Testimony of Evan Cooke
API Task Force Virtual Hearing, 1/26/16

Bio

Evan recently joined the U.S. Digital Service team at the White House where he supports Administration cybersecurity and healthcare interoperability initiatives. Prior or joining USDS, Evan was Co-founder and Chief Technology Officer at Twilio, Inc., a cloud API platform for building intelligent communications applications. As CTO, Evan recruited and led the technology team and defined a culture of rapid delivery and high-availability. He completed his M.S., Ph.D. and postdoctoral fellowship in Computer Science at the University of Michigan, with a focus on network security and distributed systems and his undergraduate degrees in Electrical Engineering, Computer Science and Psychology at the University of Wisconsin.

Remarks

First, thank you for the opportunity to participate on this panel. It is an honor to be here on behalf of the U.S. Digital Service and to support the application of modern API technology to improve healthcare outcomes in our nation.

I'd like to start by emphasizing the incredible power of APIs. The Department of Education recently launched an updated College Scorecard tool built on top of an open API with data from 7,000 colleges and universities going back 18 years. This API makes it easier for software developers and researchers to extract, customize, and build upon the data to support students and families to help them make better college choices. The result has been diverse ecosystem of partners that supports better college search and choice tools, better advising and support for students, and more comprehensive rankings with new outcomes data. This is just the beginning of what is possible and it gets me incredibly excited.

The idea I'd like to explore in my comments today is the notion of APIs as collections of technologies and standards rather than monoliths. As we look to the future of APIs in healthcare and how to promote security, privacy, innovation, and interoperability, it helpful to consider a fine-grained approach.

A common way to describe APIs is as "software contracts" between parties. Those parties could be private companies, individuals, or public entities like Federal, state or local governments. APIs can capture almost any form of business process or exchange of information if the data can be represented in digital form that can be exchanged over a network.

I'll start by sharing a brief story from my previous experience in the private sector. Years ago when I co-founded Twilio, we were able to implement our own REST API with tailored security mechanisms and data formats. During the first few years, the API changed quickly and the ability to make and deploy changes to the API was critical for meeting customers needs, providing better scalability, and improving security. While some of parts of the APIs changed quickly, other pieces such as serialization formats like WAV or MP3 did not change.

Rather than a single entity, APIs are composed of many parts such as network protocols, security mechanisms, authentication and authorizations means, request/response methods, and serialization formats. Those parts may need to change at different rates based on their maturity and broader changes in the products that the APIs support. Thus, as we think about APIs and processes for

standardization and certification, it may be helpful to think through each component separately as appropriate.

Because the requirements for each part of any API can be different, the specificity of guidance may also need to be adjusted depending what component of the API is referenced. For example, we might decide to dictate a specific technical format for a mature serialization format but provide higher-level guiding principles for the request/response approach. As an illustration of possible levels of abstraction, consider the NIST Cybersecurity Framework that describes four different level of specificity including Function (Identity, Protect, Detect, Respond, Recover) Categories (e.g., Governance, Data Security), Subcategories (e.g., "Organizational information security policy is established"), and Informative References (e.g., NIST Special Publication 800-53).

The implication of this more fine-grained approach to APIs is that a uniform technical specification of an API and a corresponding certification of that specification may be difficult. One approach would be to standardize or certify parts of an API together or independently. Another approach, and one commonly used by private-sector cloud API providers supplying resources like storage, compute, workflows as services, is to certify the organization providing the service. That approach puts focus on the provider of an API rather than on the technical protocol.

There of course a lot to the question of building trust with API providers and with the data provided by APIs but in the interest of time, i'd like to conclude by reiterating my excitement for potential of APIs and to advocate for the notion that APIs should be considers as collections of separate functions and technologies that may need to change at different rates and may require different level of specificity in guidance.

Thank you for the opportunity to participate today and I look forward to the discussion.