Now Is the Time for a Security and Safety Standard for Consumer Smartphones Controlling Diabetes Devices

Journal of Diabetes Science and Technology
2017, Vol. 11(5) 870–873

DOI: 10.1177/1932296817723259
journals.sagepub.com/home/dst
SAGE

David C. Klonoff, MD, FACP, FRCPE, Fellow AIMBE[1], David Kerr, MD, FRCPE[2], and Dave Kleidermacher, BS[3]

Keywords
cybersecurity, diabetes, mobile phone, safety, security, smartphone, standard

The medical Internet of things is growing exponentially in terms of ideas, technologies, devices, and data. For diabetes care the future is digital and immediate—very soon multiple versions of artificial pancreas systems will be available commercially, supported by cloud computing receiving real-time data from glucose monitoring devices and delivering minute-by-minute changes for insulin delivery systems. In addition smart insulin pens will soon become available supported by decision support systems aimed at calculating safe and effective insulin dose recommendations. In turn people with diabetes are asking that new digital systems for diabetes management should be embedded within their smartphones rather than asking them to carry additional devices around.[1]

A ubiquitous experience for any smartphone user is regular updates of the operating system whether it is iOS or Android based as well as other software including smartphone applications (apps). At the recent Digital Diabetes Congress (https://www.diabetestechnology.org/ddc/),[2] representatives from the US Food and Drug Administration (FDA) highlighted the necessity for any update to not impact patient safety. In other words, making sure that a smartphone-based "command and control" system is not adversely affected by a software update of the phones operating system, which could have catastrophic consequences such as insulin over delivery from a fully automated closed-loop system.

At present the FDA does not permit the use of software running on off-the-shelf consumer smartphones in life-critical medical control contexts (for example, using an app to change the delivered insulin dose in an individual connected to an infusion system) because doing so brings the mobile software and its underlying mobile platform into the domain of the approved medical device regulatory umbrella, and such consumer products are not built according to the existing approval standards for a medical device. Consumer smartphones and apps generally suffer from a lack of assurance across the multistakeholder medical community sufficient for this environment. The current inability to deploy life-critical medical software from mobile platforms prevents a variety of potential clinical and health economic benefits that would otherwise be possible if such software could meet regulatory requirements. It is clear that the burden for individual companies trying to mitigate against the negative consequences of smartphone updates on their medical devices is likely to be substantial for an individual company and could lead to stifling of innovation and a loss of benefit for people with diabetes.

What Is Needed?

Diabetes Technology Society (DTS) intends to create a standard that defines the requirements and an evaluation program for generating assurance that software on commercial off-the-shelf mobile devices is consistently safe and secure for medical control applications. In the diabetes sphere, these could include a remotely controlled insulin infusion device, a multihormonal artificial pancreas, or a remotely controlled glucagon rescue system. Beyond diabetes, this standard could be promulgated across the medical technology industry and achieve recommended and/or mandated status by the FDA and other regulatory agencies. The project to develop the standard and associate evaluation program will be known as the Diabetes Technology Society Mobile Platform Controlling a Diabetes Device Security and Safety Standard (DTMoSt).

The Strategy

A DTMoSt Steering Committee will, a priori, outline the scope of the project to agree on standards for the security and

[1]Mills-Peninsula Medical Center, San Mateo, CA, USA
[2]William Sansum Diabetes Center, Santa Barbara, CA, USA
[3]Google, Mountain View, CA, USA

Corresponding Author:
David C. Klonoff, MD, FACP, FRCPE, Fellow AIMBE, Mills-Peninsula Medical Center, 100 S San Mateo Dr, Rm 5147, San Mateo, CA 94401, USA.
Email: dklonoff@diabetestechnology.org

Table 1. Steering Committee Stakeholder Group Membership.

(1)	Independent cybersecurity and network cybersecurity
(2)	Academic researchers in medical control technology
(3)	Medical device manufacturers
(4)	Mobile device and operating system manufacturers
(5)	Federal regulatory agencies
(6)	Standards development organizations
(7)	Physicians
(8)	Diabetes educators or nurses
(9)	Health care professional organizations
(10)	Regulatory experts from law and mathematics
(11)	Insurance/risk management
(12)	Diabetes community

safety of consumer smartphones controlling diabetes devices. Initial discussions have suggested developing the program similar to the process involved in establishing the Diabetes Technology Society Cybersecurity Standard for Connected Diabetes Devices (DTSec) (https://www.diabetestechnology.org/dtsec.shtml).[3] Early discussions have centered on developing the standard in collaboration with members across the stakeholder community (clinical, academic, technological, regulatory and from the diabetes community) including the FDA. Initial FDA feedback has been extremely positive. Based on the DTSec experience this type of standard development will be required to be managed by a nonprofit (rather by a commercial or government) organization. Subsequently an assurance program will be created and run under the same auspices at first but will be transferred to other FDA-recognized Standards Development Organization(s) as needed.

