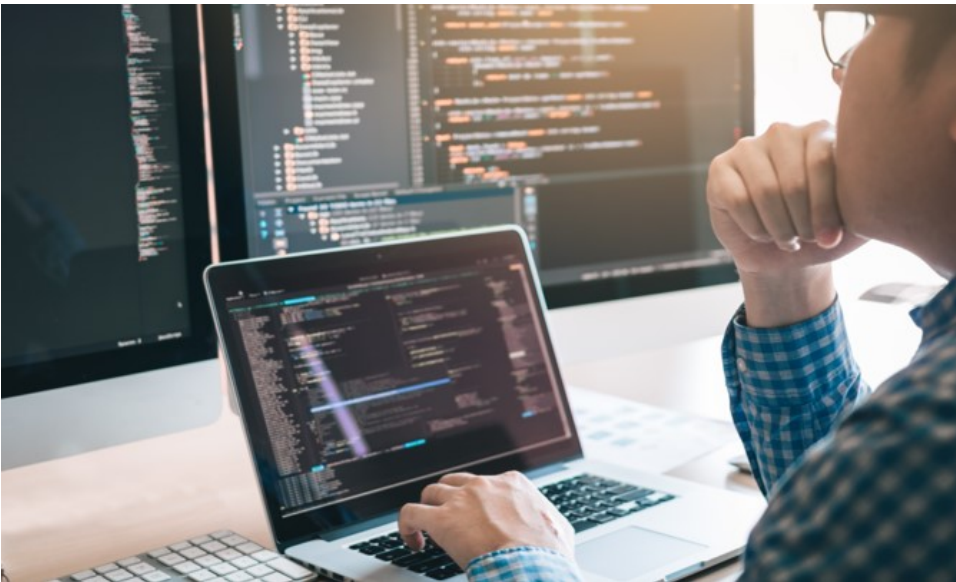


Software Security Assessment

BlackBerry QNX Professional Services



The security of your embedded system is only as good as its least secure hardware and software components. This is why it's important to fully understand security across the supply chain and entire lifecycle.

Backed by 30 years of experience in cybersecurity and a proven binary code scanning solution, BlackBerry® has developed a set of security services including an OSS Assessment, a Software Security Assessment and a Penetration Testing service package to identify potential threats that may come with using certain open source or third-party software in your embedded system.

BlackBerry has deep expertise and decades of security research and development to help you protect your products. Traditional security consultants test to find holes, and then go home just as the real work begins. BlackBerry will support your organization from product design to ongoing incident response. Our security engineers can advise on secure architecture design, development, deployment, and supply chain management whether you are looking to build a secure platform, harden a product, or deploy a secure and effective IoT capability.

You can also rely on the embedded system expertise of BlackBerry QNX services team. QNX® has been the operating system of choice for mission-critical embedded systems for the last 40 years and we have helped thousands of customers design safe, secure and reliable systems.

Software Security Assessment

BlackBerry will assess the security of your embedded software through an in-depth static analysis of software deliverables. Leveraging BlackBerry's static binary application testing platform, BlackBerry Jarvis™, we will assess the security risks of your embedded system software, including your open source software (OSS) and any third party or API usage.

The specific techniques employed will be tailored to the system being assessed, the strategy being to understand weaknesses in the software's security design by inspection of the binary, support and operating system files. BlackBerry will provide a walkthrough of detailed dashboards showing statistics on software craftsmanship, code quality and vulnerability.

BlackBerry Jarvis supports a wide range of binaries such as Linux, QNX, VxWorks, Integrity, Renesas and other vendor specific varieties. A deliverable of the service is the Open Source Software Bill of Materials including:

- Library license and version detection
- Public vulnerability (CVE) detection and linkage
- License Compliance

BlackBerry Jarvis

BlackBerry Jarvis is a cloud-based, binary static application security testing (SAST) platform. Through cutting-edge system exploration technology, Jarvis provides powerful capabilities to examine a complete software product for security vulnerabilities and software craftsmanship. Since BlackBerry Jarvis extracts the characteristics and attributes from compiled binaries, access to source code isn't required.

Package Details

This security assessment will include:

- Automated composition analysis of binary images, support files
- Static firmware analysis with a focus on application vulnerabilities, as well as logic flaws
- OSS bill of materials
- Review of the operating system (QNX, Android and Linux varieties)
- Presentation of the findings and mitigation recommendations in a two-hour workshop

About BlackBerry® QNX®

BlackBerry QNX is a leading supplier of safe, secure, and trusted operating systems, middleware, development tools, and engineering services for mission-critical embedded systems. BlackBerry QNX helps customers develop and deliver complex and connected next generation systems on time. Their technology is trusted in over 150 million vehicles and more than 300 million embedded systems in medical, industrial automation, energy, and defense and aerospace markets. Founded in 1980, BlackBerry QNX is headquartered in Ottawa, Canada and was acquired by BlackBerry in 2010

For more information, visit blackberry.qnx.com and follow [@QNX_News](https://twitter.com/QNX_News).

