Mind your (R, Φ)s: Location-Based Privacy Controls for Consumer Drones

Tavish Vaidya Micah Sherr
{tavish, msherr}@cs.georgetown.edu

Georgetown University, Washington, DC USA

Abstract. This position paper explores the threat to individual privacy due to the widespread use of consumer drones. Present day consumer drones are equipped with sensors such as cameras and microphones, and their types and numbers can be well expected to increase in future. Drone operators have absolute control on where the drones fly and what the on-board sensors record with no options for bystanders to protect their privacy. This position paper proposes a policy language that allows homeowners, businesses, governments, and privacy-conscious individuals to specify location access-control for drones, and discusses how these policy-based controls might be realized in practice. This position paper also explores the potential future problem of managing consumer drone traffic that is likely to emerge with increasing use of consumer drones for various tasks. It proposes a privacy preserving traffic management protocol for directing drones towards their respective destinations without requiring drones to reveal their destinations.

1 Introduction

The proliferation of consumer drones raises a number of important security and privacy questions. Citizens may rightfully ask "What/Who will govern the movement of drones and prevent them from crashing into each other? Will there be a constant hoard of drones hovering over people's backyards? Will there be a drone police to monitor the drones?" Drone operators may pose a different set of questions: "Is my drone allowed to fly over my not-so-friendly neighbor's yard? Am I allowed to operate a drone in a particular vicinity? How do I become informed about flight restrictions? Are there restrictions on flight parameters such as speed and altitude? When am I allowed to record audio or video? To what legal risks am I being exposed by operating a drone?"

Despite the increasingly widespread use of drones, there is surprisingly little existing work that examines technical means by which ordinary citizens may protect their privacy from intrusive drone use, nor is there adequate existing literature that presents technical approaches that drone operators may apply to prevent accidental privacy (or legal) violations due to drone use. Also, with the increasing popularity and usage of consumer drones, the problem of managing drone traffic in areas of allowed drone operations will emerge.

In this position paper, we propose a policy-based access control language for specifying rules governing the operation of drones in a geographical area. We then describe how the access control mechanism might be realized in practice with current consumer drones and provide a discussion on enforcement challenges with respect to adherence to the specified policies. Further, we describe a protocol for managing drone traffic that can be realized by modifying the proposed access control mechanisms.

Background. Currently available consumer drones offer various capabilities with respect to onboard sensors and navigational control. Most consumer drones are now equipped with multiple cameras, GPS, accelerometer and other sensors to ensure smooth operation and various flight capabilities. The majority of drones provide a real-time video feed using an onboard camera; this feed allows navigation when the drone is far or not in view of the operator. Some more recent drones with onboard GPS can fly autonomously once their flight path is marked with GPS coordinates [2, 7]. Other drones can autonomously follow a paired device

having a GPS antenna [4, 5]. We believe that the set of onboard sensors will increase in future consumer drones, much as has been the case with smartphones and other mobile devices. These new highly equipped drones will enable more functionalities, which in turn pose addition privacy and security risks.

2 Privacy and Security Challenges of Widespread Use of Drones

Present day consumer drones pose several privacy and security challenges:

Threat to privacy. Consumer drones are equipped with high resolution cameras that can continuously record and relay live video streams to their operators. Microphones can easily be placed on these drones to record audio. Although such sensors provide diverse functionalities, misuse of drones can have serious consequences to privacy. Widespread use of drones will also raise issues concerning illegal surveillance and stalking. A prime motivation of this position paper is that such privacy concerns should be carefully addressed *before* drones become ubiquitous.

Security and safety issues. Operating in the physical world, the drones themselves pose safety hazards. Many incidents have been reported by commercial airline pilots with drones coming in close proximity of aircraft during landing and takeoff [3]. Some consumer drone manufacturers have made efforts to prevent such incidents by hardcoding the locations of major airports onto the drones and adding controls that prevent their use in the vicinity of these locations [6]. However, such a solution clearly addresses only a specific issue and does not scale. Additionally, such protections are brittle: we tested one of the consumer drones available in the market and were able to take control from the operator by spoofing the control packet stream. More generally, the lack of device authentication and other security mechanisms allow an attacker to hijack drones and potentially cause physical damage to infrastructure and injury to persons. We argue that there is a strong need for a standard policy language and secure enforcement mechanism for limiting the flight paths of drones.

