Written Testimony of

Dr. Frederick R. Chang

Bobby B. Lyle Centennial Distinguished Chair in Cyber Security

Southern Methodist University

Before the

Committee on Science, Space and Technology

U.S. House of Representatives

Hearing on

“Is Your Data on the Healthcare.gov Website Secure?”

November 19, 2013

Chairman Smith, Ranking Member Johnson, Members of the Committee, thank you for the opportunity to testify before you in today’s hearing on the topic of data security and the new healthcare.gov website. My name is Frederick R. Chang and I consider it an honor and a privilege to come before this Committee again. I have very recently made a return to academia and I am now the Bobby B. Lyle Centennial Distinguished Chair in Cyber Security and Professor in the Department of Computer Science and Engineering at Southern Methodist University in Dallas, Texas. I am also a Senior Fellow in SMU’s John G. Tower Center for Political Studies, an Adjunct Professor in the LBJ School for Public Affairs and a Distinguished Scholar in the Robert S. Strauss Center for International Security and Law at the University of Texas at Austin. In prior positions, I have served at the National Security Agency (as Director of Research); in academia (at

The University of Texas at San Antonio and at The University of Texas at Austin); and in the private sector (at 21CT, Inc., SBC Communications, Pacific Bell, and Bell Laboratories). I would also mention that I have served as a member of the CSIS Commission on Cybersecurity for the 44th Presidency and I am currently a member of the Texas Cybersecurity, Education, and Economic Development Council.

Regarding SMU, it is a nationally ranked private university in Dallas founded 100 years ago. The university enrolls nearly 11,000 students - including about 4,600 graduate students - who all benefit from the academic opportunities and international reach of seven degree–granting schools. SMU is recognized by the Carnegie Foundation as a university with “high research activity," which ranges across disciplines from particle physics at the Large Hadron Collider at CERN, to geothermal energy, to the science of human speed, to cyber security through the Bobby Lyle School of Engineering.

A brief historical observation

As we meet today to talk about Internet data security and healthcare.gov, I think it is an interesting coincidence that this hearing is being held in the same month that we observe the 25th anniversary of the Internet worm of November 1988 (also known as the Morris worm). It was the first worm to receive widespread media attention as it caused a major disruption on the Internet in its day. Today, our opponents in cyberspace are intelligent, seam-seeking, shape-shifting adversaries, that have an uncanny ability to penetrate and evade cyber defenses and compromise the targeted system. I am very pleased to be part of a discussion that will explore ideas that may serve to enhance the security of a web application that will be accessed by so many Americans.

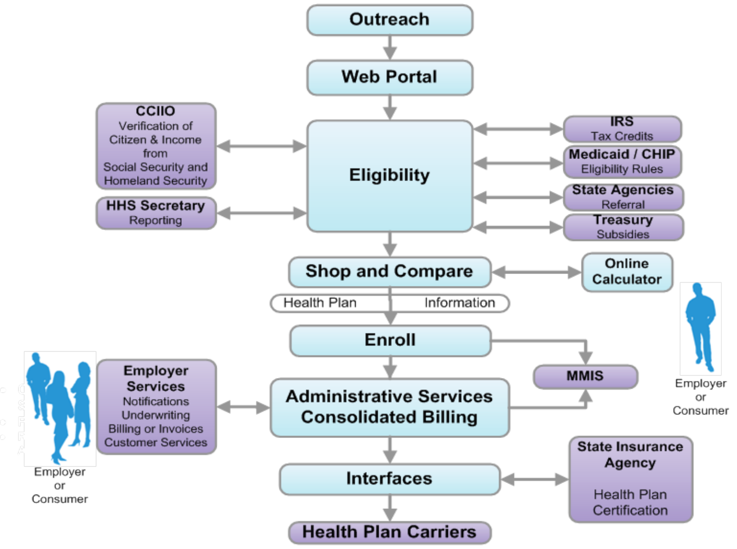
Complexity Risk

When it comes to security, complexity is not your friend. Indeed it has been said that complexity is the enemy of security. This is a point that has been made often about cybersecurity in a variety of contexts including, technology, coding and policy (1, 2, 3).

The basic idea is simple: as software systems grow more complex, they will contain more flaws and these flaws will be exploited by cyber adversaries.

This is a difficult dilemma. We want more functionality and capability in our software applications but the price we pay is added complexity which results in a corresponding increase in security vulnerabilities.

