UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
WACO DIVISION

Webroot, Inc. and Open Text, Inc., ***Plaintiffs,*** **v.** **CrowdStrike, Inc. and CrowdStrike Holdings, Inc.,** ***Defendants.***	**Civil Action No. 6:22-cv-00241-ADA** **JURY TRIAL DEMANDED**
CrowdStrike, Inc. ***Counterclaim-Plaintiff,*** **v.** **Webroot, Inc. and Open Text, Inc.,** ***Counterclaim-Defendants.***	

FIRST AMENDED COMPLAINT FOR PATENT INFRINGEMENT

Plaintiffs Open Text, Inc. ("OpenText") and Webroot, Inc. ("Webroot") (collectively "Plaintiffs") allege against Defendants CrowdStrike, Inc. and CrowdStrike Holdings, Inc. (collectively "CrowdStrike" or "Defendants") the following:

1. This case involves patented technologies that helped to revolutionize, and have become widely adopted in, the fields of malware detection, network security, and endpoint protection. Endpoint protection involves securing endpoints or entry points of end-user devices (*e.g.*, desktops, laptops, mobile devices, etc.) on a network or in a cloud from cybersecurity threats, like malware.

2. Before Plaintiffs' patented technologies, security platforms typically relied on

signatures (*i.e.*, unique identifiers) of computer objects (*e.g.*, computer programs) that were analyzed and identified as "bad" by teams of threat researchers. This approach required antivirus companies to employ hundreds to thousands of threat analysts to review individual programs and determine if they posed a threat.

3. The "bad" programs identified by researchers were compiled into a library and uploaded to an antivirus software program installed on each endpoint device. To detect threats, a resource intensive "virus scan" of each endpoint device was conducted. These virus scans could take hours to complete and substantially impact productivity and performance.

4. Despite substantial investments in resources and time, the conventional systems still were unable to identify and prevent emerging ("zero-day") threats from new or unknown malware. New threats persisted and were free to wreak havoc until a team of threat analysts could identify each one and upload these newly identified threats to an update of the "bad" program library. The updated "bad" program library, including signatures to identify new threats as well as old, then had to be disseminated to all of the endpoint computers, which required time and resource consuming downloads of the entire signature library to every computer each time an update was provided.

5. By the early-to-mid 2000s, new threats escalated as network connectivity became widespread, and programs that mutate slightly with each new copy (polymorphic programs) appeared. These events, and others, rendered the traditional signature-based virus scan systems ineffective for these modern environments.

6. Plaintiffs' patented technology helped transform the way malware detection and network security is conducted, reducing and often even eliminating the shortcomings that plagued signature-based security products that relied on human analysts.

7. Instead of relying on human analysts, Plaintiffs' patented technology enabled the automatic and real-time analysis, identification, and neutralization of previously unknown threats, including new and emerging malware, as well as advanced polymorphic programs.

8. For example, Plaintiffs' patented technology uses information about the computer objects being executed—including, for example, information about the object's behavior and information collected from across a network—along with machine learning technology and novel system architectures—to provide security systems that are effective in identifying and blocking new security threats in real-time in real-world, commercial systems.

9. Plaintiffs' patented technology further includes new methods of "on execution" malware analysis; new architectures that efficiently and effectively distribute workloads across the network; new forensic techniques that enable fast, efficient, and accurate analysis of malware attacks; and new advanced memory scanning techniques.

10. Plaintiffs' patented technology makes security software, platforms, and appliances better at detecting malware by, for example, reducing false positives/negatives and enabling the identification and mitigation of new and emerging threats in near real-time. These improvements are accomplished while at the same time reducing the resource demands on the endpoint computers (*e.g.*, not requiring downloading and using full signature databases and time-consuming virus scans).

11. Plaintiff Webroot has implemented this technology in its security products like Webroot SecureAnywhere AntiVirus, which identifies and neutralizes unknown and undesirable computer objects in the wild in real-time.

12. Over the years, Plaintiff Webroot has also received numerous accolades and awards for its products and services. For example, Webroot has received 22 PC Magazine Editor's Choice

providing services in connection with the Accused Products including installing, maintaining, supporting, operating, providing instructions, and/or advertising CrowdStrike's Falcon Platform within this District. End-users and partner customers infringe the Asserted Patents by installing and operating Falcon Platform software, which performs the claimed methods in the Asserted Patents within this District.

41. Defendants encourage and induce their customers of the Accused Products to perform the methods claimed in the Asserted Patents. For example, CrowdStrike makes its security services available on its website, widely advertises those services, provides applications that allow partners and users to access those services, provides instructions for installing, and maintaining those products, and provides technical support to users. (*See* WBR_CSK000001 (https://www.crowdstrike.com/contact-us/); https://www.crowdstrike.com/contact-support/ (redirect to same).)