Threat model and goals: Our threat model considers the adversary to be the users of consumer drones who, with unrestricted use of drones and their capabilities, can pose a threat to security, privacy and the safety of other individuals and cause damage to property. Even in the absence of any malicious intent, consumer drones can inadvertently invade personal privacy or operate in restricted areas due to lack of any active guiding/restricting signals that can inform the drones (and their operators) of any such restrictions.

Ideally, all drones will have tamper-resistant components that enforce authenticated policies and cannot be overridden. Unfortunately, such a scheme is likely impossible to achieve in reality. However, the current status quo—i.e., a total lack of security and privacy protections—is also clearly undesirable. In this position paper, we consider drones produced by law-abiding manufacturers (as opposed to home-built drones that can clearly be constructed to ignore advertised policies). Our claim is that by developing *and widely adopting* flexible standards for expressing drone restrictions, meaningful privacy protections can be achieved that benefit both the general public and drone operators.

3 Policy-Based Location Access Control

This position paper proposes a policy-based location access control mechanism for consumer drones. Our approach requires sub-mechanisms for (i) specifying policy-based restrictions, (ii) communicating these restrictions to the drone and (iii) enforcing these restrictions on the drones. For specifying policy-based restrictions, we propose a policy language which can be used by a privacy conscious individual to specify

restrictions on the usage of drones and/or their allowed capabilities near a particular location (e.g., his house).

Figure 1 motivates our proposed system model. Here, a homeowner specifies a policy as to whether a drone can operate over his property, and if so, what restrictions are in place. To be useful, the policy language clearly needs to define both geographic boundaries and be sufficiently flexible to support authorizations over a large set of actions (see below). Once the user has written a policy, he configures his wireless access point (WAP) to periodically broadcast it over WiFi. WiFi-equipped drones listen for such policy "beacons" and verify that their current state and their planned actions do not violate the advertised policy. As we discuss in more detail below, drones that lack GPS-capabilities (or more generally, the ability to accurately and precisely determine their current locations) can use the signal strength of the beacons to estimate its distance to WAP.

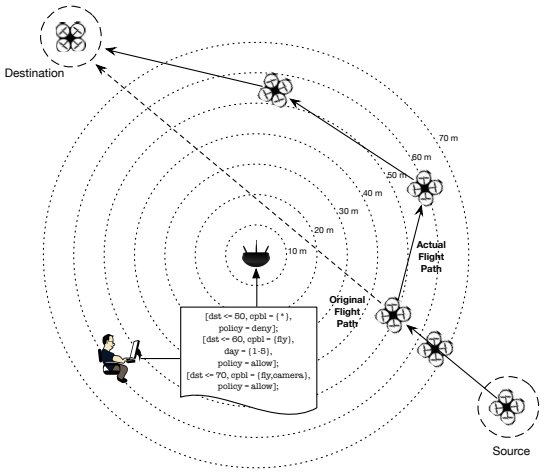

Fig. 1: System Model: A homeowner specifies a location access policy for various capabilities, distances and day of the week. His WAP broadcasts the policies for that particular day and time as beacons. The dashed line indicates the original intended flight path; the solid line denotes the adjusted path that conforms to the homeowner policy.

Policy language. We describe a simple policy language for specifying location access rules as a context-free grammar $L = (V, \Sigma, R, S)$. The rules for L are provided in Figure 2.

Grammar L generates strings that define location based access policies for various capabilities. Once the user specifies the values to be assigned to the variables, the generated policy strings will be translated into rules before being broadcasted.

Examples. Let us consider the following example with policies specified in decreasing order of priority from top-to-bottom:

(a) $0900 - 1400 : Mon; Tue; Wed; Thur$: ϵ : Sphere : $(10, 10, 10)$: ϵ : Allow
(b) ϵ : $Sat; Sun$: Fly : Cylinder : $(10, 12, 13)$: 60 : Allow
(c) ϵ : ϵ : ϵ : Sphere : $(20, 10, 15)$: 0 : Deny

Policy a allows all capabilities between 9 am to 2 pm, Monday to Thursday at any distance greater than 10 meters from the beacon without any limit on noise levels. Policy b only allows drones to fly at a distance of 10 m and 12 m high from the beacon while implicitly restricting other capabilities on Saturdays and Sundays and also places a limit of 60 decibels on the drone's noise level. Policy c denies any use of drones and their capabilities within 15 m of the beacon throughout the day for every day of the week and has a zero noise tolerance policy. Use of drones nearby airports can be restricted by using policy c. Policy c also acts as the default policy in this case, restricting the use of drone outside of the time interval specified by policy a and on Fridays.

