Landpower Essay

No. 12-1
October 2012

An Institute of Land Warfare Publication

Just Don't Take Away My Smartphone

by

Jeremy Rasmussen

Introduction

The Connecting Soldiers with Digital Applications (CSDA) program under the Army Capabilities Integration Center (ARCIC) published a white paper titled "Building the Case for a Bring-Your-Own-Device Solution" in which it outlines the need for fully enabled smart devices because "Digital Age Soldiers and civilians continually seek information and want their information needs gratified immediately."[1] In fact, it says that access to technologically advanced smartphones and tablets, as well as reach-back to the cloud for anytime/anywhere access to information, is vital to professional development and operational knowledge. However, it also laments the inability to bring this technology to the Army because of: 1) cost, 2) security and 3) policy. It further states:

> Any attempt to provide the "Bring Your Own Device" environment would most likely require . . . a *radical change* in how [the Department of Defense] and the Army protect its information from one of protecting the network to a philosophy more like the commercial world in which the Internet is for the most part unprotected and each organization protects [its] own information at the source.[2]

The problem

As a parent, the worst possible punishment I could give my teenage children is taking away their smartphones. The threat of this alone is enough to get the dishes, lawn and bedrooms all looking beautiful in short order. Any other sort of threat—for example, revoking television, computer or even driving privileges—does not seem to carry as much weight. That is because Generation Y (generally, people born after 1980) are a connected bunch. They text, chat, connect to social media sites and watch Internet videos—all on their phones. Come to think of it, my feeble threats do not amount to much at all. *The New York Times* stated that Americans ages 12 to 34 are spending less time in front of TV sets while increasingly turning to their tablets and smartphones to watch video.[3] You can get just about anything on your phone. I watched NCAA March Madness live on my iPhone connected to Gogo in-flight Internet while flying at 30,000 feet. Also, the phone itself is a powerful computer, running just about every sort of game and other application (app) you can imagine. And as for a ride—well, they can always just call or text a friend to pick them up. Actually, the Federal Highway Administration showed that from 1983 to 2008, the share of 16- to 39-year-olds with driver's licenses had declined markedly.[4] In 1983, 69 percent of 17-year-olds had a driver's license; by 2008, that had dropped to 50 percent. Clearly, continuous virtual contact with friends through electronic means is reducing the need for face-to-face visits. The smartphone has really become the conduit for a young person's connectedness to the world. Without it, they feel naked and alone.

The Landpower Essay series is published by AUSA's Institute of Land Warfare. The series is designed to provide an outlet for original essays on topics that will stimulate professional discussion and further public understanding of the landpower aspects of national security. This paper represents the opinions of the author and should not be taken to represent the views of the Department of the Army, the Department of Defense, the United States government, the Institute of Land Warfare, or the Association of the United States Army or its members. For more information about AUSA and the Institute of Land Warfare, visit our website at www.ausa.org.

Perhaps I am being a little over-dramatic here, but then I think about how the moment I walk into a secure military facility, the first thing they ask me to do is to remove my cell phone. All of the ingenious capability that enables me to connect, share, learn and accomplish the mission—not to mention the "cool" factor—gets left at the front door. Even though I am over 40, I am still fond of my smartphone and have grown quite attached to it. I cannot imagine that I was as productive before I got it, despite the occasional sessions on Angry Birds or Words with Friends. Since more than 90 percent of the Army is under age 40[5] (and one could safely say that it is an active force of Generation Y members), I expect these Soldiers feel the punishment that much more when they have to surrender their phones.

The power of a smartphone is not just that it is a fully functional computer in your hand; it is a customizable system to facilitate individual e-learning and provide positional or operationally relevant data in real time. Furthermore, it is a reflection of one's own personality. The apps people have on their phones reveal a great deal about what they value and how they operate. For example, I am a seasoned traveler, so my phone includes an app for finding good local cuisine and another to locate the best deals on local gas prices. Another app helps me find ATMs within my credit union network, and another reports on news from back home. I especially like the one that allows me to access my DVR at home remotely to set up recordings. These examples might seem somewhat trivial, but just as these apps buy me situational awareness in the domestic urban environment, one can conceive of many ways in which a smartphone could enable an operator in the field across all six warfighting functions: movement and maneuver, fires, intelligence, sustainment, command and control and protection.

