What You Don't Know About Your Software Vendor May Hurt You...

By Eric Raisters, Secure Software Development Manager

All software – no matter how well designed and written – contains bugs. A certain percentage of those bugs result in vulnerabilities that attackers can exploit.

Commercial, off-the-shelf (COTS) software is licensed from a vendor and does not provide for easy review of the architecture, design or source code for security issues. Instead, you must rely on the vendor to provide information on security-related issues.

Attachmate works hard to develop reliable, standards-driven solutions that are validated by well-known industry and third-party certifications. Our products are designed and tested to meet important security requirements and to deliver a superior experience with the latest platforms and technologies. In addition, our products meet rigorous industry standards and have been awarded a number of certifications to ensure the highest possible level of security.

For software products, there are many considerations to keep in mind when evaluating software. The IT department is often held responsible for any breaches that may occur within the organization, although it is C-level executives (CIO, CSO, CISO) who are ultimately responsible for data breaches or fraud. While zero-vulnerability software is likely to remain an unattainable "Holy Grail" for software development, there are steps you can take to assess the risk of using COTS software and protect your organization. This requires information from vendors about their security competence, secure development practices and vulnerability disclosure policies.

Look Beyond Security Certifications

Security certifications, while important, are not enough to show that a product is secure. Products can be validated to standards like FIPS 140-2, ISO 27001 or Common Criteria, but these usually are just validations of the cryptographic and security related parts of products – not necessarily the product in its entirety. A bug in non-security features can be even more exploitable than one in the security functionality because the defense in depth and fail-safe principles of secure software may have not been implemented there.

After establishing that a vendor has acquired security certifications for its product, organizations need to dig deeper to ensure they are deploying safe software.

Secure Development Practices

It is important to understand if security is built into the vendor's entire lifecycle development process, or if security is simply an afterthought.

After justly being criticized for the security of the Windows operating system due to a series of well-publicized vulnerabilities, Microsoft developed and now advocates a rigorous Secure Development Lifecycle (SDL) methodology that it has made public, including many tools for Windows developers to assist in implementation. It took Microsoft nearly three years to complete the initial round of this process for the Windows operating system.

Most vendors don't have the knowledge, staff or funding to spend three years retooling their software development practices. However, all vendors have the capability – and the responsibility – to adopt SDL methodologies appropriate for the type of software they develop, the platforms they support, and the markets they target.

Vendors should build a culture of security awareness, equip developers with the training and resources they need and let them intelligently implement SDL practices in phases. The first phase involves selecting and adopting those SDL tasks that provide the most "bang for the buck" to secure applications. Subsequent phases quickly enhance follow-on releases with improved security and stability.

Architects and developers need to be regularly trained on the best practices for designing and writing secure applications and servers. This includes designing in fail-safe mechanisms. When (not if) a bug is encountered, such mechanisms help prevent software failure that may result in an exploitable vulnerability. In addition, analyzing an application's attack surface and performing threat modeling are techniques software architects and developers can use to understand where their products need special attention, and to ensure that vulnerabilities are properly mitigated.

In addition, application security should be tested during development and prior to shipping. Many tools used by hackers are now available to "white hat" testers, who can use them to check applications for known exploitable vulnerabilities. Tools, like static and dynamic analyzers, are often built into development environments to catch common errors that lead to exploitable vulnerabilities. File and protocol fuzzing also can be performed to find the kind of particularly egregious bugs that result in exploitable vulnerabilities.

Attachmate adheres to an SDL methodology as a part of its standard product development process.

Vulnerability Disclosure Policies

It also is important to communicate with the vendor around vulnerabilities that may be found in your software environment after purchase. Prior to purchase, ensure you have the following information pertaining to software vulnerabilities that may be discovered after purchase.

- Determine if your vendor has a process by which you can securely report vulnerabilities. Often the troubleshooting logs required by the vendor contain sensitive information that cannot or should not be sent in the clear. It's also preferable for vulnerabilities to be communicated privately to give vendors an opportunity to address them before they are posted in open forums. Attachmate maintains a site specifically for reporting potential security vulnerabilities.
- Learn the vendor's process to inform you of their findings, communicate possible workarounds and let you know when patches are available. If a vendor does not notify you of vulnerability, you cannot assess its potential to affect your business. More importantly, you cannot take steps to prevent an attacker from exploiting the vulnerability in your environment.
- Understand the vendor's vulnerability disclosure policy. Oftentimes, vendor vulnerability statements do not provide sufficient information to determine what a particular vulnerability can exploit, so you do not know if it applies in your environment. This typically means you have to apply security patches as soon as they become available – no matter how often that might be. Vendors claim this is so that other attackers cannot use the information to develop exploits against unpatched installations, but attackers tend to share their findings (and exploit tools) widely.
- Find out if the vendor works with national computer security incident response programs like US-CERT, CSIRT-UK or FIRST.org. These reporting organizations enable customers to keep track of reported vulnerabilities and available patches in vendor-supplied software. They also maintain vulnerability databases that allow vendors to obtain information on software vulnerabilities in other vendor products that may be included as third-party libraries or add-ons in their own products. In addition, vendors may find vulnerabilities in their own products that are similar to the ones reported in a competing vendor's product.

Buy Smarter, Sleep Better

When evaluating COTS software products, it is also important to evaluate the vendor who produces the software. Asking to talk to your software vendor's security specialists about their security competency, secure development, and vulnerability disclosure practices will go a long way to increasing your confidence in the security of the product when used in your environment. You'll sleep better at night knowing that your COTS products are not the weak link in your organization's security chain.

If you have any questions about Attachmate products' certifications, please visit our security updates page or contact us.

About our author

Eric Raisters, CISSP, CSSLP is the Manager of Secure Software Development at Attachmate Corporation and has 15 years of testing and development experience on Attachmate's security products.