4 Towards a Practical Realization

To communicate with drones, beacons broadcast the specified policy for that particular day and time. Any interface that can successfully communicate with the drones can be used as a beacon, with 802.11 WiFi being an obvious choice.

- V is the finite set of variables representing different parameters to be specified in location access rules.
  $V = \{S, \text{Day}, \text{Time\_Interval}, \text{Capability}, \text{Region}, \text{Coordinate}, \text{Noise\_Limit}, \text{Policy}\}$
- Σ is the finite set of values the variables can take, such that:

$$\begin{aligned}\Sigma = \{&d | d \text{ is a day}\} \cup \\ &\{hhmm - HHMM | hhmm, HHMM \text{ are valid times in 24-hr format and } hhmm < HHMM\ \} \cup \\ &\{\text{Hover, Fly, Camera, Microphone, IR-Imaging}\} \cup \\ &\{\text{Sphere, Cylinder, 3D-Polygon}\} \cup \\ &\{(x, y, z) | (x, y) \text{ are GPS coordinates and } z \text{ is the height from ground}\} \cup \\ &\{k | k \text{ is the noise limit in decibels}\} \cup \\ &\{\text{Allow, Deny}\} \cup \\ &\{\epsilon\} \cup \{:, ;\}\end{aligned}$$

- R is the finite set of rules for governing the generation of semantically valid access rules. The set of rules R is:
  - $S \rightarrow \text{Time\_Interval} : \text{Day} : \text{Capability} : \text{Region} : \text{Coordinate} : \text{Noise\_Limit} : \text{Policy}$
  - $\text{Time\_Interval} \rightarrow hhmm - HHMM; \text{Time\_Interval} | hhmm - HHMM | \epsilon$
  - $\text{Day} \rightarrow d | d;\ \text{Day} | \epsilon$ (where d is a day of week)
  - $\text{Capability} \rightarrow$ Hover | Fly | Camera | Microphone | IR-Imaging | ϵ
  - $\text{Region} \rightarrow$ Sphere | Cylinder | 3D-Polygon
  - $\text{Coordinate} \rightarrow (x, y, z); \text{Coordinate} \mid \epsilon$
  - $\text{Noise\_Limit} \rightarrow k \mid \epsilon$
  - $\text{Policy} \rightarrow$ Allow | Deny
- $S \in V$ is the start variable.

Fig. 2: A CFG for a location access control policy language for drones. We remark that our proposed policy language is designed to be extensible. New capabilities of future drones can be easily incorporated by adding a new value to set Σ.

We tested the feasibility of communicating with a drone from its operating environment by using a bottom of the line Parrot AR 2.0 consumer drone and performing a toy experiment. The drone is equipped with an 802.11 wireless interface which acts a WiFi hotspot to which any 802.11 supporting client device can connect. In our toy experiment, we connected two devices: the first was the operator that controlled the drone and the second acted as the beacon. The beacon sent ping broadcasts which were received by the operator via the drone. This toy experiment shows that any device supporting 802.11 can be used as a beacon to relay information to the drone or the drone's controller. Relaxing the security constraint, the drone can estimate the distance from the beacon using the observed signal strength and compare it with the distance specified in the access rule broadcast [8, 9, 14]. In a practical setting, a beacon can keep trying to connect to a drone. When a drone is in range, it can then broadcast the specified rules thereby relaying access control information to the drone.

Higher-end drones are GPS enabled and can therefore better estimate their positions. The beacons can also broadcast GPS coordinates specifying a space perimeter in the access control rules and drone can learn the restrictions on its operation in that space perimeter. Future drones equipped with other sensors (e.g., proximity sensors) can gather data from multiple sources to get more accurate estimates of their position in space with respect to the beacon and the policy specified region in space.

5 Enforcement?

An obvious challenge — and one that we do not claim to completely solve — is to *enforce* the access control policies.

One potential approach is to rely on tamper-resistant hardware [22]. Our proposed approach requires the drones to act on the access control information relayed by the beacon. To enforce the restrictions on the drone, we propose a tamper-resistant "enforcer" module that manufactures could incorporate into the design of their drones. The enforcer would intercept data from drones' sensors and would act as a security layer, similar to SELinux's Linux Security Module architecture for regulating the use of system calls. The enforcer decides which sensor is allowed to operate in the current environment based on the received access control rules (Figure 3). With respect to movement, the enforcer can directly send navigation commands to the actuators to ensure compliance to broadcasted restrictions or slowly land the drone as a failsafe mechanism.

