

Powering clients to a future shaped by growth

A Frost & Sullivan White Paper

BlackBerry® Jarvis® Secures Embedded Systems by Uncovering Hidden Vulnerabilities and Exposures



Contents

- 3** The Role of Embedded Devices in the 21st Century
- 4** Recent Notable Attacks
- 5** Challenges in Securing Embedded Systems
- 6** Growing Interest from Regulators and Governments
- 7** BlackBerry Jarvis 2.0
- 10** A Watershed Moment in Embedded Security
- 11** Conclusion


The Role of Embedded Devices in the 21st Century

The digital revolution in the 20th century transformed how societies, economies, and businesses function. Individuals and businesses today are dependent on simple gadgets such as Wi-Fi routers and more complex ones such as magnetic resonance imaging machines. All these devices depend on embedded circuits and software within them. While Internet of Things (IoT) devices such as smart thermostats and connected cars are becoming ubiquitous, the IoT is also laying the groundwork for futuristic concepts including smart cities, autonomous driving, and remote health monitoring.

However, the number of embedded smart sensors and devices joining enterprise or home networks has also contributed to an expanded attack surface, leaving users vulnerable to cyberattacks. Numerous recent security breaches have proved that connected devices have insufficient or ineffective security controls. Considering that they are responsible for mission-critical applications such as patient health and occupant safety, their security is emerging as a top priority. A disproportionately large number of vulnerabilities affecting connected devices stem from the embedded software within them.

Although IoT devices form only a subsegment of embedded devices, they are experiencing phenomenal growth and have also been the root of multiple security breaches. In 2020, 26.4 billion devices were in service, a number that will reach approximately 66 billion by 2026, according to Frost & Sullivan. A recent study by Ponemon Institute indicates that 9 out of 10 companies adopting IoT expect to experience a breach or attack caused by unsecured IoT devices in the next 2 years. Such expectations are not unfounded, as the first half of 2021 saw a staggering 1.5 billion attacks on IoT devices, up from 639 million in the first half of 2020. Alarming, cyberattacks are not only growing in number but also in intensity. The average cost of a data breach is at an all-time high of \$4.2 million USD globally, with a breach in the United States costing upward of \$9.05 million USD on average.

Undeniably, IoT and embedded devices are in the crosshairs of cyberattackers and will remain so. While organizations are becoming wary of integrating IoT devices into their networks, regulators are waking up to the need for stricter controls on connected devices and the supply chain behind them. Considering the risks that compromised IoT devices can pose to mission-critical applications, original equipment manufacturers (OEMs) and device manufacturers have realized the need to improve visibility in their supply chains and the vulnerabilities hidden within them.



“ Nine out of 10 companies adopting IoT expect to experience a breach or attack caused by unsecured IoT devices in the next 2 years. ”

Recent Notable Attacks

Throughout the years, several serious hacks and compromises have targeted embedded devices, the severity of which has been growing as the number of IoT devices grows, along with their involvement in mission-critical applications. As the examples below state, compromised embedded devices are existential threats to businesses and can also result in fatalities.

In 2017, St. Jude Medical (Abbott) discovered that its pacemakers contained a vulnerability that would allow unauthorized individuals to access them through radio frequency transmissions, thereby allowing them to issue unauthorized commands and modify device settings. After discovery, hospitals called in 465,000 patients for firmware updates because over-the-air (OTA) updates were not possible.

In another instance, a researcher from KU Leuven University in Belgium discovered a combination of security vulnerabilities in Tesla Model X cars and key fobs that allowed him to reverse engineer the key for the car, subsequently unlocking, starting, and even driving it. Specifically, the researcher found that the OTA update mechanism for the key fob was inadequate and allowed him to take control of it to unlock the car. Tesla later worked on a patch and included it as an OTA update to secure its vehicles.

While connected devices have been the target of attackers, they have also, at times, been a gateway to larger attacks. In 2017, a casino based in North America faced a large-scale data breach in which the attackers stole more than 10 gigabytes of data from its systems. After analysis, investigators found that a connected thermometer in an aquarium in the casino's lobby was the source of the breach.



Challenges in Securing Embedded Systems

When it comes to cybersecurity, the spotlight is on embedded devices—and rightly so, considering their role in numerous attacks. Recognizing the need to secure embedded systems, embedded software developers and users of embedded devices have started exploring ways to harden the security posture of their systems and have discovered unique challenges in the process. These challenges are ingrained in the embedded devices ecosystem and require adjustments to the software and device development process.

In the competitive IoT market, companies are accelerating their product development cycles to launch new products and features before the competition. Ultimately, the competitive pressure is compelling development and testing teams to push final products to market within short timelines. Owing to such short development cycles, open-source and third-party code have become indispensable parts of embedded software development. Such a complex codebase brings a host of vulnerabilities and licensing rules that the developer might not be aware of.

