SHOOK, HARDY & BACON L.L.P.
Gary Miller (Appearance *Pro Hac Vice*)
gmiller@shb.com
111 S. Wacker Drive, 51st Floor
Chicago, Illinois 60606
Telephone: 312-704-7700
Facsimile: 312-558-1195

Jesse J. Camacho (Appearance *Pro Hac Vice*)
jcamacho@shb.com
Mary J. Peal (Appearance *Pro Hac Vice*)
mpeal@shb.com
Ryan D. Dykal (Appearance *Pro Hac Vice*)
rdykal@shb.com
Daniel M. Staren (Appearance *Pro Hac Vice*)
dstaren@shb.com
2555 Grand Boulevard
Kansas City, Missouri 64108
Telephone: 816-474-6550
Facsimile: 816-421-5547

Mayela C. Montenegro (SBN: 304471)
mmontenegro@shb.com
5 Park Plaza, Suite 1600
Irvine, California 92614
Telephone: 949-475-1500
Facsimile: 949-475-0016

Attorneys for Defendant
TELESIGN CORPORATION

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA – SAN JOSE DIVISION

TWILIO INC., Plaintiff, v. TELESIGN CORPORATION Defendant.	CASE NO.: 5:16-cv-6925-LHK JUDGE: Hon. Lucy H. Koh **DECLARATION OF SETH NIELSON, Ph. D.** **JURY TRIAL DEMANDED**

I, SETH NIELSON, declare as follows:

1. I have been engaged by TeleSign Corporation and asked to opine on issues stated herein related to the Claim Construction proceeding involving U.S. Patent Nos. 8,306,021 (the "'021 Patent"); 8,837,465 (the "'465 Patent"); and 8,755,376 (the "'376 Patent") ("Asserted Patents"). I am being compensated at a rate of $400 per hour. No part of my compensation is dependent on my opinions or the outcome of this proceeding, and I have no financial or beneficial interest in any of the parties to this Claim Construction proceeding.

2. This declaration is based on the information that was available to me as of the date of this declaration. I may revise, supplement, or expand my opinions based on further review and analysis of information and opinions provided to me subsequent to the filing of this declaration, including information and opinions submitted by Twilio or its expert(s).

3. I received a B.S. in Computer Science in 2000 and an M.S. in Computer Science in 2004, both from Brigham Young University in Provo, Utah. I received my Ph.D. in Computer Science in 2009 from Rice University in Houston, Texas. A copy of my C.V. is included as Ex. A, including a list of all cases in which, during the previous four years, I have testified as an expert at trial or by deposition. Ex. A also includes a list of all publications that I have authored in the previous ten years.

4. I am the recipient of the Brown Fellowship and a Graduate Fellowship from the Rice University Computer Science Department. I was also a John and Eileen Tietze Fellow at Rice University.

5. During my final undergraduate semester, I worked both as a teaching assistant for a Computer Networking course and as a researcher in the Networked Computing Lab. In these capacities, I assisted students in debugging and designing TCP/IP protocol stacks, Address Resolution Protocol implementations, and Remote Procedure Call projects. I have collaborated on investigations of statistical traffic engineering for bandwidth allocation, including a published paper entitled, "Effective Bandwidth for Traffic Engineering."

6. My initial research work as a graduate student related to software engineering topics, with a special emphasis on how programmers think while creating and modifying code.

During my course work, I took a special topics class called "Programmer Cognition" as well as a graduate-level neuroscience class from the Psychology department. Based on my research, I proposed a concept called "Design Dysphasia," wherein a programmer or software developer becomes trapped in their approach to solving problems based on the paradigms and design approaches of the programming language. My research was published as "Design dysphasia and the pattern maintenance cycle," in the *Journal Information and Software Technology* August 2006. This work also was a major component of my Master's thesis.

7. Another part of my Master's thesis was the identification of how certain programming language concepts can be "mixed" together. I investigated practical mechanisms whereby the Python programming language could be extended to support features known as "functional programming" and "logic (or declarative) programming." Languages with this mix of features are known as "multi-paradigm" programming languages.

8. While working towards my Ph.D. at Rice University, I studied and published research related to networking and computer security. In 2005, I published a paper entitled, "A Taxonomy of Rational Attacks." This paper categorized and described the various types of attacks that one might see in a decentralized, peer-to-peer network. When there is no centralized authority, users have to cooperate to obtain service. The term "rational attacks" refers to the economic incentives to not cooperate while still exploiting the system for service.

