Steal Your Life Using 5 Cents: Hacking Android Smartphones with NFC Tags

Carlos Bermejo*, Huber Flores†, Pan Hui*†
*Department of Computer Science and Engineering, Hong Kong University of Science and Technology
†Department of Computer Science, University of Helsinki
cbf@cse.ust.hk, huberflo@cs.helsinki.fi, panhui@cse.ust.hk

***Abstract*—Nowadays, privacy in the connected world is a big user's concern. The ubiquity of mobile devices permits billions of users to browse the web at any time, anywhere. Near Field Communication (NFC) appeared as a seamless communication protocol between devices. Commercial services such as Android Pay and Apple Pay offer contactless payment methods that are spreading in more and more scenarios. However, we take risks while using NFC on Android devices, we can be hacked, and our privacy can be affected. In this paper, we study the current vulnerabilities in the NFC-Android ecosystem. We conduct a series of experiments, and we expose that with NFC and Android devices are vulnerable to URL/URI spoofing, social network information hacking, and user's device tracking via fingerprint and geolocation. Smart devices based on NFC communication should inform and protect the users about the privacy risks using these contactless services.**

I. Introduction

The web has become an essential part of our society, and it is currently the main medium of information delivery. Billions of users browse the web daily, and there are single websites that have reached over one billion user accounts. In this environment, the ability to track users and their online habits can be very lucrative for advertising companies, yet very intrusive for the privacy of users. Many web services and Internet service providers (ISPs) aims to track the mobility and usage patterns of client hosts, the so-called device fingerprinting [1]. The rising awareness of privacy concerns is illustrated in the cookies consent notices, or location access notification messages on smartphones or browsers. Device fingerprinting using the browser as the information channel is a main concern in the privacy field as it raises the conflict between adaptability and user privacy. Fingerprinting techniques are still in the newspapers[1] as it appeared to be the recent suspicious method to Apple users that use Uber. If we want to have more user-friendly individual characteristics user interface (UI) in the ubiquitously connected world, we might need to open some doors to developers to enable them to access these individual features. However, this open door is not always used to improve the user experience (UX) but to collect individual information, the fingerprint, where our traces in this connected world can be uniquely identified.

The forthcoming Internet of things (IoT) paradigm is stepping inside our lives. Since modern smartphones, we have been surrounded by a myriad of sensors to improve our lives, such as sensor networks, wearable devices. NFC protocol has received special attention in many research studies and commercial systems as an efficient and simple approach to interact with the IoT ecosystem. NFC's main features are low energy requirements and its limited data transmission in comparison with other wireless protocols such as Bluetooth, WiFi. Despite its prevalent use in current mobile networks, there are several existing or potential vulnerabilities of NFC protocols. In [2], the authors investigate a wide range of these weaknesses, including eavesdropping, URI obfuscation, tag tampering, relay attacks, data corruption, man-in-the-middle, and worms or malware attacks. It also provides possible detection and mitigation mechanisms towards each of these susceptibilities. Most NFC communications do not include an encryption mechanism since it assumes that the short communication range (i.e., less than 4 cm) can guarantee the security. However, as we describe in Section III, there are several studied vulnerabilities with the NFC protocol. Due to the ubiquity of NFC as a fast, simple protocol for small data transactions such as public transport[2], contactless payments[3], supermarkets[4], and building access (hotel rooms) (Figure 1), we need to be aware of the vulnerabilities that our mobile devices can face. Some of the real-world applications require high-security measurements to avoid attacks (i.e., payments, location access).

In this paper, we explore different approaches to garner users' personal information using NFC-based connections. The simplicity of NFC transmission (tap and share) raises privacy threats that users are not fully aware of or do not require any user interaction to accept the requested transmission of data. We evaluate these attacks and their countermeasures against different Android OS and stock versions.

Summary of Contributions:

- We propose three different attacks based on the NFC protocol and Android devices: *(i)* social network, *(ii)* location, and *(iii)* fingerprinting attacks.
- We address these privacy threats and propose counter measurements to these attacks with a simple and efficient UI approach based on a notification permission request.

[1] https://www.nytimes.com/2017/04/23/technology/travis-kalanick-pushes-uber-and-himself-to-the-precipice.html?_r=0

[2] https://securityintelligence.com/is-nfc-still-a-vulnerable-technology/

[3] http://www.himanshutech.com/what-is-nfc/

[4] http://www.rfidjournal.com/articles/view?8793/2

(a) Transport.