As I was preparing for this hearing, I came across a number of articles that commented on the size, functionality and complexity of what was being accomplished as part of healthcare.gov. I even came across a graphic from Xerox published in the Washington Post dated October 9, 2013 (4) and in the hope that a diagram is worth a thousand words, here it is:



In displaying this diagram, I don't intend to describe it, but rather use it to give a sense for the application's complexity. I would observe though that to get a quote for health insurance, the task of the "back end" software seems especially complex and challenging. As I understand it, the system needs to access servers and databases at the Internal Revenue Service, Medicaid/Children's Health Insurance Program, various state agencies, Treasury, the Social Security Administration, the Department of Homeland Security, and Health and Human Services. It also needs to connect to all the health plan carriers to get pre-subsidy pricing. All this input is fed into the on-line calculator for display to the end user.

While we are on the subject of the back end, I would also make a point about the back end databases that are listed in the top half of the diagram. These databases obviously contain a tremendous amount of sensitive information and as a result would be attractive targets for attackers. The new services that are being introduced are increasing the access channels into these sensitive databases and as a result the size of the “attack surface” has increased. I believe that this increased attack surface is a risk worth mentioning.

Web Applications Risk

Over the past many years we’ve all grown accustomed to conducting business over the web: buying books, videos, airline tickets and so much more. The convenience and business benefits of conducting transactions via applications that run over the web are clear. For some, it might be getting hard to remember a time when we didn’t use these web applications to conduct business. But with the convenience and benefits come security risks. The web was originally designed to display static, read-only pages, and as a result there was little intrinsic security. Some web security technologies were added later, and while things are improving somewhat, the majority of websites have security vulnerabilities today (5). We’ve known for some time now about the security risks associated with websites and indeed they have been analyzed, cataloged and rank ordered by an open-source, non-profit organization known as the Open Web Application Security Project (OWASP). For about the past decade or so they have been publishing the top 10 web application security risks and a 2013 list has recent been published (6). If you look through this reference you’ll read about items like: injection flaws, cross-site scripting, cross-site request forgery and more. Without going into the details, I’d just mention that these risks are of concern as they could lead to attackers querying or compromising the website with the goal of obtaining sensitive information.

I have not performed any analysis personally to determine whether these risks (and related risks) are present on the healthcare.gov website, but there have been some web posts based on unobtrusive, passive analysis that have raised some concerns along these lines (e.g., 7, 8). I understand that improvements to the website are on-going, so some of these concerns may have been addressed since they were reported.

Risk from bogus websites

The arrival of the new healthcare.gov website was accompanied by an array of fake websites that are designed to capture sensitive information that users enter into that fake website, believing that it is authentic. The information could then be used for purposes of identity theft. There was one report that mentioned that within the first few weeks of the introduction of healthcare.gov, over 700 fake websites had sprung up (9). I believe that this is a substantial threat vector, and others have observed this as well (10). Indeed, one need only to look at the results from fraudulent tax returns in the U.S. due to identity theft to conclude that considerable concern is warranted (11).

The fact that there is not one single place to sign up for health care coverage will lead to confusion by the public. There is the main federal site, individual state sites, as well as legitimate third party sites. As I understand it, there is no official designation or marking that a consumer can use to determine whether they are on the correct site or not. As people seek to register for health care coverage they may find that there are a dizzying array of websites to select from. When it comes to typing in information like a social security number into a web form, many people might be cautious about doing so, but given that it has do with health insurance coverage people might be more inclined to do so (particularly if they think the request is coming from a legitimate website). These two factors could combine to create a ripe circumstance for personal information to get into the wrong hands. It is difficult to estimate how much traffic these fake websites will siphon off, but it could be significant (12).

A variant of the above scam would be for a bogus website to trick a user into downloading a piece of malware ("malware" is a catch-all term that refers to malicious software that may take the form of a virus, a worm, a trojan horse, a keylogger and the like). That malware could cause the user's computer to become part of a botnet or could capture keystrokes representing sensitive personal information leading to identity theft. A related variant would be that custom malware gets written specifically for the purpose of capturing information being entered into a health insurance exchange website, similar to what has happened in the context of on-line banking (e.g., Zeus malware).