A potential solution

Meanwhile, back in a lab somewhere, some guys are—ahem—*modifying* a commercial off-the-shelf phone to achieve greater performance. Okay, we will just call it "hacking." But these guys are ethical hackers—white hatters—and their ultimate aim is better security. They have rooted an Android phone; that means they have taken the stock configuration of the phone, exploited a security flaw in the firmware (for example, unlocking the boot-loader with the command "fastboot oem unlock" from a connected development PC) and written a new system image with Superuser enabled. This lets them do whatever they need to do from the kernel upward without having modified the phone's underlying hardware or firmware. According to the U.S. Copyright Office, this is completely legal, and many modern devices feature unlockable boot-loaders. However, there are still some carriers who are resistant to offering this feature because: 1) they give up some control over the device and 2) open-source modifications offer features for which they would otherwise charge a premium (for example, tethering).

What exactly are these guys in the lab doing with their root privilege? They are removing the Android Debug Bridge (ADB)—the tool that comes with the Android Software Developers Kit and without which it is much more difficult to modify the phone. They are adding in a monitoring program that logs entries and notifies the user of security events. They are removing all USB capability, meaning you can charge the phone only from a wall outlet. And they are adding in layers of encryption for data at rest and data in transit using native functions of Android's underlying Linux kernel.

By the way, these hackers are at the National Security Agency (NSA), and they are merely demonstrating the new way of doing business. That is, they are taking untrusted hardware from Korea and untrusted firmware from India and adding some layers of security hardening and monitoring to create a system that can process Top Secret data. They already have a working pilot device that has an Authority to Operate (ATO)—but they cannot bring it inside the building at NSA to use it. Thus, even though the technology exists to secure commercial systems, advances in policy are lagging behind.

The need for change

If secure systems are too difficult to use, people will circumvent them in favor of something easier. No amount of threatened punishment will change this. Therefore, modern thinking is that some security risk is acceptable to accomplish the mission. Some security is better than none at all.

During a "Community Day" when NSA invited research and development (R&D) partners to witness the unveiling of its new plans for securing emerging mobile technology, Troy Lange, Mission Manager for Mobility in NSA's Information Assurance Directorate (IAD), related an interesting story. When he worked in NSA's counterterrorism unit, he was notified of an incident that caused him to call his FBI counterpart at home in the middle of the night. Mr. Lange asked the other fellow if he could go secure, and he said he thought he could, because he had a Secure Mobile Encrypted Portable Electronic Device (SME PED); however, he had never used it. It struck Mr. Lange that this guy must surely have had earlier opportunities to discuss sensitive incidents and had evidently been finding other means to do so on his own (non-SME PED) phone. Mr. Lange was not necessarily indicting his FBI counterpart; rather, he attributed it to the fact that if one has something to discuss quickly and productively, one will use the means that best facilitate this. By many accounts, SME PED is cumbersome, difficult to configure and manage and requires a savvy user; therefore, it is often left on the shelf.

The failure of SME PED—and Mr. Lange himself dubbed it that—stems from the fact that the old NSA certification cycle was just too long. By the time a system was certified, the technology on which the design was built had largely become obsolete. While SME PED development began in the early 2000s, something seismic happened while it was winding its way through the maze of NSA testing: the iPhone. By the time the SME PED hit the market in earnest in 2009, the iPhone 3GS had been developed. The BlackBerry-like SME PEDs are functional but do not have cameras, GPS or music—or, for that matter, Siri or Draw Something. They also require a lot of infrastructure to support in terms of configuring a special e-mail server and some learning curve on the part of the user. Note that iPhones ship without much of a user's manual; they're designed to be turned on and used intuitively.

