How to Secure Your Modern Mobile Apps and Prevent Data Breaches

datatheorem

Prevent AppSec Data Breaches

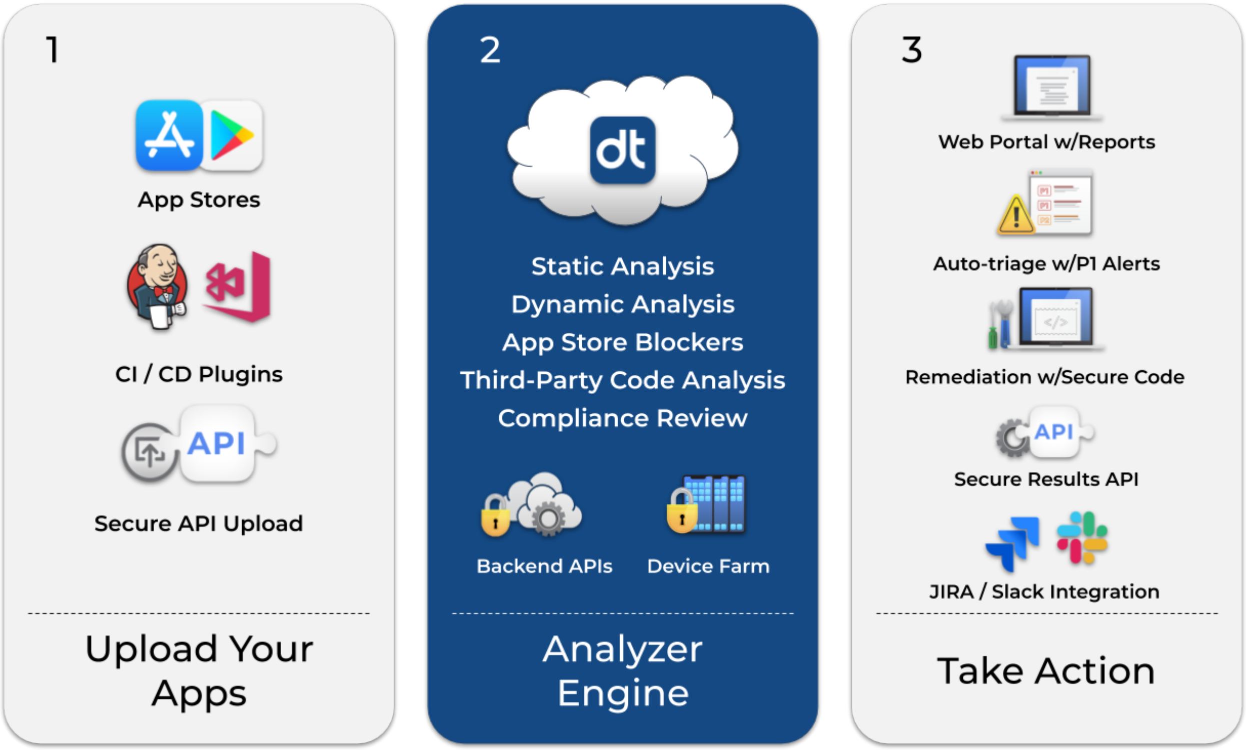
Introduction

The footprint of a modern application is constructed by multiple teams. Mobile application developers use a number of languages to natively write their code. Many developers will include third party open source libraries or commercial SDKs for various reasons like performance, compression, analytics and advertising. On average, a given mobile application will include twelve to eighteen third party SDKs, with only 50-60% owned by the publisher. While the iOS and Android security models sandbox applications, there is no separation for embedded third party software. Since these modernized applications exhibit the common practice of fully integrating this third-party code, it becomes a challenge for fast paced development teams to keep up with securing all application paths. How can you protect your data and business in this age of data breaches? According to a Ponemon Institute Study, companies have a 28% chance of having a data breach incident in the next two years. And according to Verizon's data breach report, 56% of data breaches took "months or longer" to discover. If you rely on traditional testing or device security to protect your apps, you may want to reconsider your strategy.

Limitations of Traditional approaches

Web and mobile applications are the most frequently attacked and compromised vectors in a company's security posture, and that includes all business verticals. The underlying architecture of most modern systems is not compatible with traditional security approaches. The open, universal, and connected inherent nature of Application Programming Interfaces (APIs) works great for developers, but it creates security problems that common approaches like endpoint security tools and penetration tests cannot solve in a comprehensive way. Mobile apps and APIs create encrypted, machine-to-machine interactions on the network. A shadow API won't show up as a compromised endpoint, as oftentimes embedded APIs within a mobile app are often not being governed by IT security personnel who are asked to be responsible for data privacy and compliance. As a result, a masked attacker can appear to be an approved user, unbeknownst to you or your security filters.