Steering Committee

The standard should be steered by a multistakeholder community. Membership will include stakeholders from various stakeholders in the area of mobile control of diabetes devices (see Table 1). Furthermore, additional input on selected issues will be sought from advisors who will be from the 12 stakeholder groups.

FAQs About the Proposed Standard

Several questions will need to be addressed by the DTMoSt Steering Committee at the beginning of the standard development process:

(1) Should DTMoSt leverage IEC 62304 (a standard that defines the life cycle requirements for medical device software, including the development and maintenance of medical device software when software is itself a medical device or when software is an embedded or integral part of the final medical device) and other existing standards for the medical software safety?

(2) How much of the hardware platform needs to be evaluated (for security and/or safety), or can we leverage existing standards such as the ISO 15408 mobile device security evaluations for them? For example, Samsung, LG, BlackBerry, Microsoft, and Apple have all been involved in the US government's mobile device platform security evaluation program using ISO 15408.

(3) What level of assured isolation between malware and personal apps/services does medical control software require to be sufficiently "safe" and "secure"? Is hardware-enforced isolation below the main mobile OS required (eg, TrustZone or hypervisor) or is app-level security sufficient (such as app-level containers like BlackBerry Dynamics and EHR/EMR app security)? In reality there is generally a trade-off between platform flexibility and security.

(4) What kind of dynamic attestation/monitoring is required and how is this managed?

(5) What is the impact of battery exhaustion[4] and requirements for user authentication that could impact access to life-critical function when using a mobile platform for control?

(6) Regarding configuration, how "locked down" must a device be to meet requirements? There is a further trade-off between locking down and ease of use.[5] Since locking down may require remote management, which may or may not be practical, then who is responsible for this?

Additional Issues

Several additional issues will be addressed at the beginning of the DTMoSt development process:

(1) *Notional architecture*: Hardware/device + isolation technology + medical apps. Isolation technology can be built into the medical apps themselves or a separate evaluable component (eg, container, TEE, etc).

(2) *Minimum certification for a security/safety standard*: How much safety and security does the hardware and its base OS itself need? NIAP is the National Information Assurance Partnership, which is a United States government organization that oversees evaluations of commercial information technology (IT) products for use in national security systems. NIAP is operated by the National Security Agency (NSA), and was originally a joint effort between NSA and the National Institute of Standards and Technology (NIST). NIAP certification may be a simple base standard to consider for security because it has been met successfully by a large number of mobile device vendors (Samsung, LG, HTC, Apple, Microsoft, BlackBerry). The downside of NIAP certification is that NIAP imposes a significant time and resource

hurdle. An alternative is to include the device itself within scope of the DTMoSt effort.

(3) *A security standard*: We should use the DTSec approach for defining the threat model, corresponding security requirements, and resulting protection profile(s) for the "medical software domain on commercial off-the-shelf (COTS) mobile devices." We believe that management of DTMoST will be assumed by a comanaged alliance of two well-known Standard Development Organizations: For safety, we should use existing medical standards, e.g., IEC 62304 because it is not practical to reinvent safety standards at this stage. These standards should apply at a minimum to medical apps since these will generally be developed by medical companies already well versed in medical safety standards. It is likely impractical to impose medical standards on widely deployed consumer/enterprise COTS hardware and isolation technologies.

(4) *Isolation technology*: Whether built into apps or contained within apps, isolation technology must be included in the scope of the standard. There needs to be assurance that the medical function is properly protected from malware and other threats on a COTS mobile device. However, the standard must be flexible enough to account for the fact that high assurance of protection may be achieved by leveraging a combination of relatively lower assurance components within a mobile device, higher assurance components within the same device (separated via high assurance isolation technology), and/or components/products external to the mobile device. For example, a smartphone-based remote controller app on a consumer smartphone could leverage TrustZone software to verify the user's remote control command or, alternatively, could leverage the connected medical device itself to verify the user's remote control command.

(5) *Protection profiles and/or security targets*: These constructs should probably leverage multiple assurance packages representing multiple levels of assurance.[6-8] One can imagine, for example, three levels of assurance:

   a. Low, AVA_VAN.1-3: eg, COTS device that lacks NIAP with medical apps on top, with isolation technology that cannot meet a higher level of assurance but at least implements the basic required security functions derived from the threat model
   b. Medium, AVA_VAN.4: eg, COTS device with NIAP and isolation technology that is evaluated to protect against moderate attack potential threats.
   c. High, AVA_VAN.5: customized device (eg, customized hardware and/or firmware) that meets NIAP and isolation technology evaluated to protect against high attack potential threats. This could also be used for more narrow use cases on COTS devices that do not require assurance of the entire mobile OS.

