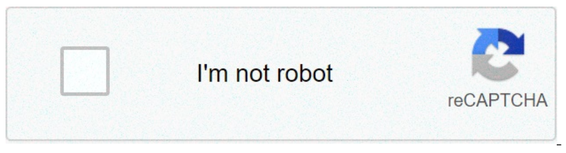

Continue

Fortify application security

Application security, piracy and prevention all are hot topics recently, with good reason. Without a robust application marketplace, that hundreds of thousands of new activations each month number isn't going to be maintainable, and a robust market isn't possible without the backing of developers. We've seen that Android has a built-in solution to prevent piracy, and we've also seen just how easy it is to get around it if you're determined, and if the scheme is left in its basic form. Google hinted that they had some more information to share about the whole subject, and true to their word they have done so. After the break, let's have a look at a Googler's methods to provide safe, secure, and user friendly way to protect applications. [Android Developers Blog] The Android Market Licensing Service and the License Verification Library are powerful tools for developers to try to circumvent application piracy. The problem, as was demoed recently, is that out of the box it isn't very hard to bypass. Since people are people, and many will spend more time than it's worth to crack a 99-cent application from the Market, Trevor Johns (one of Android's Dev. Program Engineers) has laid out a handy set of tips to strengthen the supplied tools, and make the anti-piracy measures work better.The four key areas are:Code ObfuscationCode obfuscation is a trick used by developers that changes the source code, making known functions, packages, classes and variables very hard to track down by providing an alias to each. Take this imaginary function for example -- onRedraw(). Each place you use the function in the source code, it's right there, easy to read and possibly exploit. A code obfuscator will replace the human readable function with a generated alias -- wy23() is a good example. A quick glance (or an automated tool) looking for functions isn't going to work, as it takes some serious digging to see exactly what wy23() really means. There are commercial Java code obfyousk8tors (ha!) available, and Trevor recommends ProGuard, and plans a future article on the Android Developers Blog about working with ProGuard.Modifying the License LibraryGoogle recommends that developers change the source of the supplied license libraries as much as possible while still retaining the original function. This is one case where the path taken is unimportant, as long as the destination is reached. Developers can bury function in if/then statements, loops, even mve the entire library into their own code block.To go a step further, developers are encouraged to used hash checks and other encryption methods to generate new constants, and change the code to look for the new constants instead of using the ones provided by Google in the example code. Be sure to hit the source link to see a great example right from Google showing how this can be done. And don;t forget to obfuscate the code here either!Make your application tamper resistantThis one's simple. For a hacker thief to remove the licensing from your application, he or she has to reverse engineer and rebuild the application. Use CRC checks to prevent this. Google has another handy tool for this area too -- verify that the Android Market was the installation source of your application, and if not, don;t let it run. Again, there's a nifty example of this at the source link.Move the license verification to a remote serverIf your application uses online components, Google recommends that you move the LVL information and response out of the app and on to your server. When the user uses the app, your server checks with Google, and if everything isn't kosher, no content is served. While simple, it's also very effective as to get around this, someone would have to change not only the application, but content on your server as well. Remember, local data is never safe, but a properly maintained and secured server is a pretty tough nut to crack. Lastly, Google remembers us -- the end users, and recommends that these tricks be used in a way that's transparent and user friendly. If you're an application developer interested in the integrity and piracy prevention of your app (and you should be!) be sure to check out the source link. It gets all geeky and fuzzy and lays it all out for you. For the rest of us, this is more of a reminder about how Goggle loves it's dev's, and we can feel good knowing that big G is doing what it can to help. Security cannot be an afterthought when developing and deploying Big Data commercial applications. If you must ask whether your big data applications are secure, the answer is probably no. What’s the answer? Mitigating risks and ensuring security requires the ability to leverage existing identity infrastructure, control access privileges, and conduct user-level audits.Commercial applications are increasingly available on Big Data infrastructure, allowing the enterprise to leverage extremely large data sets to swiftly curate, manage, and process information. But it’s crucial to address the potential security risks inherent in processing such large volumes of data.Big Data applications