(b) Contactless payment.

(c) NFC tags product labeling.

(d) NFC-tag closeup.

Fig. 1. NFC application examples.

- In light of the results, we discuss other innovative solutions. Moreover, we also highlight other different approaches that can be used for the deployment of the explored solutions.

II. NFC Characteristics

NFC is a set of protocols that enables wireless communication between two electronic devices within a distance of less than 4 cm. We can see NFC communications in public transport systems, office buildings as access cards, and also in the commercial contactless payments from VISA, Apple, and Google. The NFC protocol consumes very little energy, and its transmission speed capabilities are limited to less than 500 Kbps.

There are two types of devices to interact via NFC: *NFC-full devices*, NFC active device that can interact with other NFC peers, *NFC tags*, NFC passive data stores that can be read or written by another NFC-full device. The NFC-full devices can work in three different modes: *(1) Card emulation mode*: it enables mobile devices such as smartphones to act as an NFC card that an external NFC-reader can access. *(2) Reader/writer mode*: it enables the NFC device to read/write NFC-tags. *(3) Peer-to-peer mode*: it allows the NFC device to exchange data with other NFC peers, called Android Beam, for Android devices.

NFC support has started since Android version 2.3 (Gingerbread), December 2010. Android Beam has started in later versions since 4.0.1 (ICS). More complex NFC modes such as Host Card Emulation (HCE) have been supported since version 4.4.x (KitKat). Depending on the interaction intended, the device should be unlocked to complete the action embedded in the NFC, such as turn on WiFi or Bluetooth [3]. Other actions, such as create a new contact or open a web URL, might not require the users to unlock their device. These differences depend on the Android OS and stock versions (e.g., Xiamoi MIUI, Samsung TouchWiz).

In this paper, we focus on the NFC vulnerabilities in the NFC tag communication approach. Android devices look for NFC tags when the screen is unlocked, and depending on the OS version; the NFC tags can be read and trigger actions in locked users' smartphones. The default behavior is that the NFC embedded intents such automatically turn on the WiFi handle the action without asking the user what application to use[5]. Therefore, opening a URL link written in the NFC tag will automatically open and without users' interaction, the browser in the respective web address.

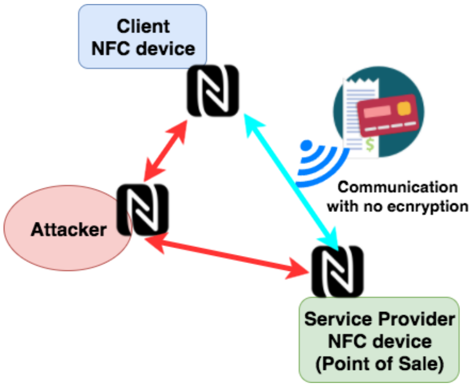

Client NFC device
Communication with no ecnryption
Attacker
Service Provider NFC device (Point of Sale)

Fig. 2. NFC Point of sale eavesdropping attack example.

III. NFC Vulnerabilities

In this Section, we will describe some documented NFC vulnerabilities and other possible protocol weaknesses of the ecosystem NFC-Android devices. For the latter, due to the myriad of devices and Android OS versions combinations, not all the mentioned vulnerabilities affect the devices analogously.

Communication vulnerabilities. Most NFC communications do not include encryption mechanisms during its data exchange [4]; it relies on the short-range (i.e., less than 4 cm) to guarantee the absence of eavesdropping attacks. However, the attacker can still place the device (i.e., NFC tag or NFC reader/writer) between client and NFC provider (i.e., NFC contactless point-of-sale) to trigger a specific attack such as eavesdrop, URL/URI spoofing see Figure 2. This vulnerability can also be exploited to jam the data exchange between two parties by sending out a specific packet at the right timing, which can lead to a deny-of-service attack toward the NFC-service provider. Other attacks use rely-techniques to extend the coverage of the NFC protocol and, for example, make customers waiting in the line pay for another customer at the contactless NFC-based terminal (Point-of-Sevice) [5].