Countermeasures

Ideally, security is built into an application from the very beginning rather than having it "bolted on" afterwards. Many in the security field have emphasized this point (e.g., 13). With the rise in cyber attacks, the "breach then fix" model is becoming untenable. Data breaches are harmful to its victims, time-consuming and costly to repair, damaging to enterprise reputation, and more. An application can't be perfectly secure, but there are proactive things that can be done to reduce the risk of a successful attack. Let me mention a few such items here. Security should be integral to the application design (e.g., think like an attacker, secure the weakest link, fail securely). Security should be part of the software development lifecycle. Secure coding practices should be employed – in fact there are now published lists of top programming errors that coders make that lead to security problems (14). I previously mentioned the OWASP Top 10 Web Application Security Risks initiative and each of those risks are listed along with, among other things, secure coding countermeasures. Security penetration testing should be routine and continuous – before and after the system goes operational. Indeed I know of one company that uses a third-party service to conduct quarterly, unscheduled penetration tests after the application has been fielded, understanding that cyber adversaries will constantly adapt and modify their attacks.

As it relates to consumers, when the topic of "Internet security" comes up, it is easy to begin thinking about traditional technologies like network firewalls and anti-virus software. And while those technologies are certainly valuable and should be used, they won't help much when it comes to most of the web application security risks that have been discussed. Regarding the risk of bogus websites, it is very important for consumers to understand that they need to be absolutely certain that they are accessing the correct health insurance exchange website. As I mentioned earlier, there are already many, many fake websites, and to the extent that users are confused about where to go, they may be lured to the wrong place. Users should start their search for coverage on the actual healthcare.gov website and not via a search engine.

Science of cybersecurity

As I was preparing for today's hearing, I was reminded on a few occasions of my previous appearance before this Committee earlier this year on the topic of cybersecurity research and development. I spoke of the need for a science of cybersecurity. In our desire to move from reactively responding to cyber intrusions to proactively getting ahead of the problem we are limited by a lack of rich and reliable sources of data; solid, well-honed metrics; a deep research base providing understanding of the social science (e.g., economics, psychology) issues and consequences; laws or principles from which we can make reliable predictions about relevant cyber phenomena – and so much more. As we talk about important shorter term measures that can be taken to improve the security of healthcare.gov, there are myriad longer term issues that need to be addressed as well. At the beginning of my remarks, I mentioned Internet security issues dating back to 25 years ago. When it comes to cybersecurity, the problem is not going to go away anytime soon. Creating a cybersecurity science will be of critical importance to us in the long struggle ahead.

Thank you again for allowing me the opportunity to be here today. I look forward to your questions.

References

1. Schneier, B. (2000). Crypto-Gram Newsletter, March 15, 2000, Software Complexity and Security. https://www.schneier.com/crypto-gram-0003.html#8
2. McGraw. G. (2006). *Software Security: Building security in.* Addison Wesley, Boston, MA.
3. Geer, D.E. (2008). Complexity is the enemy. IEEE Security & Privacy, vol. 6 (6), pp. 88.
4. http://www.washingtonpost.com/blogs/the-switch/wp/2013/10/09/heres-everything-you-need-to-know-about-obamacares-error-plagued-web-sites/
5. http://www.darkreading.com/vulnerability/websites-harbor-fewer-flaws-but-most-hav/240154118
6. https://www.owasp.org/index.php/Top_10_2013-Top_10
7. http://blog.isthereaproblemhere.com/search/label/Healthcare.gov+Security
8. http://h30499.www3.hp.com/t5/HP-Security-Research-Blog/Healthcare-gov-Affordable-Care-should-be-Secure-Care/ba-p/6227015
9. http://washingtonexaminer.com/obamacare-launch-spawns-700-cyber-squatters-capitalizing-on-healthcare.gov-state-exchanges/article/2537691
10. http://blog.trendmicro.com/coming-risk-scam-obamacare-sites/?sf17662278=1
11. Anderson, R., et al (2012). *Measuring the cost of cybercrime.* In 11th Workshop on the Economics of Information Security, Berlin, Germany, June 2012.
12. http://washingtonexaminer.com/obamacare-launch-spawns-700-cyber-squatters-capitalizing-on-healthcare.gov-state-exchanges/article/2537691
13. G. McGraw, "Software security", IEEE Security & Privacy, vol. 2 (2), 2004, pp.80-83.
14. http://www.sans.org/top25-software-errors/