



5. Both the iTunes App Store and the Google Play Store list Credit Karma Mobile among the top 10 free applications in the Finance category. The application has been downloaded over one million times.
6. When a consumer creates an account through the Credit Karma Mobile application, the application transmits sensitive personal information to Credit Karma, including the consumer's email address, password, security question and answer, first name, last name, date of birth, street address, apartment number, city, zip code, phone number, and Social Security Number. During the account creation process, the application also transmits the consumer's answers to "out of wallet" questions, which are multiple choice questions validating the consumer's identity (*e.g.*, questions about a past mortgage provider or the payment amount on a loan).
7. Credit Karma outsourced the software development of both the iOS and Android versions of the Credit Karma Mobile application to application development firms that acted as its service providers and agreed to certain product security requirements.

### **SECURE SOCKETS LAYER CERTIFICATE VALIDATION**

8. Consumers frequently use mobile applications on public Wi-Fi networks in venues such as coffee shops, shopping centers, and airports. Consumers may use the Credit Karma Mobile application in such public environments. Indeed, Credit Karma marketed Credit Karma Mobile on the iTunes App Store and the Google Play Store as a way for consumers to get "free on-the-go credit monitoring."
9. Online services often use the Secure Sockets Layer ("SSL") protocol to establish authentic, encrypted connections with consumers. In order to authenticate and encrypt connections, SSL relies on electronic documents called SSL certificates.
10. In the context of mobile applications, an online service (*e.g.*, Credit Karma) presents an SSL certificate to the application on a consumer's device (*e.g.*, Credit Karma Mobile) to vouch for its identity. The application must then validate the SSL certificate – in effect verifying the identity of the online service – to ensure that the application is connecting to the genuine online service. After completing this process, the online service and the application on the consumer's device can establish a secure connection that is both authenticated and encrypted.
11. If the application fails to perform this process, an attacker could position himself between the application on the consumer's device and the online service by presenting an invalid certificate to the application. The application would accept the invalid certificate and establish a connection between the application and the attacker, allowing the attacker to decrypt, monitor, or alter all communications between the application and the online service. This type of attack is known as a "man-in-the-middle attack." Neither the consumer using the application nor the online service could feasibly detect the attacker's presence.

12. On many public Wi-Fi networks, attackers can use well-known spoofing techniques to facilitate man-in-the-middle attacks.
13. To protect against these attacks, the iOS and Android operating systems provide developers with application programming interfaces (“APIs”) that allow applications to create secure connections using SSL. By default, these APIs validate SSL certificates and reject the connection if the SSL certificate presented to the application is invalid.
14. The developer documentation for both iOS and Android warns developers against disabling the default validation settings or otherwise failing to validate SSL certificates. The iOS documentation explains that failing to validate SSL certificates “eliminates any benefit you might otherwise have gotten from using a secure connection. The resulting connection is no safer than sending the request via unencrypted HTTP because it provides no protection from spoofing by a fake server.” Similarly, the Android documentation states that an application that does not validate SSL certificates “might as well not be encrypting [the] communication, because anyone can attack [the application’s] users at a public Wi-Fi hotspot . . . [and] the attacker can then record passwords and other personal data.”
15. Application developers can easily test for and identify SSL certificate validation vulnerabilities using free or low-cost, publicly available tools.

### **CREDIT KARMA’S SECURITY FAILURES**

16. From July 18, 2012 to January 2013, the Credit Karma Mobile application for iOS failed to validate SSL certificates, overriding the defaults provided by the iOS APIs. On or around January 1, 2013, a Credit Karma user informed respondent that its iOS application was vulnerable to man-in-the-middle attacks because it did not validate SSL certificates. Respondent’s in-house security engineers issued an update to the application in January 2013 that enabled SSL certificate validation by restoring the iOS API default settings.
17. During the iOS application’s development, Credit Karma had authorized its service provider, the application development firm, to use code that disabled SSL certificate validation “in testing only,” but failed to ensure this code’s removal from the production version of the application. As a result, the iOS application shipped to consumers with the SSL certificate validation vulnerability. Credit Karma could have identified and prevented this vulnerability by performing an adequate security review prior to the iOS application’s launch. In February 2013, one month after addressing the vulnerability in its iOS application, Credit Karma launched the Android version of its application, again without first performing an adequate security review or at least testing the application for previously identified vulnerabilities. As a result, like the iOS application before it, the Android application failed to validate SSL certificates, overriding the defaults provided by the Android APIs.