[5] https://developer.android.com/guide/topics/connectivity/nfc/nfc.html

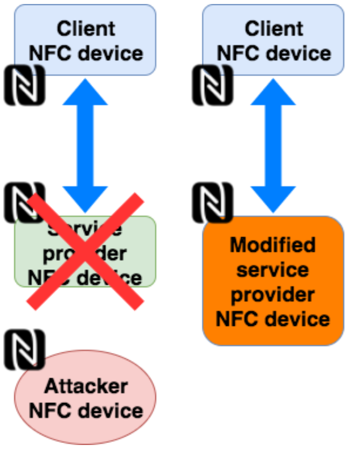


Fig. 3. NFC tag-device replacement.

URL/URI spoofing. NFC-based readers will read the content of an NFC tag. The information included in the NFC tag is not verified, and the NFC device will open the corresponding URL/URI automatically. The spoofing attacks can be performed to trick the user into seeing false information as a valid one [6]. The attacker designs an exact copy of a trusted website to users. Therefore, the attackers use the counterfeit website to collect users' login credentials or other personal information of the users. Besides, the work also describes a uniform resource identifier (URI) or uniform/universal resource locator (URL) in combination with other attacks such as cross-site request forgery.

Automatic and non-user intervention URL/URI connection. The proposed attack takes advantage of the non-user intervention when the device detects another NFC device in its proximity. The malicious NFC provides an URL/URI to attack the user's device, as the Android system does not request any user intervention, the device will automatically open the provided link by either other smartphone or NFC-tag. This situation opens security and privacy threads for the device's owner. Once the device opens the link, it can be attacked by fingerprinting mechanisms or share the user's location, for example (see more details in Section IV. The URI can also open application services such as *contacts* to automatically add malicious contacts without user permission requests.

NFC authentication vulnerabilities. When the NFC reader reads information from another NFC-enabled device, there are not any authentication mechanisms available. Therefore, there is a potential risk of tag replacement and tag hiding (TRTH) attack [6]. In the TRTH scenario, the NFC tags are overwritten or replaced (Figure 3) by an attacker with malicious information, malware injection [7]–[9], or worms that are installed into users' devices and are activated every time by similar tags (deployed by the attacker) [6].

IV. Attacks

In this Section, we proceed to enumerate the different attacks that can be leveraged using the NFC protocol and Android devices. Due to the myriad of Android devices and the different configurations regarding hardware and software, the proposed and also claimed vulnerabilities in this Section can be effective or not. We have tested several device configurations, see Table I.

TABLE I
Mobile Devices Tested

Device	Android OS	Stock Version	Browsers
One Plus 3T	Android 7.1.1	Stock	Chrome Firefox Pyro
Xiaomi Mi3W	Android 5.1	MIUI 7	Chrome Firefox Native Browser
Xiaomi Mi3W	Android 6.0.1	MIUI 8	Chrome Firefox Native Browser
Samsung C7	Android 6.0.1	Touchwiz	Chrome Firefox Native Browser

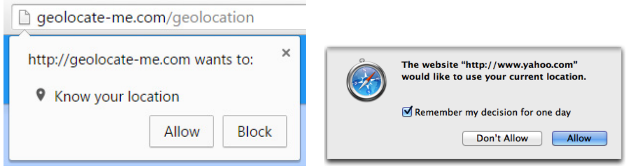
Threat model. Android NFC aims to work seamlessly without user intervention. However, once the device is unlocked, and NFC enabled, it starts looking for nearby NFC tags in order to read the data stored. This scenario can lead to our proposed attack: open a new channel to attack the user's device. The attacks we focus on this paper are: *(i) social attack* take advantage of already logged sites to take actions (i.e., liked a specific web site via Facebook) or gather social network information from user's profile; *(ii) location attack*, the user's device will send its current location via NFC tag; *(iii) fingerprinting*, the NFC-tag will trigger a fingerprint attack on users' devices. The attacks can be achieved, placing NFC-tags in different locations so that the attacker can infer users' movements, social information, and device characteristics. The location of these malicious NFC-tags can be: *(1)* in areas where the public transport uses NFC-based transactions; *(2)* placing NFC-tags under coffee tables or in locations where users tend to leave the device unlocked. For both situations, we can also collect the mentioned social network profiles or leverage more complex attacks in combination with other documented browser vulnerabilities. The enumerated attacks are included in a Github repository (see footnotes).