housing sensitive and personally identifiable information (PII) are typically governed by corporate security and regulatory compliance policies. Least-privilege access should be granted only to users who need access to these resources to perform their jobs.The enterprise also needs to control access, manage privileges, audit activities, and associate all access activities back to an individual. This allows you to mitigate threats resulting from identity-related risks while successfully addressing audit and compliance requirements.For example, by deploying the Centrify Server Suite, the enterprise can leverage existing Microsoft Active Directory infrastructure to standardize Big Data cluster operations. IT can secure and simplify Big Data environments at the operating system layer without deploying and managing new identity infrastructure. You can also increase security by implementing privileged identity management solutions for leading Big Data environments, such as Apache Hadoop as well as Big Data solutions from Cloudera, Hortonworks, and MapR Technologies.Enabling simplified and secure accessThe enterprise can benefit from simple and secure access to these Big Data environments, and leverage Active Directory to avoid the hassle of introducing alternative solutions that do not scale and require additional training for IT.Implementing single sign-on (SSO) for both IT administrators and Big Data users allows you to further extend the power of Active Directory’s Kerberos and LDAP capabilities to Big Data clusters. By adding SSO functionality to Big Data environments, you can make users more productive and secure by allowing them to log in as themselves, rather than having to share privileged accounts.Tracking user activities and documenting compliance with regulatory requirements and enterprise policies also reduces identity-related risks. Deploying an identity management solution that can track user activity and associate it with an individual in Active Directory therefore helps the enterprise make Big Data more secure. By enforcing access controls and least-privilege security across Big Data infrastructure, the enterprise can also benefit from cost-effective compliance through combined access and activity reporting.For additional information on how to make sure that your big data commercial applications are secure, download the How Identity Management Solves Five Hadoop Security Risks whitepaper. Copyright © 2016 IDG Communications, Inc. By Yen Hoe Lee, Director, KPMG AdvisoryThere is an old joke among software developers that if you’d just write it correctly the first time, you’d never have to waste time debugging or testing your code. Anyone even remotely familiar with software development can see the absurdity of perfection that makes this a joke. Today, even the simplest of applications is a machine of almost unimaginable complexity in which a single misplaced character or seemingly innocuous line of code can create a serious flaw or vulnerability.When it comes to application security (AppSec), it might appear as if some organizations have taken this joke seriously. Instead of performing AppSec testing on every release of every application in their portfolios to detect and remediate critical vulnerabilities, they’re testing only sporadically or selectively, perhaps testing only the three or four most “risky” apps each release. Of course, no one really believes that they’ve just written it correctly and so no testing is needed. The unfortunate truth is that many aren’t testing every release because they simply don’t have the resources or the time to do it.No time to testFueled by new techniques and methodologies such as agile, DevOps and CI/CD, the pressure on developers to deliver faster has never been greater. Reportedly, many of the well-known Internet platforms push updates every few hours – not weeks, not days. So, what do you do when your thinly-staffed AppSec team tells you it needs two or three days to properly evaluate a release (not counting time for remediation) when you’re trying to maintain a pace like that across not just one but maybe even dozens of applications?For some, despite a genuine desire to do the right thing, the answer is the equivalent of closing your eyes, plugging your ears and repeating “la-la-la” because if you don’t know about the vulnerabilities, they can’t hurt you, right?The not so secret secret is that many organizations have such visibility gaps in their application security. Many see AppSec as a major source of friction in the development process, and so it’s natural and tempting to want to reduce or eliminate that friction – or at least to try throwing technology at the problem to compensate for the lack of time and resources. Technology can indeed help – in fact, it’s indispensable – but it’s not a panacea.Automation is a tool, not a solutionWhile there are a dozen or so well-respected software packages designed to automate AppSec testing, none can truly evaluate vulnerabilities in the context of your business or your industry. They can’t prioritize the most critical issues specific to you and your organization and its customers, and often simply spit