Traditional mobile security measures like penetration testing or scanners won't protect your app from reverse engineering or tampering. But you don't need a Fortune 500 budget or large security staff to tackle this problem. Data Theorem's Mobile Application Security Program, App Secure, provides a common platform for establishing and monitoring application security. Our platform gives developers the ability to implement security practices that they may not know how to do on their own—or have time to do. As a program, Data Theorem takes security one step further by ensuring that all endpoints are accounted for and that you can immediately take action. A high level view of Data Theorem's approach is shown below:



In the next few paragraphs, we will go into each pillar and how this approach works to protect your applications.

Taking Action

Protecting your application can be seen as a burden and the thought of additional resources or slowed development is often a concern. When working with a security program, there must be a few considerations, including shared and agreed upon risk definitions which involve analyzing both back-end APIs in the application native code and third-party code, as well as a review which includes app store blockers and regulatory compliance. The former requires both static and dynamic forms of analysis while the latter is oriented around keeping up to date with the latest standards and processes.

The Analysis

At minimum any application security program should include static and dynamic analyses. Static analysis of your binary exposes threats such as calls to third parties or insecurely stored user data or classes. Similarly, dynamic analysis is run on an emulator or on a device farm and will reveal any issues through testing all paths of the code. While your company may have set definitions of security internally, whether these need to be fixed or not can be convoluted. To simplify this, Data Theorem defines the highest priority P1, defined as vulnerabilities that allow a remote attacker to export data from an app. Data Theorem checks for back-end APIs and third-party SDKs and libraries and auto-triages them as P1 for you, while including secure code examples on how to securely implement remediations.

Additionally, the Brand Protect feature will ensure that your brand's reach is not being exploited by malicious actors. Our Analyzer Engine continuously monitors global app stores for counterfeit and fraudulent clones of your apps. Leveraging machine learning, we identify any pattern matched logos or similar brand usages for fraudulent apps protecting your brand.

Review and Keeping up to Date

When it comes to publishing on the app store a two-party system exists consisting of Apple and Google. In the United States, these app stores implement processes that affect development and security enforcing prerequisites to publishing releases or updates. They have low and inconsistent security requirements which if not met, can result in removal of apps from the app store, potentially halting your business.

Even beyond the app store requirements, there exist federal, state security, and compliance standards. Companies need to make sure they are compliant on a regular basis. There are old and new mobile privacy laws - PCI, GDPR, CCPA, and more privacy legislation has been proposed by other U.S. states. Privacy is nothing new to mobile, but now there are giant compliance issues that did not exist years ago. For example, is your app collecting geolocations? How are you storing it and who are you sharing it with? Let us assume you are using the data for a legitimate business purpose, but you are storing that data with an analytics SDK that is housed in California. CCPA guidelines state that in California, the geolocation or trackers cannot be sent to a third-party without consent.

Now even as a New York-based company, you are not compliant with CCPA and will be subject to fines for a breach of this data. Now that you may better understand the implications of not securing your application, you may consider the effectiveness of a few common strategies shown in the table below.

Security Strategy	Time to Fix/Secure	Time to Discover	Cost (Least to Most Economical)	Customized on Premise	Continuous Security	Avg. Security Score
SaaS, fully automated tool	🟩	■	🟩	🟨	■	■
In-house security staff	🟩	🟨	🟥	■	🟧	🟨
External contractors	🟥	🟧	🟧	🟩	🟥	🟧
No additional counter-measures taken	🟥	🟥	■	🟥	🟥	🟥

	Key
🟥	At risk
🟧	Moderately at risk
🟨	Additional effort required
🟩	Compliant security
■	Comprehensive security

As seen in the table, no additional countermeasures may be most economically beneficial, but may result in risk of data breaches. Both in house staff as well as external contractors are a common solution but continuous monitoring will be a concern. Keeping in sync with security personnel on a regular basis will be vital to catch any vulnerabilities that arise throughout days, weeks, or months. Meeting with an external contractor weekly or hiring one for a specific project may seem to streamline security activities but may result in missed security concerns. A low cost, SaaS-based approach is easily integrated and catches flaws quickly and effectively. Not only can the Data Theorem mobile program help streamline some of the security steps required for production, it can also save you time and money.