Social attack. Several social API such as Facebook or Twitter include automatic actions that can be triggered using parameters inserted in URL (written in the NFC-tag). These parameters can enable, for example, the collection of users' profiles by an attacker's profile using the Facebook API[6]. In the case of Twitter, its API provides web intents that redirect the users to follow a specific Twitter account or like a particular tweet[7]. Some of the mentioned methods in these social APIs might require user intervention. The attacker also requires the creation of a fake profile or account to lead the users to 'request friendship' or 'follow.' These attacks are

[6] https://github.com/solrac1986/nfc_attacks/blob/master/facebook.md
[7] https://github.com/solrac1986/nfc_attacks/blob/master/twitter.md

TABLE II
MOBILE DEVICES RESULTS

Device	Android OS	Status	Result
One Plus 3T	Android 7.1.1	locked	unsuccessful
		unlocked	unsuccessful
Xiaomi Mi3W	Android 5.1	locked	successful
		unlocked	successful
Xiaomi Mi3W	Android 6.0.1	locked	successful
		unlocked	successful
Samsung C7	Android 6.0.1	locked	unsuccessful
		unlocked	successful


geolocate-me.com/geolocation

http://geolocate-me.com wants to:

Know your location

The website "http://www.yahoo.com" would like to use your current location.

☑ Remember my decision for one day

(a) Chrome location pop-up. (b) Safari location pop-up.

Fig. 4. Browser location-request information pop-ups.

online successful if the users are already logged in in the social accounts, which is usually the case.

Location attack. Location information is a valuable asset that even modern browsers and mobile OS try to protect via notification request messages, see Figure 4. NFC-tags open two novel approaches to track users' location in-the-wild. The *first* approach requires the attacker to embed in the URL the physical location (i.e., latitude and longitude) as parameters, which is known by the attacker[8]. Once a user's device approaches a specific NFC-tag, the device will open the browsers with the corresponding URL address. The *second* approach leverages a less fine-grained approach to collect the device's location. The IP contains information such as the city that can be retrieved from the Internet[9]. The attack will be a success if the device is unlocked, and in cases of locked devices, it will depend on the Android OS and stock version installed on the user's device, see Table I.

Fingerprint attack. This method uses the fingerprint attack [1] to retrieve a user's device information. The information retrieved by this technique can lead to obtaining a unique identifier for each particular device[10]. The attacker can then identify and track users deploying several NFC-tags. There are extensive work and real attacks that use the 'cookie-less' methods to track and collect users' information [10], [11]. The fingerprinting techniques collect as much information as possible from a device (e.g., OS version, screen size, OS language, keyboard layout), so the combination of all collected parameters can identify a unique device. The individual identification of a device and, together with the previous location attack, provide a new source of information that connects unique users and their location patterns. This

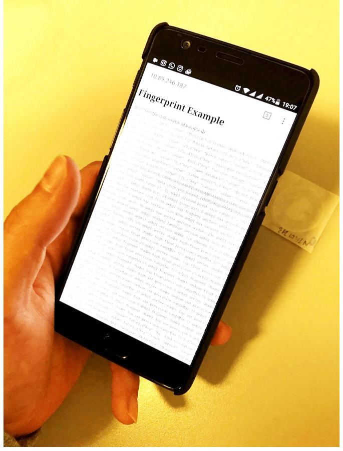


Fig. 5. Device fingerprinting using the NFC tag.

location information attack can be precious as the browsers now request users action to retrieve location. These three mentioned attacks can be combined to obtain more powerful attacks such as fingerprinting and location attacks.

Table II describes the success of the all mentioned attacks according to the device status (locked/unlocked). When the device is being locked, the reading of the NFC tag (unsuccessful cases) triggers a notification message. In the case of locked devices success, the device prompt automatically and without user consent, the website written in the NFC tag. When users' devices are unlocked, the attack is successful for OS version lower than 7.0. As we can see, the OS with a version lower than 7.0 and MIUI stock versions are affected by our threat model. Although the stock versions might affect the performance of our attack, the OS version below 7.0 includes 31.4% Android users according to to [12] that can be affected by our threat model.

Furthermore, Android devices include automatic intends that can be embedded in NFC-tags. These intents, once a device reads the tag, will open the respective application, usually without user interaction [8]. We can use some user's default application permission to access via browsers (i.e., Chrome, Native, Firefox) device's microphone, camera, or in cases of documented and not patched security threads: the device's file system. For example, an NFC-tag can be used to activate the Bluetooth or WiFi module, so more sophisticated attacks can be used. Other example is creation of a fake contact in users' contacts[11]. Not every Android user is experienced with technical details, application permissions, and not every Android device is updated with the latest version.