We do not yet know how to build a policy enforcement mechanism when the adversary can build their own drones with custom hardware and software. We likely will never be able to stop such an adversary from posing a threat to others' privacy and safety.

Despite this important shortcoming, we argue that there are clear benefits for standardizing an access control language and enforcement mechanism: First, including enforcement mechanisms on commodity drones raises the bar for violating access control policies by requiring drone operators to either build their own hardware or defeat the tamper-evident features of commodity drones. Second, and most importantly, including an automated enforcement system *aids* well-intentioned drone operators by ensuring that they do not mistakenly violate federal, state, or local laws or ordinances concerning the use of drones. For example, the U.S. Federal Aviation Administration (FAA) bans all use of drones in Washington, DC [1]; it is unlikely that these rules were intended for hobbyist drone operators located miles away from federal buildings, and indeed such restrictions are not widely publicized. Including integrated policy enforcement mechanisms protects drone operators from unintentionally violating others' privacy and helps ensure compliance with legal restrictions.

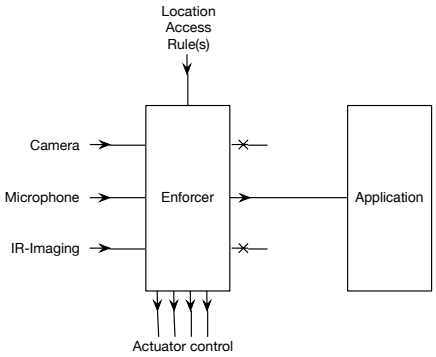

Fig. 3: Enforcer layer between sensors and application layer. Based on the location access rule(s), only allowed sensor data is passed to the application.

6 Privacy Preserving Traffic Management for Consumer Drones

Autonomous drones are not a distant reality and will be seen flying in neighborhoods doing various tasks. As the number of drones operating in an area increases, the lack of coordination on flight parameters among the drones may lead to collisions causing physical harm and other undesirable consequences. (Consider, for example, the perils of ten drones simultaneously approaching an intersection without any coordination.) A potential solution for coordinating movements is to have each drone communicate with every other drone in its vicinity. Such solution will clearly scale poorly as the number of drones increase.

A better approach for managing drone traffic at regions of contention is to have a central coordinator, i.e., a drone *traffic controller*, that regulates the movement of all drones passing through that region and directs them to towards their respective destinations. Such drone traffic controllers are feasible by extending the use of beacons for broadcasting navigational directions instead of (or in addition to) access policies.

Additional assumptions. We assume that drones can securely communicate with the beacon, e.g., via TLS. We note that this may require beacons to obtain a verifiable certificate, e.g., from LetsEncrypt[1]. We further

[1] https://letsencrypt.org/

assume that the beacon is connected to the Internet and can provide a communication link between drones and the Internet. This latter assumption allows us to offload computations to the cloud on behalf of the drone.

A non-privacy-preserving protocol. Beacons act as drone traffic controllers for contentious regions and route incoming drones towards their respective destinations. The controllers require destination information for each incoming drone to make scheduling and routing decisions. Each drone approaching the contentious region broadcasts its destination information encrypted using the beacon's public key. Beacons maintain state information about traffic—for example, the number of drones previously directed in each direction.

For regulating traffic, beacons compute a routing function that takes into account the state information and the destination of the drone. The routing function is computed for each incoming drone request and outputs the flight parameters to be passed on to the requesting drone. The scheduling of drones can be controlled by injecting time delays in broadcasting the response to each drones' request for directions.

A privacy-preserving traffic management protocol. The above protocol allows for managing drone traffic using beacons. However, it requires the drones to reveal their final destinations to beacons. This may be problematic for future uses of drones. (Consider, for example, package delivery systems in which purchasers may not want their locations broadcast during the delivery process.)

We propose the use of secure-multiparty computation (SMC) for computing the routing function, such that drones never reveal their final destinations to the beacons. SMC is of course resource intensive and drones have limited computational resources. We remark, however, that Carter et al.'s recent work on off-loading resource intensive steps of SMC to the cloud [11] can be directly applied in our drone setting.