out a list of issues that’s so long it might as well be infinite. And they can’t look at your organization as a whole to see if it has fostered a culture of security, if it’s optimized to maintain application security in the most efficient way or if its security practices are aligned with overall business goals.As with most tools, the quality of the outcome is only as good as the skill of the person using it. Many organizations struggle to find skilled and seasoned AppSec data analysts who can interpret the output and make informed recommendations in terms of business risk.It’s also not uncommon for large organizations to have different teams using different AppSec tools, or to outsource development projects to multiple teams that have their own tools, which also makes it difficult to get a single, complete view of application security – and the potentially catastrophic business risks you may be exposed to.Without that visibility, it’s nearly impossible to determine where to focus resources to maximize efficiency and effectiveness, or to properly manage risk. It’s the software equivalent of driving a car without being able to see completely out the windshield, not knowing when to hit the brakes or when you can accelerate. As they say, good brakes aren’t designed to make cars to go slower – they’re designed to allow them to go faster.The same is true for AppSec. Done right, AppSec can not only be a frictionless part of the development process, it can actually help you accelerate it.NextGen AppSec SolutionsIn the last few years, a new generation of on-demand AppSec solutions have evolved to address exactly that goal, including on-demand solutions from KPMG. These NextGen solutions are designed to scale to meet the needs of organizations with large application portfolios, multiple development teams and aggressive development schedules. They’re also designed to eliminate gaps in AppSec visibility by aggregating data across disparate tools and teams – without having to first normalize the data.There are many potential efficiencies that can be exploited with better visibility. For example, you might see that SQL injection issues are common across multiple development projects. Rather than addressing each discretely and repeatedly, you might add more training around SQL injection, or create a central, common library of code that’s been designed to address the problem.Enhanced visibility also applies to what’s happening outside of your organization, too. It can be enormously helpful to benchmark your performance against industry norms to understand where more attention needs to be paid – and where you’re already doing well.Another key aspect of an on-demand solution is that it’s designed to consider the entire process, not just the applications themselves. AppSec doesn’t exist in a vacuum or in discrete application silos. Optimizing your organization for greater efficiency and speed of development might involve disciplines such as risk management, digital transformation, organizational change management, regulatory compliance, and more – things that no software-only solution can deliver.You can learn more on how KPMG is helping organizations achieve modern delivery of IT here or drop me an email to talk more about your AppSec requirements.This article represents the views of the author only, and the information contained herein is of a general nature and is not intended to address the circumstances of any particular individual or entity. No one should act on such information without appropriate professional advice after a thorough examination of the particular situation.Some or all of the services described herein may not be permissible for KPMG audit clients and their affiliates or related entities. Copyright © 2020 IDG Communications, Inc. Last week, I was talking to a colleague about companies that monitor employees’ online and computer usage. He retold a tale about a female coworker who had announced she was leaving the organization. One day, he looked over at her workspace and saw that the cursor was moving around the desktop on its own. Folders were opening. Files, opening. Files, closing. Someone on the IT team was managing her desktop and computer files remotely. An eerie, intrusive experience. (He turned off her monitor and tried to think nothing more of it.)Shortly after that conversation, I was talking to a friend who’d left one of his previous employers because of a falling out with the GM. Even though he was on fine footing with the company’s founder, the GM had it out for him. When he got the sense that she was snooping around on his computer after hours, he began to leave Easter eggs for her: Word documents that contained text like “I know you’re reading this,” fake file folders with provocative names, and so forth.That’s one way to deal with smaller-scale, grassroots surveillance, but how can employees work assured that more organized efforts won’t cramp their work style? An article taken from CIO magazine has some hints. It’s not just that you monitor employees (which I find somewhat questionable in most cases, granted), it’s how you do so. Something to ponder.[via George’s Employment Blawg]