   The use of multiple assurance levels recognizes that security/benefit trade-offs will vary widely across the medical community. DTMoSt should not impose policy on what is "good enough" for a particular use case but rather offer a reasonably flexible (whereby too much flexibility is not good either) selection of assurance that policy makers can then apply. For example, policy makers/regulators can leverage various levels of attack potential depending on how much of an effect an attack would have, whether a threat is or is not from a wide-area network attack, whether compensating controls exist in the environment, or whether compensating controls beyond the evaluated configured conformation are even applicable.

(6) *Threats*: An initial list of threats to consider for the medical software domain include network eavesdropping, network-borne attack of the medical domain, physical attack, malicious app, persistent presence of malicious software. Other threats to consider include denial of service (DoS) attacks and privacy threats which are more problematic to subdue because DoS is difficult to mitigate in COTS technology and privacy threats relate to a level of complexity that goes beyond the initial goal of safe and secure medical control.[9]

(7) *Security objectives*: Solutions to counter threats to medical software might include: a secure communications channel to the medical device; secure configuration/management of medical domain; user authentication to the device/domain (which is difficult to build in a medical context); integrity protection of the medical domain; or isolation of the medical domain from the rest of the device (execution domains).

Conclusions

DTMoSt will provide a path for security and safety for mobile platforms controlling a diabetes device. (Note: the standard will not be intended for interoperability, eg, communications standards between smartphones and medical devices). Increasingly mobile platforms are being used to control a variety of devices—but not yet medical devices at this time. Upon completion, DTMoSt will be useful to regulatory officials, the mobile platform industry, and the medical device industry to develop hardware and software solutions to securely and safely control medical devices.

Abbreviations

COTS, commercial off the shelf; DoS, denial of service; DTMoSt, Diabetes Technology Society Mobile Platform Controlling a Diabetes Device Security and Safety Standard; DTS, Diabetes Technology Society; DTSec, Diabetes Technology Society Cybersecurity Standard for Connected Diabetes Devices; FDA, United States Food and Drug Administration; IT, information technology; NIST, National Institute of Standards and Technology; NSA, National Security Agency.

Acknowledgments

The authors thank Annamarie Sucher for her expert editorial assistance.

Declaration of Conflicting Interests

The author(s) declared the following potential conflicts of interest with respect to the research, authorship, and/or publication of this article: DCK is a consultant for Ascensia, Lifecare, Novo Nordisk, Onduo, Trividia, and Voluntis. D Kerr is medical advisor to Glooko, Vicentra, and Novo Nordisk and has received research funding from Samsung, Dexcom, Abbott Diabetes Care and Lilly. D Kleidermacher is an employee of Google.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

1. Blauw H, Keith-Hynes P, Koops R, DeVries JH. A review of safety and design requirements of the artificial pancreas. *Ann Biomed Eng*. 2016;44(11):3158-3172.
2. Klonoff DC, Kerr D. Digital diabetes communication: there's an app for that. *J Diabetes Sci Technol*. 2016;10(5):1003-1005.
3. Klonoff DC, Kleidermacher DN. Now is the time for a cybersecurity standard for connected diabetes devices. *J Diabetes Sci Technol*. 2016;10(3):623-626.
4. Zavitsanou S, Chakrabarty A, Dassau E, Doyle FJ III. Embedded control in wearable medical devices: application to the artificial pancreas. *Processes*. 2016;4(4):35.
5. Ehrler F, Blondon K, Baillon-Bigotte D, Lovis C. Smartphones to access to patient data in hospital settings: authentication solutions for shared devices. *Stud Health Technol Inform*. 2017;237:73-78.
6. Common Criteria for Information Technology Security Evaluation. Part 1: Introduction and general model. Version 3.1. Revision 5. CCMB-2017-04-001. April 2017.
7. Common Criteria for Information Technology Security Evaluation. Part 2: Security functional components. Version 3.1. Revision 5. CCMB-2017-04-002. April 2017.
8. Common Criteria for Information Technology Security Evaluation. Part 3. Security assurance components. Version 3.1. Revision 5. CCMB-2017-04-003. April 2017.
9. Altawy R, Youssef AM. Security tradeoffs in cyber physical systems: a case study survey on implantable medical devices. *IEEE Access*. 2016;4:959-979.