A drone is able to communicate with the cloud infrastructure via communication relayed through the beacon. We note that this link can be secured using end-to-end encryption between the drone and cloud, e.g., using TLS. A drone, with the help of cloud, and the beacon perform a two-party SMC for calculating the routing function without revealing their inputs. The Whitewash protocol [11] allows for off-loading the garbled circuit generation for the routing function on behalf of the drone to the cloud, while the beacon evaluates the output of the circuit. The inputs to the routing function, the state information and drone's destination, are garbled and supplied by the beacon and the drone respectively to the cloud. Thus, no party learns the original inputs of any other party. After evaluating the garbled circuit for the routing function, the beacon releases the outputs of the function to the drone. The outputs of the routing function will be the flight parameters to be sent to the drone. The beacon will distribute the results of protocol runs to each drone, encrypted with the ephemeral key provided by the respective drone, at pre-scheduled intervals to prevent collisions.

The proposed protocol leverages the cloud infrastructure to benefit resource-constrained consumer drones, allowing the dissemination of flight parameters without revealing drones' final destinations.

7 Related Work

Mechanisms for detecting malicious applications and limiting their access to various functionalities and sensitive data have been thoroughly researched [10, 12, 15, 18, 25], with the primary aim of protecting the privacy of device owners from malicious applications. Other works on permissions and privileges in operating systems have focused on providing minimum privileges to applications by default and granting access to sensitive data only when required to protect the user [21, 23, 24]. On the contrary, our work focuses of protecting the privacy of *other individuals* from the drone operators.

Jana *et al.* [16] proposed fine-grained access permissions for augmented reality application to protect user privacy by allowing the user to specify permissions at the object level. However, the user who wants to protect his privacy himself specifies the permissions, unlike the case with the operation of drones.

Our work is closely related to the work done by Roesner *et al.* [22] that proposed the use of external indicators to convey permission information of allowed capabilities to wearable devices, freeing the device user from the burden of managing permissions in changing environments. Our work builds off of a similar idea to allow individuals to specify location access control rules that restrict the use of drones in a specified geographical region.

Policy languages have been proposed for specifying security policies with complex constraints in different applications such as firewalls, file permissions, etc. [13, 19, 20, 26]. Of particular note, GeoXACML is an extension to eXtensible Access Control Markup Language (XACML) that allows the declaration of location-specific access rights [17]. These policy languages are too general and verbose to be well-suited for specifying simple policies for the drones. Our proposed language is purposefully compact and is specifically tailored for users of consumer drones with ease-of-use in mind.

8 Conclusion

This position paper discusses the potential issues and questions that will be raised as consumer drones become more commonplace. We argue that the community's emphasis should be on finding solutions to privacy problems *before* drones become ubiquitous. Towards this end, we propose a policy-based location access control mechanism to counter the privacy and safety threats posed by consumer drones. We believe that further discussion and research is required to realize the balanced use of drones and their capabilities.

We also address the issue of managing consumer drone traffic, set to arise with ubiquitous use of drones in the future. We propose the use of drone traffic controllers, and sketch a privacy preserving protocol for directing drone traffic without requiring drones to reveal their destinations.

Acknowledgments

This work is partially supported by the National Science Foundation through grants CNS-1064986, CNS-1149832, CNS-1223825 and CNS-1445967. The views expressed are those of the authors and do not reflect the official policy or position of the National Science Foundation.

Bibliography

[1] NOTAM Number : FDC 0/8326. http://tfr.faa.gov/save_pages/detail_0_8326.html. Last Accessed January 4, 2015.

[2] DJI Phantom 2. http://www.dji.com/product/phantom-2/feature, 2014. Last Accessed January 3, 2015.

[3] Near-collisions between drones, airliners surge, new FAA reports show. http://www.washingtonpost.com/world/national-security/near-collisions-between-drones-airliners-surge-new-faa-reports-show/2014/11/26/9a8c1716-758c-11e4-bd1b-03009bd3e984_story.html, 2014. Last Accessed January 3, 2015.

[4] HEXO+. http://hexoplus.com, 2014. Last Accessed January 3, 2015.

[5] IRIS+. https://store.3drobotics.com/products/iris, 2014. Last Accessed January 3, 2015.

[6] No FLY Zones. http://www.dji.com/fly-safe/category-mc, 2014. Last Accessed January 3, 2015.

[7] Parrot Bebop Drone. http://www.parrot.com/usa/products/bebop-drone, 2014. Last Accessed January 3, 2015.