NSA introduced a new program called Commercial Solutions for Classified (CSFC). With it, NSA plans to increase trust of commercial products, increase the functionality of government solutions to be equivalent (or comparable) to commercial consumer solutions, and harden, defend and hunt within commercial networks. These are laudable goals but admittedly fraught with potential for derailment. NSA believes that if a mobile solution gets through the new, improved gauntlet of Common Criteria certification from one of the National Information Assurance Partnership (NIAP) labs and Federal Information Processing Standard (FIPS) Publication 140-2 certification for encryption, then it can be placed on the NSA "Preferred Components List" and be used to build a secure mobile platform. Mind you, NSA is not in the business of building smartphones—its domain is specifying requirements and endorsing products and, presumably, mobile carriers. The NSA pilot project aimed only to show that it could be done.

The first question one might ask is how a product could get through NIAP and FIPS 140-2 validation in, say, less than a couple of years. The response from NSA is that it will now take only three months! How is that possible, you ask? NSA plans to change the Common Criteria process and persuade industry to go along with it.

If you were not already up to speed on NIAP, let us go back to the days of the *Trusted Computer Security Evaluation Criteria*, also known as the "Orange Book." Under NSA's Trusted Product Evaluation Program (TPEP), products would be evaluated against the Orange Book criteria and graded according to their level of access control. Windows NT, for example, was evaluated at level C2, which meant it had implemented the necessary security to achieve discretionary access control. The only problem was that if a product had been evaluated under the TPEP in the United States, it would have to undergo some equivalent evaluation in Canada, the United Kingdom, Germany or wherever else one wished for it to be accepted. This was costly. So an international coalition created the Common Criteria for Information Technology Security Evaluation. Now a system certified in an approved lab in one country would have reciprocal recognition in other member countries as well.

In a nutshell, the Common Criteria are a set of generic functional and assurance requirements that cover all aspects of security for a product. When the government specifies the requirements for a product, it writes a protection profile that lists the threats, assumptions and policies governing the system and, in turn, maps these to security objectives for the system. These objectives are mapped to functional and assurance requirements. Functional requirements are technical controls that implement security in the system; for example, "The Target of Evaluation (TOE) security function will lock out a user after [specify some number of] incorrect log-in attempts." Assurance requirements specify the depth to which the security controls will be proven. The protection profile says, in essence, "This is what I want." Another document, called a security target, specifies how these requirements have been met in the actual product. This document essentially says, "This is what I have."

In addition, the stringency to which the assurance requirements are held sets an evaluation assurance level (EAL) for the system. These are roughly analogous to the old TCSEC levels. For example, NIAP says that Windows 7 has been evaluated at EAL4, augmented with ALC_FLR.3 Systematic flaw remediation; the latter caveat means that Microsoft has to establish a security center and publish security bulletins and patches. However, NSA said that it has found the assurance requirements and enforcement thereof to be very subjective, varying widely from lab to lab—and thus a source of great consternation and impediment to certification.

Therefore, NIAP is going to change a few things. First, it is going to accept only products that correspond to NSA-published protection profiles. In other words, there will be no more custom-developed protection profiles or security targets. All security targets must correspond to an NSA-published protection profile. This should theoretically make things more uniform and easier to test. There will be one protection profile for each technology—e.g., firewall, operating system, virtual private network (VPN), etc. The timetable for these approved protection profiles to be published has not yet been determined, but NSA says it will endeavor to publish them every few months. Second, NIAP is going to eliminate EALs altogether in favor of one assurance level (i.e., pass/fail). Commercial off-the-shelf operating systems will be evaluated at the same EAL as, say, high assurance guards. NSA believes these changes will result in a streamlined NIAP process that takes less than 90 days from start to finish.

Radical change, indeed!

What this means for the Army is that the certification and risk assessment process has largely been removed from NSA and now sits squarely on the Army Designated Approving Authority (DAA). If the Army develops a system using approved components from NSA's CSFC Component List, then NSA says that system can process classified data without going through the usual, protracted certification process within NSA's labs. That means it is more likely that the Army could get closer to the dream of a "Bring Your Own Mobile Device" solution. Admittedly, NSA says this approach will not apply in every mission scenario, but it can provide a path to success in an environment in which the threat vector is well understood, and the DAA deems the risk to be acceptable.

