REDACTED VERSION OF DOCUMENT SOUGHT TO BE SEALED

DECLARATION OF AVIEL D. RUBIN

I, Aviel D. Rubin, declare as follows:

I. INTRODUCTION

1. I have been retained as an independent expert in this lawsuit by the law firm of Irell & Manella LLP on behalf of Juniper Networks, Inc. ("Juniper"). I submit this Declaration in support of Juniper's Opposition to Finjan, Inc.'s ("Finjan") Motion for Summary Judgment of Infringement of Claim 10 of U.S. Patent No. 8,677,494 ("Motion").

2. As discussed in further detail in this declaration, it is my opinion that Finjan has not established that Juniper infringes claim 10 of U.S. Patent No. 8,677,494 ("the '494 Patent").

II. BACKGROUND AND QUALIFICATIONS

3. I am being paid at my customary rate of $775 per hour for time spent on this case. I am also being reimbursed for reasonable and customary expenses. My compensation is not dependent in any way on the results of the lawsuit or the substance of my testimony.

4. I provided an overview of my background, qualifications and publications in my Declaration in support of Juniper's motion on the '780 patent, which I incorporate by reference. Additional details of my education and employment history, professional service, patents, publications, and other testimony are set forth in my current curriculum vitae, which can be found here: http://avirubin.com/Avi_Rubins_home_page/Vita.html.

III. MATERIALS CONSIDERED

5. I have considered information from various sources in forming my opinions. In addition to drawing from over two decades of experience in the computer industry, I also have reviewed the following documents: (a) the '494 Patent; (b) the prosecution file history (including IPRs) for the '494 Patent; (c) Finjan's Infringement Contentions (Exhibits F-1 and F-2); (d) Finjan's Motion and supporting exhibits, including the Declaration of Dr. Eric Cole; (e) the deposition transcripts of the Juniper engineers deposed in this matter, as well as Dr. Cole; and (f) the other documents and references cited herein (not limited to the specific excerpt submitted with Juniper's Opposition), including Juniper's source code produced in this matter. I have also reviewed the Declaration of Chandra Nagarajan.

IV. LEGAL STANDARDS

6. I have been advised that patent claims are reviewed from the point of view of a hypothetical person of ordinary skill in the art ("POSITA") at the time of the filing of the patent. In my opinion, a POSITA for the '494 patent would be a person with a Bachelor's degree in computer science or related academic fields and three to four years of additional experience in the field of computer security or equivalent work experience. More education can substitute for work experience, and vice versa (e.g., a PhD without work experience outside of the university setting). In arriving at my opinions in this declaration, I have considered the issues from the perspective of a hypothetical POSITA. This level of skill is approximate and my opinion would not change if a somewhat lower or higher level of skill were adopted. My understanding of the other applicable legal standards is included in my declaration on the '780 patent motion, which I incorporate by reference.

V. STATE OF THE ART

7. In the field of computer security, there are many different ways that a program can determine whether a file is malware. One example is an anti-virus scanner which compares the hash of a file to the hashes contained in a virus database to determine whether the file is one of the viruses in the database. Such anti-virus scanners have been around since at least the mid-1980s.

8. Another example of malware detection is "static" analysis, where the features and characteristics of a file are analyzed without actually executing the code and checking, for example, for specific byte sequences or other patterns in the code, or using heuristic analysis that identifies features such as if the file has an invalid digital signature, has a high entropy, or has no publisher information. Static analyzers have also existed since at least the mid-1980s.

9. Another example of malware detection is "dynamic" analysis or "emulation," which means that the file is actually executed or "detonated" in a safe, simulated environment known as a "sandbox" that determines what the file actually does when it is executed. Dynamic analysis has existed since at least the early 1990s.

10. By the early 1990s, there was already a mature anti-malware community that had developed numerous ways to try to protect against malware, including each of the strategies described above. As of that time, one way to implement a system that did anti-virus or static analysis detection

was through the use of a scanner. By that time, scanners performing static analysis that used parsing techniques[1] to decompose code and flag specific extracted components were well known in the art. *See, e.g.*, Ex. 19 at p. 5 ("Hexadecimal patterns may be used to detect the presence of the virus with . . . a dedicated virus scanner."; Ex. 3 at p. 3 ("Scanning techniques are further complicated by the fact that the [polymorphic] viruses do not have any scan strings in common even if their structure remains constant. When polymorphic technology improved, statistical analysis of the opcodes was used. . . . The next shift many scanners are presently experiencing is away from known virus only detection to detection of unknown viruses. The method of choice is *heuristics*. . . . This is most often done by looking for a pattern of certain code fragments that occur most often in viruses and hopefully not in *bona fide* programs."); Ex. 17 at p. 2 ("So, what is a scanner? Simplifying, a scanner is a program which searches files and disk sectors for byte sequences specific to this or that known virus. Those byte sequences are often called virus signatures.").