A. Countermeasures

We propose possible solutions to avoid these harmful situations in mobile NFC-enabled environments. One solution to avoid privacy leakage is in cases where the NFC module reads an NFC-tag. It requests the user's permission to access the link, see Table III. Similar to the current permission

[8] https://github.com/solrac1986/nfc_attacks/blob/master/location.md

[9] https://github.com/solrac1986/nfc_attacks/blob/master/location_js.md

[10] https://github.com/solrac1986/nfc_attacks/blob/master/fingerprint.md

[11] https://github.com/solrac1986/nfc_attacks/blob/master/intent.md

TABLE III
PERMISSION REQUEST EVALUATION

Device	Android OS	Stock Version	Permission request
One Plus 3T	Android 7.1.1	Stock	Valid
Xiaomi Mi3W	Android 5.1	MIUI 7	Valid
Xiaomi Mi3W	Android 6.0.1	MIUI 8	Valid
Samsung C7	Android 6.0.1	Touchwiz	Valid

request approaches in modern mobile OS, such as the location permission notification in mobile applications example, our approach generates a notification message that requires user interaction before opening the default intended application (e.g., browser for URL links). Another solution can work analogously as with BLE Beacons. Once the device reads an NFC-tag, it creates a notification in the notification bar, so the user can access the address when she desires (no work-flow interruption).

V. DISCUSSION

Naturally, there is room for further work and improvements. We discuss a few points here.

Threat model: Our proposed model focus on using the URL/URI written in NFC tags to trigger actions in users' browsers. The lack of users permission request in many Android OS versions increases the chances of our attacks to be unnoticed by users. The combination of location and fingerprinting attacks offers an interesting channel to identify individuals and locate them at the same time uniquely. The three proposed attacks use simple NFC-tags that can be installed in any physical place and can trigger unlocked or locked device actions. In the case of the social network attacks, the assumption is that users are already logged in the online social networks, so there is no login request once the NFC-tag triggers the URL action.

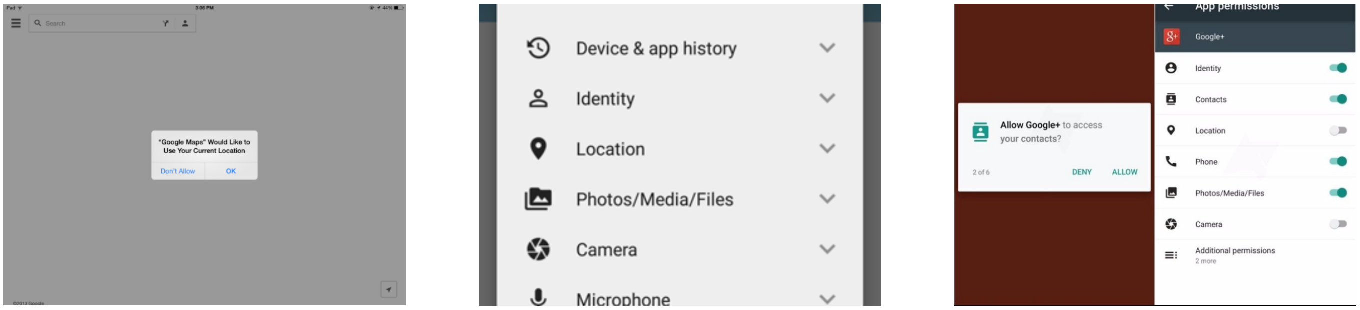
Countermeasures: We propose a simple and efficient solution to avoid the threat of the proposed NFC-based attacks. This solution can also inform users about the privacy risks of using NFC always on modules. In cases of non-disruptive permission requests, we propose a notification message in the system that does not interrupt users' interactions with the smartphones but pauses the triggered action by the NFC-tag until users accepted in the notification bar. The wide use of wearable devices[12] can offer novel security measures to open NFC connections. For example, the use of a smartwatch in conjunction with a smartphone for double authentication can potentially solve the problem with unwanted NFC connections and the data collection threats. Another option is the use of the accelerometers and gyroscopes of a smartphone to detect intended (by users) NFC connections. Due to the position and necessary closeness of NFC-tags, the embedded sensors can detect when the smartphone is moving to read an NFC-tag.