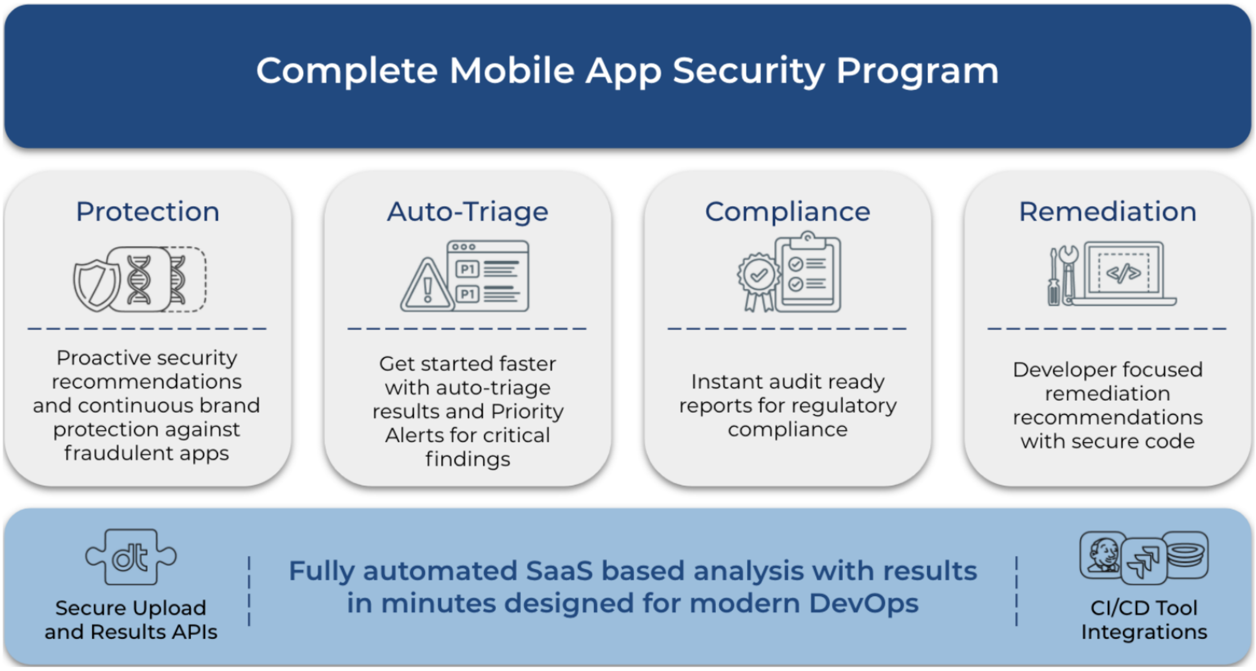
Automation in the SDLC with DevSecOps integrations

Penetration testing (or pen testing) is a common strategy for application security. But let us review why this strategy should be abandoned. Pen testing was a reliable strategy to evaluate app security before major releases. But software development today requires updates monthly, weekly and in many cases daily. This is why the development world has now moved on to DevOps strategies in which apps are scanned and tested automatically and updates are released non-stop. New, automated security tools for apps and APIs make it possible to balance these competing requirements offering a DevSecOps model.

Continuous scanning enables DevOps teams to narrow the gap between the discovery of a security issue and its remediation. Security flaws can appear at anytime, including during transition points such as a new product version release. In these moments, users can accidentally leave APIs exposed to malicious access and other threats. The Data Theorem program increases visibility into all relevant APIs and setting alerts to discover shadow APIs. APIs need constant watching, for example finding authentication and encryption vulnerabilities in APIs based on their definitions and API specifications. Malicious actors can set up shadow APIs that steal user data by mimicking real APIs.

In summary, Data Theorem maintains DevSecOps capabilities which includes:

1. Audit-Ready 24/7: Allows anyone in the company to pull a report, at any time, to prove or analyze regulatory compliance status.
2. True DevSecOps: Integrates with CI/CD tools such as Jenkin or App Center. Jira, Bugzilla, etc to send alerts, analyzes the best way to resolve quickly and safely, and provides secure code samples and recommendations for remediation.
3. Beyond Pen Testing: Continuous (24/7), automated dynamic and static analysis of vulnerabilities and app store blockers.



Complete Mobile App Security Program

Protection

Proactive security recommendations and continuous brand protection against fraudulent apps

Auto-Triage

Get started faster with auto-triage results and Priority Alerts for critical findings

Compliance

Instant audit ready reports for regulatory compliance

Remediation

Developer focused remediation recommendations with secure code

Secure Upload and Results APIs

Fully automated SaaS based analysis with results in minutes designed for modern DevOps

CI/CD Tool Integrations

A DevSecOps approach gives users the best-performing, most-secure app experience and companies remain compliant.

How Data Theorem Can Help

Now that you have learned a bit about how Data Theorem can help, you may wonder how clients have benefited from the program. Here's an example of an on-boarded customer who began scanning all of their apps in pre-production shows tremendous results. In their first 90 days, they were able to close out over 75 security issues, and identified 51 compliance and regulatory issues. Today, multiple product and security teams within the company use our tools to be proactive against data breaches instead of reacting to issues on a daily basis. Another customer says "Data Theorem has helped secure our mobile banking app and catch potential vulnerabilities prior to end user release."We understand that your reputation is a critical piece to your business and that avoiding a data breach is of the highest priority. This is why security automation is necessary to not only find the vulnerabilities, prioritize them, and offer remediation options- but also to take action and save time for your team without additional overhead or resource. The end result is worth it when it comes to protecting your users' data.

Data Theorem is a leading provider of modern application security. Its core mission is to analyze and secure any modern application anytime, anywhere. The Data Theorem Analyzer Engine continuously scans APIs and mobile applications in search of security flaws and data privacy gaps. Data Theorem products help organizations build safer applications that maximize data security and brand protection. The company has detected more than 400 million application eavesdropping incidents and currently secures more than 4,000 modern applications for its Enterprise customers around the world.