scans in the queue. In addition, a timeout can be set for long running scans that will cancel the scan if breached and free the sensor to pick up a new scan request. The -sto (--scan-timeout) option is used to specify the maximum amount of time a scan job can run: https://www.microfocus.com/documentation/fortify-software-security-center/2120/SC_SAST_Guide_21.2.0.pdf#Setting%20the%20Maximum%20Run%20Time%20for%20Scans

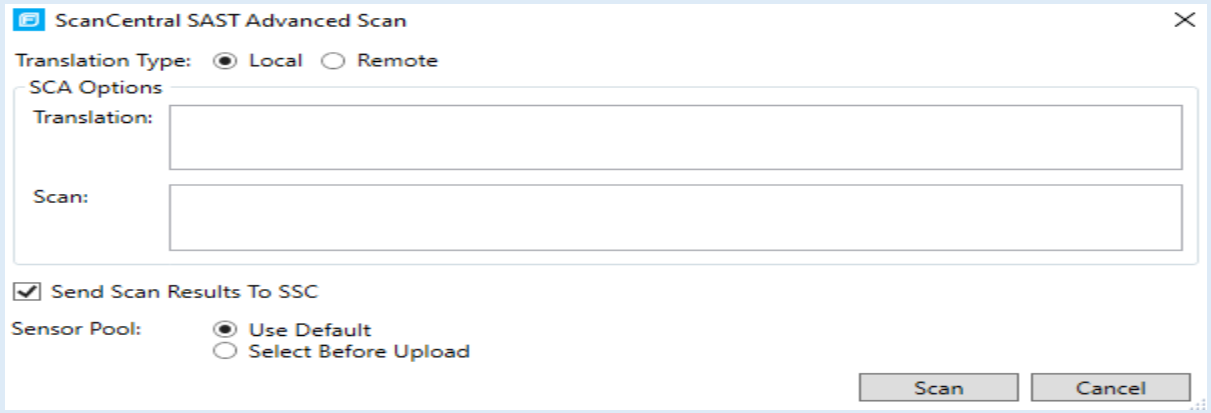
Sensor Pool Assignment Improvement

When starting up a sensor, you can assign it to a specific sensor pool without having to use the Fortify Software Security Center UI. The - pool (--submit-to-pool) option specifies the sensor pool into which a sensor is to be placed at startup.

ScanCentral SAST scan options

- Scan from VS with Fortify extension plugin





ScanCentral SAST scan options

- Local Scan with SC client:

$ sourceanalyzer -b cs-sample -clean

$ sourceanalyzer -b cs-sample msbuild /t:rebuild Sample1.sln

$ sourceanalyzer -b cs-sample -show-files

Local scan without SSC upload - Fortify_ScanCentral_Controller_21.2.3\tomcat\jobFiles folder

$ scancentral.bat –url start -b cs-sample –scan

Local scan with SSC upload

scancentral.bat -sscurl <ssc_url> -ssctoken <ScanCentralCtrlToken> start -upload -versionid 10 -b <mybuildId> -uptoken <ScanCentralCtrlToken> -scan

ScanCentral SAST scan options

- **Remote Scan with SC client:**

Offloading Both Translation and Scanning

scancentral.bat -url start -bt msbuild -bf mySolution.sln

-bf, --build-file <file>

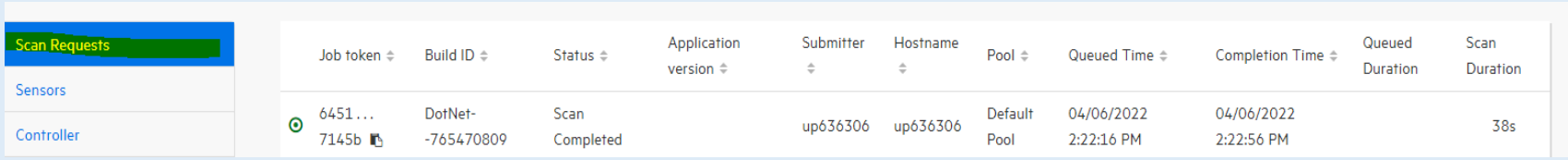
-bt, --build-tool <name>

Example:

Checking for updates...
No update available or auto update is disabled on the controller.
Verifying controller URL...
Initializing client authorization token...
Controller at http://<ctrHostname>:8082/scancentral-ctrl is UP
No email address detected. No status emails will be sent for this job.
Setting up SCA version...
Retrieving SCA version...
Gathering project information...
ScanCentral integration mode ("C:\Users\\AppData\Local\Temp\sc_msbuild_args4771552954934297433"). Writing down the SCA translation command:
scancentral_integration @"C:\Users\\AppData\Local\Temp\sc_msbuild_args4771552954934297433\Sample1.rsp"

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:03.82
Packaging project...
Compressing job files...
Uploading job...
Submitted job and received token: 6451585b-d521-4324-ba17-011dde97145b



ScanCentral SAST scan options

- Remote Scan with SC client:

Offloading Both Translation and Scanning

Step1: Alternatively, you can save the project package [create a package] locally, as follows:

scancentral package -o <path to package> --build-tool msbuild --build-file <solution file>

//

scancentral.bat package -bt msbuild -bf mySolution.sln -o myPackage.zip

Step 2: To send the package to ScanCentral SAST, run:

scancentral -url <controller_url> start -package <package path>

Fortify SAST documentation:

- SAST guide: https://www.microfocus.com/documentation/fortify-software-security-center/2120/SC_SAST_Guide_21.2.0.pdf
- Fortify system requirements: https://www.microfocus.com/documentation/fortify-static-code-analyzer-and-tools/2120/Fortify_Sys_Reqs_21.2.0.pdf
- VS SCA extension plugin: https://www.microfocus.com/documentation/fortify-static-code-analyzer-and-tools/2120/VS_Ext_Guide_21.2.0.pdf
- SCA guide: https://www.microfocus.com/documentation/fortify-static-code-analyzer-and-tools/2120/SCA_Guide_21.2.0.pdf