Deployability of the attacks: The current extended use of contactless transaction at supermarkets or public transport together with the increasing use of services such as Apple Pay open more possibilities for these NFC-based attacks. The placement of malicious NFC-tags in public transport services can be unnoticed by users and do not interfere with the transaction process of entering a metro station or bus via contactless transactions [13].

VI. RELATED WORK

The NFC communication protocol is vulnerable to several threads [14]: *eavesdropping*, it is a key thread of wireless communications, as the data transmitted via NFC channel can be intercepted or received by an attacker; *data corruption*, the data transmitted can be modified (corrupted) by an attacker [15]. Denial-of-service (DoS) attacks, and NFC-tag overwriting can be considered within this thread group; *data modification*; *data insertion*, during the exchange of messages data can be inserted; *man-in-the-middle attacks*. Mulliner et al. [6] identify several vulnerabilities and threads on NFC-enabled mobile phones: *mobile telephony service attacks*, using URI spoofing the attacker can leverage malicious SMS, telephone receiver number; *URI/URL spoofing*, URL spoofing enabled via NFC communication protocol [8]. In [16], the authors contribute with the enumeration of more possible NFC attacks: *relay attack*, request confidential information from the secure element or extend the limited (5cm approx.) range of NFC devices to open new threats such as contactless payment at stores [5] (more information Section II); *phising attack*, NFC tag that will execute commands in the user's device(i.e., send email, WiFi AP connection setup) [9]; *ticket cloning*, related to the copy of e-tickets. Eun et al. [17] propose conditional privacy protections against impersonation attacks from NFC eavesdropping (man-in-the-middle) using the user's public key schemes, pseudonyms and addition trusted the third party to protect the privacy of users. Mobile device's fingerprinting is the information collected via web-based methods (i.e., JavaScript, browser-plugins, cookies), sensor-based (i.e., accelerometer, GPS, WiFi) [11]. Hupperich et al. [18] propose a system with modern web-based fingerprinting techniques for mobile devices. The authors discover that some features used in desktop environments lose their importance in mobile fingerprinting environments. They also test the proposed system against evasion attacks such as the changeability of features (i.e., use of a second browser, proxy). The use of fingerprinting methods to attack an NFC enabled device is an innovative approach that related literature has not been tackled yet. The possibilities of locating users via NFC using either fingerprinting or location inherit parameters in the URL opens new channels to identify and track users' locations without their awareness. NFC payment system has become a widespread channel to realize contactless transactions (e.g., Apple Pay, Google Pay). The lack of standard implementation in these transactions opens privacy threats for users. In [13], authors show that a mobile application can retrieve transaction information such as amount and date. As a countermeasure to

[12] https://www.statista.com/statistics/487291/global-connected-wearable-devices/



“Google Maps” Would Like to Use Your Current Location

Allow Google+ to access your contacts?

(a) iOS app permission requests when application request access to personal information (e.g., location).

(b) Android app permission before Marshmallow (Android 6.0). The app permission are requested before installing the app.

(c) Android app permission settings since Marshmallow (Android 6.0). The user can change the app permission in settings menu.

Fig. 6. Application permission information.

these NFC-based attacks, we can use the embedded sensors such as ambient light to avoid unwanted NFC connections (while users' devices are in their pockets) [19] or avoid relay attacks [20].

VII. Conclusion

In this paper, we depict the current state of the mobile NFC-enabled ecosystem's vulnerabilities and threads. Some previous work attacks still can be enabled in current NFC communications and the situation of application permission on Android devices. The latter is still a challenging scenario due to the differences between OS versions, how the OS approaches the permission requests (before app installation), and the no-technical experience Android users. Furthermore, location-based tracking attempts have been solved by browser location-request notifications. However, NFC-tags provide a non-user and straightforward intervention channel to track the user's location, fingerprinting, and other logged-based web site attacks. In summary, our proposed attack enables current mobile fingerprinting techniques using the NFC-tags and URL/URI Android mobile device's vulnerabilities. To conclude, we propose simple deployable solutions that will not interrupt the user-workflow to enable a secure and privacy-aware NFC-Android ecosystem.

Acknowledgments

This research has been supported in part by project 16214817 from the Research Grants Council of Hong Kong, project FP805 from HKUST, the 5GEAR project and the FIT project from the Academy of Finland.