[8] J. Blumenthal, F. Reichenbach, and D. Timmermann. Minimal transmission power vs. signal strength as distance estimation for localization in wireless sensor networks. In *Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society on*, volume 3, pages 761–766, Sept 2006.

[9] N. Bulusu, J. Heidemann, and D. Estrin. Gps-less low-cost outdoor localization for very small devices. *Personal Communications, IEEE*, 7 (5):28–34, Oct 2000. ISSN 1070-9916.

[10] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani. Crowdroid: Behavior-based malware detection system for android. In *Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*, SPSM '11, pages 15–26, New York, NY, USA, 2011.

[11] H. Carter, C. Lever, and P. Traynor. Whitewash: Outsourcing garbled circuit generation for mobile devices. 2014.

[12] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth. Taintdroid: An information-flow tracking system for realtime privacy monitoring on smartphones. In *Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation*, OSDI'10, pages 1–6, Berkeley, CA, USA, 2010.

[13] I. Fundulaki and M. Marx. Specifying access control policies for xml documents with xpath. In *Proceedings of the Ninth ACM Symposium on Access Control Models and Technologies*, SACMAT '04, pages 61–69, New York, NY, USA, 2004.

[14] D. Han, D. Andersen, M. Kaminsky, K. Papagiannaki, and S. Seshan. Access point localization using local signal strength gradient. In *Passive and Active Network Measurement*, volume 5448 of *Lecture Notes in Computer Science*, pages 99–108. 2009.

[15] P. Hornyack, S. Han, J. Jung, S. Schechter, and D. Wetherall. These aren't the droids you're looking for: Retrofitting android to protect data from imperious applications. In *Proceedings of the 18th ACM Conference on Computer and Communications Security*, CCS '11, pages 639–652, New York, NY, USA, 2011. ACM.

[16] S. Jana, D. Molnar, A. Moshchuk, A. Dunn, B. Livshits, H. J. Wang, and E. Ofek. Enabling fine-grained permissions for augmented reality applications with recognizers. In *Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13)*, pages 415–430, Washington, D.C., 2013.

[17] A. Matheus and J. Herrmann. Geospatial extensible access control markup language (geoxacml). *Open Geospatial Consortium Inc. OGC*, 2008.

[18] M. Nauman, S. Khan, and X. Zhang. Apex: Extending android permission model and enforcement with user-defined runtime constraints. In *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, ASIACCS '10, pages 328–332, New York, NY, USA, 2010.

[19] Q. Ni, S. Xu, E. Bertino, R. Sandhu, and W. Han. An access control language for a general provenance model. In *Proceedings of the 6th VLDB Workshop on Secure Data Management*, SDM '09, pages 68–88, Berlin, Heidelberg, 2009.

[20] C. Ribeiro, C. Ribeiro, A. Zúquete, P. Ferreira, and P. Guedes. Spl: An access control language for security policies with complex constraints. *IN PROCEEDINGS OF THE NETWORK AND DISTRIBUTED SYSTEM SECURITY SYMPOSIUM*, pages 89–107, 1999.

[21] F. Roesner, T. Kohno, A. Moshchuk, B. Parno, H. Wang, and C. Cowan. User-driven access control: Rethinking permission granting in modern operating systems. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 224–238, May 2012.

[22] F. Roesner, D. Molnar, A. Moshchuk, T. Kohno, and H. J. Wang. World-driven access control for continuous sensing. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 1169–1181, New York, NY, USA, 2014. ACM.

[23] J. Shirley and D. Evans. The user is not the enemy: Fighting malware by tracking user intentions. In *Proceedings of the 2008 Workshop on New Security Paradigms*, NSPW '08, pages 33–45, New York, NY, USA, 2008.

[24] M. Stiegler, A. H. Karp, K.-P. Yee, T. Close, and M. S. Miller. Polaris: Virus-safe computing for windows xp. *Commun. ACM*, 49(9):83–88, Sept. 2006.

[25] R. Xu, H. Saïdi, and R. Anderson. Aurasium: Practical policy enforcement for android applications. In *Proceedings of the 21st USENIX Conference on Security Symposium*, Security'12, pages 27–27, Berkeley, CA, USA, 2012.

[26] B. Zhang, E. Al-Shaer, R. Jagadeesan, J. Riely, and C. Pitcher. Specifications of a high-level conflict-free firewall policy language for multi-domain networks. In *Proceedings of the 12th ACM Symposium on Access Control Models and Technologies*, SACMAT '07, pages 185–194, New York, NY, USA, 2007.