“ Source code analysis fails to provide the most accurate picture of the vulnerabilities within the final binary executable shipped with the product. ”

Embedded systems also feature a complex and globally distributed supply chain in which hardware and software components of the device may come from a wide range of suppliers over whom the OEM might have limited visibility and control. Most OEMs do not even have access to the source code of embedded software, leaving them with little visibility of the software they have inherited from their vendors. Even in cases where an OEM does have access, source code analysis fails to provide the most accurate picture of the vulnerabilities within the final binary executable shipped with the product. The introduction of severe vulnerabilities in the embedded software during the compiling and configuration stage is always possible. In such a situation, OEMs may ship their devices without complete knowledge of vulnerabilities within them.

One of the main contributing factors of the weak security posture of modern embedded systems is the shortage of embedded designers with security expertise. With companies unable to find the developers with the right security skill sets, their security procedures are time-consuming and ineffective, leading to the circulation of vulnerable software without effective oversight.

Adding to the challenge of securing embedded systems is the fact that embedded devices operate in a wide range of environments, and OTA updates can be difficult to push across all devices. While some modern devices do support periodic updates, certain devices still do not facilitate OTA updates and have to be shipped with the most secure version of the software.

Growing Interest from Regulators and Governments

Regulatory bodies and governments globally have recognized the severity of threats posed by embedded devices. While they have already framed and implemented some standards and regulations, stricter and more industry-specific regulations will come into force soon.

The United States has led the charge in implementing a stronger security mechanism for connected devices and has undertaken numerous initiatives in this vein. One of the strongest actions was the Executive Order on Improving the Nation's Cybersecurity that President Biden issued 12 May 2021. It directs numerous government agencies to define strict guidelines to ensure the security of products by design. Under the executive order, contracts for vendors selling to the federal government will include security requirements, and vendors will have to provide a software bill of materials (SBOM) for their products. Essentially, all software and hardware vendors will have to provide proof of security to sell to the federal government.

Following this lead, the US Food and Drug Administration (FDA) has also sought more legislative authority to mandate medical device manufacturers to provide an SBOM as a pre-market submission. The FDA has also sought post-market authority to mandate device manufacturers to support security updates and patches for their devices throughout their life cycle.

Even the automotive industry has recognized the need for stronger and more consistent standards. In June 2020, the United Nations Economic Commission for Europe incorporated the WP.29 cybersecurity regulation, which mandates automakers to manage cybersecurity risks and secure their vehicles by design, respond to incidents across their fleet throughout their life cycle, and receive the type of approval they need to bring a vehicle to the market. The regulation applies to automakers manufacturing or importing vehicles to more than 50 countries, including the United Kingdom, Japan, South Korea, and the entire European Union.

The United States has led the charge in implementing a stronger security mechanism for connected devices and has undertaken numerous initiatives in this vein.



BlackBerry Jarvis 2.0

Assuring Security of Embedded Systems

BlackBerry Jarvis is a software composition analysis tool with static application security testing capabilities that allows stakeholders across the value chain to gain deep visibility into the open-source software and software licenses within embedded systems to detect any vulnerabilities and exposures. Using binary scanning, developers can generate a detailed SBOM, which provides a high-level picture of the entire software supply chain and the vendors involved in shaping the final system. Binary scanning also allows developers to dive deep into the binary executable file and detect the presence of artifacts such as debugging and test tools, license violations or common vulnerabilities and exposures (CVEs), and common weakness enumeration. The scan process also supports an OEM in evaluating the robustness and integrity of its vendors' build process.

BlackBerry Jarvis automates the process of recursively unpacking, extracting, and disassembling binaries, which can take a team of developers days or even months if carried out manually. Manual analysis is almost impossible in automotive infotainment or advanced driver-assistance systems, for example, because they contain hundreds of binary executables and thousands of individual files within them. With BlackBerry Jarvis, developers accelerate and automate the entire process of vulnerability scouting with a high level of accuracy, combined with very low false positives, thus allowing developers to expand their coverage of analysis to ensure a high level of security assurance.