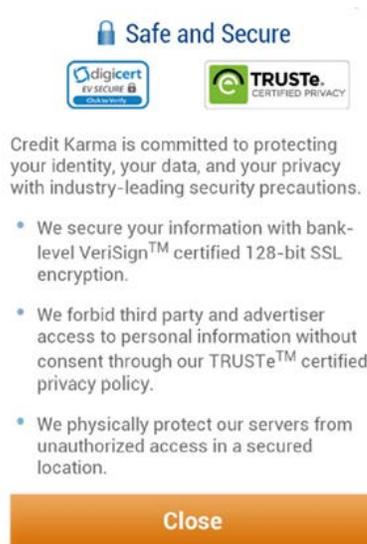
18. Credit Karma did not perform an adequate security review of the Credit Karma Mobile application until after Commission staff contacted respondent. At that time, Credit Karma's in-house security team performed a basic, low-cost security review of both the iOS and Android versions of the application over the course of several hours.
19. Through the security review, respondent discovered that its service provider had introduced the same SSL certificate validation vulnerability into its Android application that respondent had been warned about and remedied in its iOS application just one month earlier. Respondent issued an update to the Android application in March 2013, enabling SSL certificate validation by restoring the Android API default settings. Credit Karma could have prevented the re-introduction of this vulnerability in the Android version of its application had it performed an adequate security review prior to launch or at least tested the application for previously identified vulnerabilities.
20. Through the security review, respondent's in-house security team also discovered that the iOS application was storing authentication tokens and passcodes on the device in an insecure manner, contrary to security requirements that the application development firm had agreed to implement (*i.e.*, encrypting this information with the "keychain" API provided by the iOS operating system). Credit Karma could have ensured the implementation of its product security requirements by providing reasonable oversight of its service providers during the development process and performing an adequate security review of its application prior to launch.
21. Respondent engaged in a number of practices that, taken together, failed to provide reasonable and appropriate security in the development and maintenance of its mobile application, including:
  - a. Overriding the default SSL certificate validation settings provided by the iOS and Android APIs without implementing other security measures to compensate for the lack of SSL certificate validation;
  - b. Failing to appropriately test, audit, assess, or review its applications, including failing to ensure that the transmission of sensitive personal information was secure; and
  - c. Failing to reasonably and appropriately oversee its service providers' security practices.
22. As a result of these failures, attackers could, in connection with attacks that redirect and intercept network traffic, decrypt, monitor, or alter any of the information transmitted from or to the application, including Social Security numbers, dates of birth, "out of wallet" information, and credit report information. Attackers also could intercept a consumer's authentication credentials, allowing an attacker to log into the consumer's Credit Karma web account to access the consumer's credit score and a more complete version of the consumer's credit report. The misuse of these types of sensitive personal information can lead to identity theft, including existing and new account fraud, the

compromise of personal information maintained on other online services, and related consumer harms.

23. Credit Karma could have prevented these vulnerabilities and ensured the secure transmission of consumers' sensitive personal information by performing basic, low-cost security reviews, such as the one described in paragraph 18.

### **CREDIT KARMA'S PRIVACY AND SECURITY REPRESENTATIONS**

24. Since the launch of the Credit Karma Mobile application on iOS and Android, Credit Karma disseminated or caused to be disseminated to consumers the following in-app representation when a consumer created an account using the application:



25. Since at least the launch of the Credit Karma Mobile application on iOS and Android, Credit Karma disseminated or caused to be disseminated to consumers the following representation in its privacy policy:

We enable our servers with Secure Socket Layer (SSL) technology to establish a secure connection between your computer and our servers, creating a private session.

### **CREDIT KARMA'S DECEPTIVE REPRESENTATIONS (Count 1)**

26. As described in Paragraph 24, Credit Karma has represented, expressly or by implication, that it is committed to protecting Credit Karma Mobile application users' identity, data, and privacy with reasonable and appropriate security practices.
27. In truth and in fact, as set forth in Paragraphs 16 – 23, Credit Karma failed to protect Credit Karma Mobile application users' identity, data, and privacy with reasonable and appropriate security practices. Therefore, the representation set forth in Paragraph 26 was false or misleading.

**(Count 2)**

28. As described in Paragraphs 24 and 25, Credit Karma has represented, expressly or by implication, that the Credit Karma Mobile application transmits consumers' sensitive personal information over secure SSL connections.
29. In truth and in fact, as set forth in Paragraphs 8 – 19, the Credit Karma Mobile application did not transmit consumers' sensitive personal information over secure SSL connections. Therefore, the representation set forth in Paragraph 28 was false or misleading.
30. The acts and practices of respondent as alleged in this complaint constitute unfair or deceptive acts or practices in or affecting commerce in violation of Section 5(a) of the Federal Trade Commission Act, 15 U.S.C. § 45(a).

**THEREFORE**, the Federal Trade Commission this thirteenth day of August, 2014, has issued this complaint against respondent.

By the Commission, Commissioner McSweeney not participating.

Donald S. Clark  
Secretary