9. My Ph.D. Thesis was entitled "Designing Incentives for Peer-to-Peer Systems," and it built on this concept. Given a network where participants cannot be forced to cooperate, the operation of said network must induce cooperation by design of the outcomes. In other words, it must be in each participant's best interest to contribute to the cooperative operation. I conducted experiments included simulated extensions to the BitTorrent peer-to-peer protocol for long-term identities and mechanisms for cooperative anonymity. I constructed my own simulator of the BitTorrent protocol, and simulated thousands of hours of operations. I built the core simulator in C++ and integrated the Python scripting language for the rapid specification of actual simulations.

10. I have more than 15 years working in industry as an engineer and consultant. From 2001 through 2003, I worked as a software engineer at Metrowerks (formerly Lineo, Inc.), where I had substantial responsibilities relating to software architecture, computer networking, and technical project management. In particular, I developed and maintained the GUI for the Embedix SDK (Software Development Kit), ported the Linx GUI of the Embedix SDK to Windows, created an automated system to forward Linux python scripts to a Windows GUI, and developed a packaging and automated updating system for client software. To complete these assignments, I wrote tens of thousands of lines of computer code in C++, C, Python, and Perl. I also used various tools such as CVS source control and libraries such as the QT framework.

11. The Embedix SDK was a tool designed for creating third-party embedded systems and deploying them to devices. The SDK provided guidance, tools, and graphical representations for third-party developers to use in creating their own BSP's (Board Support Packages).

12. During the 2004 fall semester of my Ph.D. program at Rice University, I identified a security vulnerability in the Google Desktop Search that could have allowed hackers to compromise users' computers and obtain private information. After contacting Google and assisting them in closing the vulnerability, we published the details of our investigation.

13. In 2005, I completed an internship at Google, where I designed and implemented a solution to privacy loss in Google Web Accelerator. The Google Web Accelerator was designed to increase the speed of browsing the Internet. Once installed on a user's computer, the browser would request all content through a Google Proxy. The proxy performed pre-fetching and extensive caching in order to provide fast and responsive service to the user. At the time of my internship, news reports had identified odd problems in which users of the Accelerator were accessing other individual's private pages. During my internship, I designed and implemented a prototype solution for this issue in C++.

14. From 2005 through 2011, I worked as a Security Analyst and later a Senior Security Analyst for Independent Security Evaluators. There, I developed a parallel-processing

based security tool, developed a FIPS-certified encryption library, developed hardware-accelerated encryption algorithms, developed encrypted file-system prototypes, developed an encryption library for an ISE client, performed port-scanning analyses, evaluated security protocols using formal methods and hand analysis, and evaluated security failures. I also designed and managed the implementation of a secure communication technology that splits trust between multiple SSL Certificate Authorities (CA), so that if one CA is compromised, the communication stream can still be safely authenticated. My work on the secure communications technology project led to the issuance of multiple patents. In total, I wrote hundreds of thousands of lines of code in C, C++, and Python, including projects where I had to implement the same functionality in two separate languages.

15. In 2011, I began work as a Research Scientist at Harbor Labs and continued with that consulting firm until fall 2015. I worked with a wide range of clients, specializing in network security, network communications, software architecture, and programming languages. I analyzed an extensive collection of commercial software, including software related to secure email, cloud-based multimedia delivery, document signing, anti-virus and anti-intrusion, high-performance routing, networking protocol stacks in mobile devices, PBX telecommunications software, VoIP, and peer-to-peer communications. I also analyzed security considerations for potential technology acquisitions, re-created heuristic signatures for 1995-era viruses, and re-created a 1995-era network for testing virus scanners of that time period in gateway virus scanning. I managed teams that reviewed technologies for compliance with various standards, such as HIPAA, and for security vulnerabilities.

16. Also at Harbor Labs, I reviewed technology and source code for multiple clients related to accusations of theft and/or misappropriation of trade secrets. These engagements included an analysis of C, C++, Java, Python, and other source code languages in high-frequency trading, e-commerce, and other similar systems.

17. I also assessed the security and privacy technologies and policies provided by a third-party vendor to the Center for Copyright Infringement (CCI). CCI represents content owners, such as the RIAA and the MPAA, in finding and reducing piracy online. Because this

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.