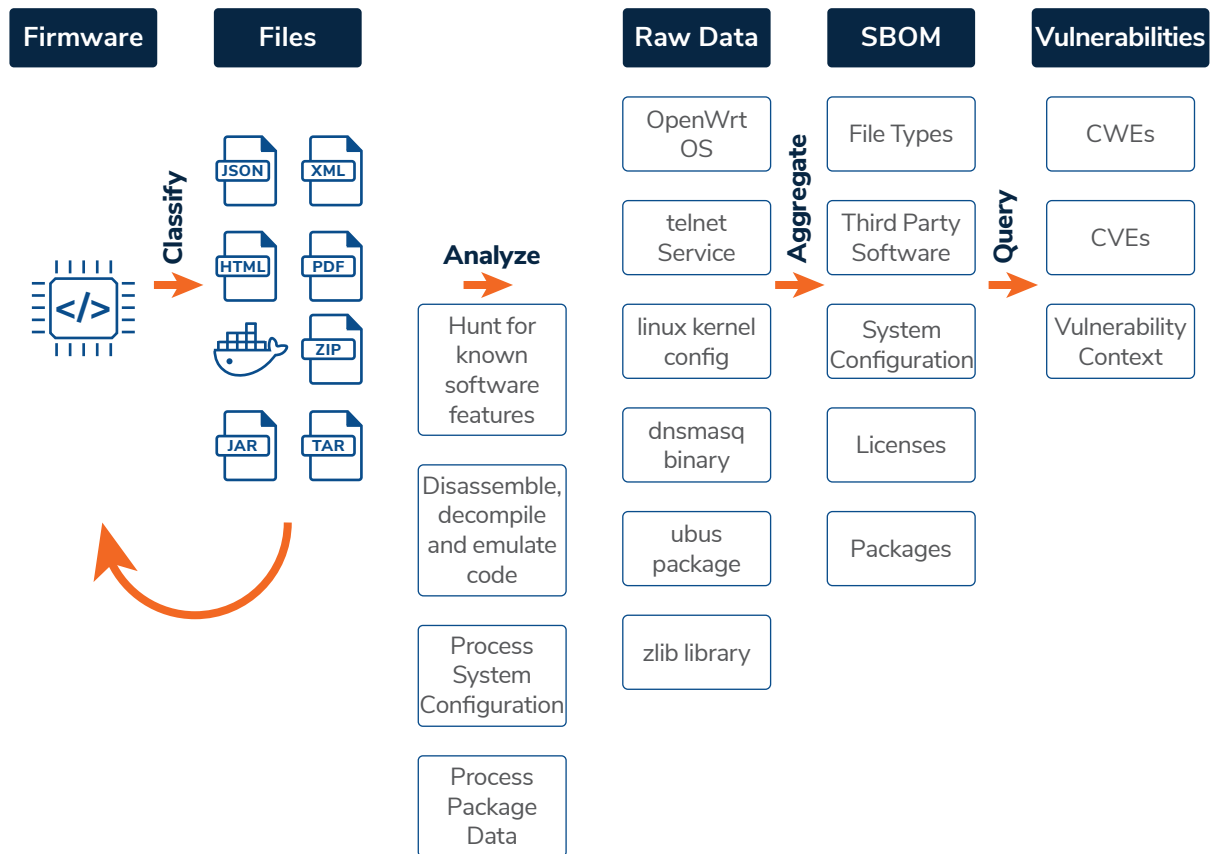
Binary Scanning Process

In principle, the binary scanning process aims to break down a binary executable into its individual components and scan each for vulnerabilities. A binary scanner thus follows the binary creation process in reverse. The binary creation process starts with a developer creating the source code using a programming language or importing libraries that might be open-sourced. For embedded systems, these source codes and libraries usually need to run on an operating system customized for specific hardware central processing unit (CPU) architectures. Compilers and linkers finally process the source code and import the libraries to create a binary executable, which is deployed into the final embedded system and ends up reaching the OEM or the end customer. This binary executable inherits any vulnerabilities and license violations that can be introduced during each step of this process.

With BlackBerry Jarvis, developers can upload the embedded software as either a singular binary executable or an entire system image. BlackBerry Jarvis unpacks and extracts the embedded software and breaks it down into individual files that may come in a wide range of formats. The extraction process uses a recursive manner to break down compressed files and file systems further to extract individual files within them. The files obtained from this extraction are further classified into file formats for additional analysis. At this stage, BlackBerry Jarvis hunts for known software issues and even disassembles or decompiles the code to look for vulnerabilities. The analysis stage produces a lot of raw data that further help BlackBerry Jarvis gain a detailed understanding of the embedded system under analysis. Aggregating the raw data and the findings of the analysis stage, BlackBerry Jarvis generates an SBOM that best describes the content of the binary or system image under analysis. The SBOM tries to capture an overall picture of the full system image and contains the breakdown of file types and third-party software and its version, licenses, packages, and system configuration. BlackBerry Jarvis then compares the third-party software and versions with the CVE database and investigates for any vulnerable piece of code to present the user with a list of vulnerabilities the binary contains.