It would appear that the recent paradigm shift by NSA is in line with the "radical change" that was called for in the CSDA White Paper. Will the Army agree that the benefit of anytime/anywhere access to information is important enough to call for a fundamental transformation of current policies and practices?

A Capability Package is NSA's new way of specifying requirements for technologies. It essentially covers product selection—e.g., NIAP protection profiles and FIPS 140-2 (if there is an embedded cryptographic component), secure configuration guidance, product evaluation testing and residual risk analysis (usually in a classified annex). Since NSA wants to move away from government-built solutions to standards-based solutions, Capability Packages are based completely on open standards and are focused on interoperability.

Many people were surprised to see the Mobility Capability Package for Voice over Internet Protocol (VOIP) version 1.1 posted on the public website nsa.gov in February 2012. (Version 1.2 became available on 26 March.) It was an unprecedented move for an organization that has traditionally been shrouded in secrecy to post a document containing the cookbook for how to develop a device that can process classified data. But as the document itself says, the government

> can no longer afford to develop its own expensive, and potentially untimely, security solutions. Instead . . . the National Security Agency, working with its partners, customers and industry, will develop security solutions based upon commercially available products that will enable customers to layer and compose solutions that ensure their systems and information are reliable, protected and available.[6]

The 100-page Mobility Capability Package for VoIP document begins with an overview of enterprise mobility and then goes into an overview of secure VoIP. However, the way it aims to achieve security is to strip out from smartphones much of the functionality that makes the device "smart" in the first place. According to NSA project engineer Dale Rhoton, "Anything [whose function] we couldn't figure out . . . got removed."[7] The first version of the Mobility Capability Package covers only VoIP; NSA is working feverishly to release the next version that covers web browsers. After the overview, the document delves into solution-specific requirements covering: 1) the operating system and mobile apps, 2) carrier services, 3) Enterprise Mobile Infrastructure and 4) secure VoIP service. Each section discusses design approach, gap analysis and risks. (Residual risks are documented in a separate classified annex).

Some questions and answers

NSA's prototype solution developed under the Mobility Capability Package assumes that a phone's hardware and firmware are untrusted and that the modifications will mitigate any risk coming from below. NSA provides monitoring for the operating system, device drivers and kernel through rooting the phone to remove unwanted apps and installing its own monitoring app. Besides dumbing down the smartphone, there are some other concerns about the security of the plan. For example, if one exploited a flaw in the firmware to root the phone and add in all of these runtime protections, what is to keep an adversary from doing the same and just removing those added-on protections himself? NSA, with much more work still to do on the project, will implement incremental changes and is actively seeking feedback (contact mobility@nsa.gov).

Suppose NSA were not going to remove from the phone every app that makes it smart. Then it would need some sort of app security vetting process. That process is not currently covered in the Mobility Capability Package, and in my own discussions with numerous NSA, Army, Defense Information Systems Agency (DISA) and other DoD leaders, I have found such a process to be lacking. In fact, I asked Brigadier General Jennifer L. Napper, Commanding General, Army Network Enterprise Technology Command/9th Signal Command (Army) (NETCOM/9SC), and Mr. Daniel Bradford, Deputy to the Commander/Senior Technical Director, NETCOM/9SC, about this at the August 2011 LandWarNet Conference, and they both said they were open to hearing ideas. I believe we have a workable solution that needs to get on the road to becoming a standard, and it stems from what was done in the last "Apps for the Army" (A4A) challenge.

That contest, which began in March 2010, asked Army developers to submit software applications for mobile phones in certain categories including morale, welfare and recreation; Army mission; information access; location awareness; and training. According to then-Chief Information Officer/G-6 Lieutenant General Jeffrey A. Sorenson, about 140 individuals or teams signed up to participate in the program and more than 50 applications were submitted by the 15 May deadline. Clearly, if the contest winners were to be announced by the August 2010 LandWarNet Conference, there needed to be a streamlined validation process, because there simply was not time to get all of those through the DoD Information Assurance Certification and Accreditation Process (DIACAP). I was part of a team that helped do the testing of those apps, and we came up with a process that looked like this:

1. Run through the DISA Security Technical Implementation Guide (STIG) Application Security Checklist, which consists primarily of interview questions for the developer on whether security considerations were baked into the application;
2. run an automated vulnerability scanning tool on the code;
3. perform some rudimentary checking of inputs/outputs to make sure that there was proper sanitization done to curtail attacks of cross-site scripting, Structured Query Language (SQL) injection, etc.; and
4. use an automated work flow tool to produce a comprehensive scorecard for the app.