11. No scanner, however, is perfect. Even decades ago it was common practice to use multiple scanners to try to detect malware. *See, e.g.*, Ex. 19 at p. 20 ("Use several scanners from dissimilar sources. The more search data that is available the better . . . . No single virus-scanner provide 100 percent protection!"). No scanner is perfect in part because each scanner's list of commands or command patterns for which it searches is different, since there is no industry standard or commonly accepted list of what is "suspicious." The problem is illustrated by the '494 Patent's own list of examples of potentially hostile operations, which includes things like "READ a file." *See* '194 Patent at 5:59. Depending on context, reading a file could be benign (e.g., a command to read an operating system registry file) or it could be hostile (e.g., a command to read an isolated plain text file). Since different developers of scanners look for different things as associated with malware, it is beneficial to use multiple scanners. Indeed, this precise benefit is touted by OPSWAT for its

[1] At the lowest level, computer instructions are in numerical form called "machine code" that is only legible to a computer, which are directly executable by a computer's central processing unit. A "disassembler" can be used to abstract the machine code into a low-level but human-readable language known as "assembly language." Instructions written in assembly language, however, are still generally difficult for a human to understand, so for analysis purposes, programs are often parsed to "decompile" the machine code into source code, which is the more human-understandable language in which a computer program was originally written. Parsing involves identifying and separating various parts of a program into the instructions and parameters that make up the program.

Metadefender product, [redacted], which packages multiple different analysis engines together. *See* https://www.opswat.com/products/metadefender ("MetaDefender multi-scanning uses multiple anti-malware engines to provide superior detection rates of known and unknown threats . . . . MetaDefender customers can simultaneously leverage the combined threat prevention capabilities of more than 30 anti-malware engines using both signature heuristic scanning and machine learning capabilities.").

12. Some of these early scanners were intended to be loaded onto an end user's personal computer. But scanning for malware at the client computer was not necessary, and it was also well-known in the art to use scanners at an intermediate node in the network, such as a network gateway or firewall. *See, e.g.*, U.S. Patent No. 5,623,600 at 2:61-3:3 ("The gateway node of the present invention is particularly advantageous because the impact of using the FTP proxy server and SMTP proxy server for the detection of viruses is minimized because only the files leaving or entering the network are evaluated for the presence of viruses and all other 'intra' network traffic is unaffected."); Ex. 5 at p. 13 ("One possibility is to use it as a type of firewall for programs entering a protected network.").

13. Prior art scanners would often rely on databases of known malware signatures. *See, e.g.*, Ex. 17 at p. 1 ("Also, as the number of viruses grows, so does the size of scanner or its database."); Ex. 5 at p. 3 ("Usually, a scanner uses a database of virus identification information which enable it to detect all viruses previously analyzed."). As was well-known in the art, databases were commonly managed using a database manager. *See, e.g.*, Ex. 18 at p. 6 ("A *database management system* (DBMS) is a set of programs used to define, administer, and process databases . . . . There are many DBMSs on the market today."). As Dr. Cole correctly notes, database managers were well-known by the time of the '494 Patent. Ex. 4 at 140:14-20 ("Q. Based on your 30 years of experience in this field you recognize that database managers existed prior to the '494 patent; correct? . . . THE WITNESS: I believe my testimony was databases existed and I was pretty sure that database managers existed prior also.").

VI. CLAIM CONSTRUCTION

14. My understanding is that the parties have agreed to the following constructions, which I have applied in my analysis:

Claim Term	Agreed Construction
Downloadable	"an executable application program, which is downloaded from a source computer and run on the destination computer"
database	"a collection of interrelated data organized according to a database schema to serve one or more applications"

15. With regard to the meaning of the construction of "database," I understand that Finjan's expert in IPR2016-00159 has stated that a POSITA would understand a "database schema" to mean a "description of a database to a database management system (DBMS) in the language provided by the DBMS." IPR2016-00159, Paper No. 17 at p. 35 (citing Medvidovic declaration at ¶ 60). Finjan's IPR expert went on to explain that "[a]n example of such a database language is structured query language (SQL), which is used in querying, updating, and managing relational databases-the de facto standard for database products." *Id*. Finjan also argued that the prior art presented in the IPR did not satisfy the "database" element because it did not store the data "in the form of a table." IPR2016-00159, Paper No. 17 (Patent Owner's Response) at p. 36; *See* Ex. 6[2] (defining "table" as "the fundamental structure of data storage and the display in which data items are linked by the relations formed by placing the items in rows and columns."). I understand that Finjan's expert in this case, Dr. Cole, confirmed that he agrees with Finjan's IPR expert as to the meaning of "database schema." Ex. 4 at 115:17-116:1. For purposes of narrowing the disputes in this case, I have adopted the meaning of "database schema" articulated by Finjan's experts Dr. Medvidovic and Dr. Cole.

16. I understand that Juniper is proposing the following additional constructions:

Claim Term	Juniper's Proposed Construction	Primary Reason/ Support For Construction
scanner	a static analyzer that uses parsing techniques to decompose the code	Finjan's disclaimer of dynamic analyzers in IPR involving Swimmer, and statements made to save patent from 35 U.S.C. § 101 challenge.
list of suspicious computer	list of all operations that could ever be deemed potentially hostile	PTAB's construction in IPR2015-01892 (Paper 58 at 12 (Mar. 15,

[2] Finjan relied on this dictionary as extrinsic evidence for the construction of terms for patents in the same family as the '494 Patent, so Finjan cannot be heard to complain about this dictionary. *See* Dkt. No. 115 at p. 5 (Finjan citing Webster's New World Dictionary of Computer Terms).

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.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.