References

[1] N. Nikiforakis, A. Kapravelos, W. Joosen, C. Kruegel, F. Piessens, and G. Vigna, “Cookieless monster: Exploring the ecosystem of web-based device fingerprinting,” in *Security and privacy (SP), 2013 IEEE symposium on*. IEEE, 2013, pp. 541–555.

[2] D. Nelson, M. Qiao, and A. Carpenter, “Security of the near field communication protocol: an overview,” *Journal of Computing Sciences in Colleges*, vol. 29, no. 2, pp. 94–104, 2013.

[3] Android, “NFC basics,” https://developer.android.com/guide/topics/connectivity/nfc/nfc.html, 2019, [Online; accessed 16-January-2020].

[4] D. Giese, K. Liu, M. Sun, T. Syed, and L. Zhang, “Security analysis of near-field communication (nfc) payments,” *arXiv preprint arXiv:1904.10623*, 2019.

[5] Y. Sun, S. Kumar, S. He, J. Chen, and Z. Shi, “You foot the bill! attacking nfc with passive relays,” *arXiv preprint arXiv:2001.08143*, 2020.

[6] C. Mulliner, “Vulnerability analysis and attacks on nfc-enabled mobile phones,” in *Availability, Reliability and Security, 2009. ARES'09. International Conference on*. IEEE, 2009, pp. 695–700.

[7] Android, “Android Security BulletinOctober 2019,” https://source.android.com/security/bulletin/2019-10-01, 2019, [Online; accessed 16-January-2020].

[8] S. Maruyama, S. Wakabayashi, and T. Mori, “Trojan of things: Embedding malicious nfc tags into common objects,” *arXiv preprint arXiv:1702.07124*, 2017.

[9] K. Gold, S. Shetty, and T. Rogers, “A testbed for modeling and detecting attacks on nfc enabled mobile devices,” in *MILCOM 2015-2015 IEEE Military Communications Conference*. IEEE, 2015, pp. 635–640.

[10] A. Kurtz, H. Gascon, T. Becker, K. Rieck, and F. Freiling, “Fingerprinting mobile devices using personalized configurations,” *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 1, pp. 4–19, 2016.

[11] T.-F. Yen, Y. Xie, F. Yu, R. P. Yu, and M. Abadi, “Host fingerprinting and tracking on the web: Privacy and security implications.” in *NDSS*, 2012.

[12] Android, “Distribution dashboard,” https://developer.android.com/about/dashboards, 2019, [Online; accessed 16-January-2020].

[13] M. Mehrnezhad *et al.*, “Nfc payment spy: a privacy attack on contactless payments,” in *International Conference on Research in Security Standardisation*. Springer, 2016, pp. 92–111.

[14] N. A. Chattha, “Nfcvulnerabilities and defense,” in *Information Assurance and Cyber Security (CIACS), 2014 Conference on*. IEEE, 2014, pp. 35–38.

[15] E. Haselsteiner and K. Breitfuß, “Security in near field communication (nfc),” in *Workshop on RFID security*. sn, 2006, pp. 12–14.

[16] C. H. Chen, I. C. Lin, and C. C. Yang, “Nfc attacks analysis and survey,” in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2014 Eighth International Conference on*. IEEE, 2014, pp. 458–462.

[17] H. Eun, H. Lee, and H. Oh, “Conditional privacy preserving security protocol for nfc applications,” *IEEE Transactions on Consumer Electronics*, vol. 59, no. 1, pp. 153–160, 2013.

[18] T. Hupperich, D. Maiorca, M. Kührer, T. Holz, and G. Giacinto, “On the robustness of mobile device fingerprinting: Can mobile users escape modern web-tracking mechanisms?” in *Proceedings of the 31st Annual Computer Security Applications Conference*. ACM, 2015, pp. 191–200.

[19] T. Halevi, D. Ma, N. Saxena, and T. Xiang, “Secure proximity detection for nfc devices based on ambient sensor data,” in *European Symposium on Research in Computer Security*. Springer, 2012, pp. 379–396.

[20] I. Gurulian, C. Shepherd, E. Frank, K. Markantonakis, R. N. Akram, and K. Mayes, “On the effectiveness of ambient sensing for detecting nfc relay attacks,” in *2017 IEEE Trustcom/BigDataSE/ICESS*. IEEE, 2017, pp. 41–49.