For each aspect of its analysis, BlackBerry Jarvis presents users with a detailed dashboard, containing statistical findings and the complete dataset from the analysis. For example, the SBOM-oriented dashboards would describe the components of the binary, such as file types, third-party components, and the vulnerabilities identified. In its findings, BlackBerry Jarvis provides the user with a list of CVEs identified, corresponding to each third-party component and its version. Notably, BlackBerry Jarvis leverages the National Vulnerability Database to extract detailed information about each CVE, such as severity, to aid the remediation effort. Regarding remediation, BlackBerry Jarvis also lists the fixed version to help the developer decide which version to update to. In addition, BlackBerry Jarvis provides developers with alternative suggestions such as links to patches, configuration changes, or removal of code. BlackBerry Jarvis also alerts the developer if the vulnerability is non-consequential and does not need fixing by providing appropriate reasoning for such a recommendation. BlackBerry Jarvis provides developers with an option to set up future alerts for scanned binaries. Exercising this option, developers can receive alerts as and when information about a new CVE that might affect them becomes available or if the severity levels of previously disclosed CVEs change.

Exhibit 1: The Binary Scanning Process



A Versatile Tool for Stakeholders across the Value Chain

With a strong set of functionalities aimed at securing embedded systems, BlackBerry Jarvis finds its application at each stage of the device lifecycle, including embedded software development, connected device development, and systems integration. At each stage of development, BlackBerry Jarvis caters to a distinct set of requirements.

- Software integrators need BlackBerry Jarvis to ensure the security of the subsystems and open-source software that will be in the production build of software under development. BlackBerry Jarvis is also a strong tool for suppliers to demonstrate the security posture of their software and embedded systems.
- For security researchers, BlackBerry Jarvis is an investigation tool that they can leverage to get in-depth visibility into embedded systems to uncover vulnerabilities quickly and reliably.
- Software developers can leverage BlackBerry Jarvis to periodically vet the security posture of the open-source libraries that they are importing into their final build.
- Regulators can leverage BlackBerry Jarvis to review and verify security claims of OEMs and their suppliers to enforce uniform implementation of stringent security standards among all stakeholders.

A Watershed Moment in Embedded Security

End users' demand for a more advanced set of functionalities is contributing to the growing complexity of software supply chains. Supply chains involving many downstream vendors and open-source libraries will inevitably bring undisclosed vulnerabilities that will be difficult to patch or resolve after deployment. As OEMs and device manufacturers grapple with the realities of devastating cyberattacks and costly product recalls, they will hold downstream suppliers responsible for their role in security incidents. The security certification process, which is a value addition today, will become a necessity in the near future. Regulators, governments, OEMs, and customers will demand proof of security as part of their buying or certification procedures. Mandatory disclosure of SBOM is a welcome first step toward this and will play an important role in bringing accountability and transparency into the process of embedded software development.

Embedded software development procedures will have to undergo a generational shift with security as a prerequisite rather than an afterthought. Relations between software suppliers, device manufacturers, and OEMs will need to be more open and collaborative to ensure that security is the priority in the software development life cycle from the planning phase with stringent checks implanted at every stage. Automation in security testing will be crucial as product development timelines shrink and talent shortages become more pronounced. Security automation will also be important in the reduction of lead time between the discovery of a vulnerability and the issuance of patches.

Developments in government regulation and industry standards will determine the direction that embedded security will take. While we already saw concrete steps taken by the US government and the automotive industry in Europe, other locations and industries are still to follow in their footsteps. Amid the tightening security norms, companies across the value chain will need to take immediate and decisive action to prioritize the security of their software.

“Relations between software suppliers, device manufacturers, and OEMs will need to be more open and collaborative to ensure that security is the priority in the software development life cycle from the planning phase with stringent checks implanted at every stage.”



Conclusion

The embedded software development landscape is changing rapidly, and security is becoming a building block of tomorrow's connected devices. Strict policing of open-source software combined with the prompt discovery and resolution of vulnerabilities is emerging as a vital component of a successful embedded software development and management strategy.

BlackBerry Jarvis is a unique match to the needs of the embedded software industry, allowing developers to gain deep visibility into embedded software while automating the important steps in the process of binary scanning. BlackBerry Jarvis has been instrumental in helping OEMs bring trust and transparency into their software supply chains.



FROST  SULLIVAN

Growth is a journey. We are your guide.

For over six decades, Frost & Sullivan has provided actionable insights to corporations, governments and investors, resulting in a stream of innovative growth opportunities that allow them to maximize their economic potential, navigate emerging Mega Trends and shape a future based on sustainable growth.

Contact us: [Start the discussion](#)

The contents of these pages are copyright ©2021 Frost & Sullivan.