As opposed to a typical DIACAP validation that takes, on average, three to four months to execute, this "certification lite" process used for the A4A contest took three to four days per app. Thus, the testing team was able to complete all 53 tests before the LandWarNet conference, with 25 of them receiving an ATO and qualifying to be placed on the Army's App Marketplace. (The others needed a bit of remediation.)

Let us go back to the part about the automated vulnerability scanning tool. This area was actually found to be somewhat lacking during the A4A vetting process. There were some general-purpose static vulnerability scanners, such as IBM's AppScan or Hewlett-Packard's Fortify, but these did not necessarily check for issues specific to mobile phones, and they were prohibitively costly (up to $87,500 for a single auditor license with another $15,000/year annual maintenance fee). As I now lead a team doing mobile security R&D, I found this to be an area ripe for innovation, and our team has been working on its own tool for analysis of Android apps in particular—the goal being to create an efficient, repeatable process that reduces the amount of manual review.

According to the *Fiscal Year 2011 Report to Congress on the Implementation of the Federal Information Security Management Act of 2002 (FISMA),* "A well-designed and well-managed continuous monitoring program can effectively transform an otherwise static security control assessment and risk determination process into a dynamic process that provides essential, near real-time security status."[8] This is further emphasized in National Institute of Standards and Technology (NIST) Special Publication 800-37, Revision 1, *Guide for Applying the Risk Management Framework to Federal Information Systems: A Security Life Cycle Approach*, where continuous monitoring is listed as a major component in the Risk Management Framework (RMF). The term "continuous monitoring" is mentioned 47 times in that FISMA report. The NSA Mobility Capability Package prototype has a monitoring program that logs entries and notifies the user of security events, even allowing users a reboot or wipe function for significant security faults. This is a great start, but even more is needed.

You may have read about the Carrier IQ rootkit scandal,[9] which is basically this: Carrier IQ sells software designed to help wireless service providers and device makers identify and diagnose service- and quality-related problems such as dropped calls and battery life. The software is installed on more than 150 million devices worldwide. However, security researchers have shown that the Carrier IQ software can be easily tweaked to conduct surreptitious and highly intrusive tracking of Android, BlackBerry and other smartphone users, equating to a stealth keystroke-logging rootkit. It can even collect data on a user's location, app use, web-browsing habits, texts, etc. In the NSA pilot, all Carrier IQ software was probably just removed, but other developers who are trying to follow the emerging Mobility Capability Package may be loath to follow that "when in doubt, throw it out" mindset for fear of hurting the user experience. Therefore, clearly some sort of host-based firewall protection is required. While there are a couple of open-source Android firewalls out there, these are primarily for preventing battery drain from apps "phoning home" to servers. My mobility R&D team put together its own firewall that has the look and feel of "Zone Alarm" (used on many home computers). It blocks egress activities from the phone and gives the user a cancel/allow capability to add sites to a white list. The NSA model may be incomplete and will certainly require a few more host-based security protections to get it ready for mainstream use.

This brings me to the final area of security—for which NSA is noted for its tremendous expertise—encryption. The NSA Capability Packages for Mobility, Multi-Homed VPNs and Wi-Fi (the latter two are drafts but due out imminently) all call for a curious scheme that incorporates a "rule of two." That is, they require a system to implement encryption inside another encryption, or a tunnel within a tunnel; and the two must be different implementations from different vendors. This raises a multitude of questions about how or why this would necessarily raise the security of the system, to which NSA responds, "Twice is nice." In other words, NSA probably did a security test (the results of which are classified) and came to a determination that this method offered low risk. People might also note that on a mobile platform, encrypting is expensive in terms of processing capability and battery power. We are still a year or two away from processors providing enough punch to enable this on a large scale, although NSA states that its pilot works well (of course, with everything but VoIP stripped off).

In the final analysis, a number of issues remain to be worked out. NSA's first pilot is not at all a scalable solution. There is poor component separation (in terms of security versus performance). And there is not wide support available from any commercial vendors for NSA Suite B Encryption compatibility. However, there are also some very positive things. In the pilot, NSA showed that all of the solutions they put into the system for secure data at rest and data in transit were implemented using open source methods—in other words, Android's underlying Linux architecture provides native methods for encryption that meet NSA's standards. According to Mr. Rhoton, the team began its efforts in January 2012 and already had a working prototype in March 2012. Compared to a traditional developmental system, this is a vast improvement. Finally, the end product has the intuitive, familiar feel of an ordinary smartphone because it was made from commercial components. Thus, there is a low learning curve to using it.

Final thoughts

In conclusion, we know that the Army must meet the requirements of Digital Age information access by developing an environment in which both Soldiers and civilians may access commercial and DoD knowledge repositories from anywhere and at any time. But access has to be through a fully technology-enabled smart device, and the look and feel of the device must be commensurate with available consumer technology. Without this, users will either unwittingly or willfully skirt the security policy and use their own means to get to the data in a quicker and more convenient manner. Further, we know that the traditional DoD model of years of certification testing by NSA does not meet the prior requirement; that process takes too long, and once a product finally gets through, it is already ready for the scrap heap. NSA has introduced a new process that layers security measures on top of untrusted commercial components. The plan still has some distance to go but shows promise for allowing the Army to keep pace with ever-improving mobile technology. The enabling factors are streamlining the NIAP certification processes and relying on open-source standards. Whether this means we now have the capability to ultimately enable "Bring Your Own Mobile Device" remains to be seen, but it is a move in the right direction. The Army should quickly seize on this new direction to enable rapid development of mobile technologies to aid the warfighter.

Endnotes

[1] Connecting Soldiers with Digital Applications (CSDA), White Paper ver. 1.1: "Building the Case for a Bring Your Own Device Solution"; Network Integration Branch, Mission Command Center of Excellence, Combined Arms Center, Fort Leavenworth, KS, 21 December 2011.

[2] *Ibid.* Emphasis added.

[3] Brian Stetler, "Youths Are Watching, But Less Often on TV," *The New York Times*, 8 February 2012, http://www.nytimes.com/2012/02/09/business/media/young-people-are-watching-but-less-often-on-tv.html?_r=3&hpw.

[4] Joan Lowry, "Young Americans less likely to drive," Associated Press, 6 April 2012, http://news.yahoo.com/young-americans-less-likely-drive-203002275.html.

[5] Defense Manpower Data Center, *Active Duty Demographic Profile—Assigned Strength, Gender, Race, Marital, Education and Age Profile of Active Duty Force*, PowerPoint presentation, September 2008.

[6] National Security Agency, Information Assurance Directorate, "Mobility Capability Package for VOIP ver. 1.2," 26 March 2012.

[7] Notes from "NSA Community Day for Securing Emerging Mobile Technology" at Johns Hopkins Applied Physics Lab (APL), 19 March 2012.

[8] White House, *Fiscal Year 2011 Report to Congress on the Implementation of the Federal Information Security Management Act of 2002*, 7 March 2012.

[9] Jaikumar Vijayan, "FAQ: Behind the Carrier IQ rootkit controversy," Computerworld.com, 1 December 2011, http://www.computerworld.com/s/article/9222332/FAQ_Behind_the_Carrier_IQ_rootkit_controversy_.

Jeremy Rasmussen, CISSP, is a Senior Principal Information Security Engineer at CACI. He has more than 20 years of experience in developing secure communications systems and cybersecurity services for the Department of Defense. He is also an adjunct professor at the University of South Florida (USF), teaching courses in Cryptography and Network Security, Computer Forensics and Investigations, and Ethical Hacking.

Association of the United States Army
2425 Wilson Boulevard, Arlington, Virginia 22201-3385
703-841-4300 www.ausa.org