42. CrowdStrike further encourages and induces its customers to use the infringing Falcon Platform by providing directions for and encouraging the "CrowdStrike Falcon agent" to be installed on individual endpoint computers (*see* WBR_CSK000090 (https://www.crowdstrike.com/blog/tech-center/install-falcon-sensor/)), which offers evaluation, installation, configuration, customization and development of the Falcon Platform.

43. Defendants also contribute to the infringement of its customers and end users of the Accused Products by offering within the United States or importing into the United States the Accused Products, which are for use in practicing, and under normal operation practice, one or more of the methods claimed in the Asserted Patents, constituting a material part of the inventions claimed, and not a staple article or commodity of commerce suitable for substantial non-infringing uses. Indeed, as shown herein, the Accused Products and the example functionality described

below have no substantial non-infringing uses but are specifically designed to practice the methods claimed in the Asserted Patents.

44. Defendants' infringement adversely impacts Plaintiffs and their employees who live in this District, as well as Plaintiffs' partners and customers who live and work in and around this District. On information and belief, Defendants actively target and offer Accused Products to customers served by Plaintiffs, including in particular customers/end-users in this District.

PLAINTIFFS' PATENTED INNOVATIONS

45. Plaintiff Webroot, and its predecessors were all pioneers and leading innovators in developing and providing modern end point security protection, including "community-based" signatureless threat detection process using AI-driven behavior analysis across the entire network to provide "zero-day" protection against unknown threats.

46. The Asserted Patents discussed below capture technology, features, and processes that reflect these innovations, and improve on traditional anti-Malware and network security systems.

Advanced Malware Detection Patents
U.S. Patent Nos. 8,418,250, 8,726,389, and 8,763,123

47. The '250, '389, and '123 Patents are part of the same patent family and generally disclose and claim systems and processes related to real-time and advanced classification techniques for as-yet unknown malware. These patents are collectively known as the "Advanced Malware Detection" Patents. Plaintiff Webroot owns by assignment the entire right, title, and interest in and to the '250, '389, and '123 Patents. Webroot has granted Plaintiff OpenText an exclusive license to the '250, '389, and '123 Patents.

48. The '250 Patent is entitled "Methods and Apparatus for Dealing with Malware," was filed on June 30, 2006, and was duly and legally issued by the United States Patent and Trademark Office ("USPTO") on April 9, 2013. The '250 Patent claims priority to Foreign

Application No. 0513375.6 (GB), filed on June 30, 2005. A true and correct copy of the '250 Patent is attached as Exhibit 1.

49. The '389 Patent is also entitled "Methods and Apparatus for Dealing with Malware," was filed on July 8, 2012, and was duly and legally issued by the USPTO on May 13, 2014. The '389 Patent claims priority to the same Foreign Application as the '250 Patent. A true and correct copy of the '389 Patent is attached as Exhibit 2.

50. The '123 Patent is also entitled "Methods and Apparatus for Dealing with Malware," was filed on July 8, 2012, and was duly and legally issued by the USPTO on June 24, 2014. The '123 Patent is a division of the application which issued as the '250 Patent and claims priority to the same Foreign Application as the '250 Patent. A true and correct copy of the '123 Patent is attached as Exhibit 7.

51. Malware detection systems in use at the time the Advanced Malware Detection Patents were filed identified malware by maintaining a database of signatures identifying known bad objects (*i.e.*, malware). The signature for an object was conventionally made by creating a hash or checksum corresponding to the object file, which uniquely identifies that object. The signature of each object was then compared to the database to look up whether it matches known malware.

52. If the signature of the object is not found in the database, it is assumed safe or alternatively, the whole file is sent for further investigation by a human analyst. The process of further investigation was typically carried out manually or "semimanually" by subjecting the file to detailed analysis, for example by emulation or interpretation, which can take days given the human involvement that is typically required. (*See, e.g.*, Exhibit 2, '389 Patent, 2:9-17.)

53. This approach had significant drawbacks, including that it required considerable

effort by the providers of such systems to identify and analyze new malware and generate signatures of objects that are found to be bad after human analysis. Large vendors of anti-malware packages typically employed *thousands* of human analysts to identify and analyze objects and keep the database of signatures of bad objects reasonably up to date.

54. However, as the volume of network traffic increases, the task of keeping up with identifying suspect objects and investigating whether or not they are bad becomes practically impossible. (*Id.*) It can take days to subject a suspicious file to detailed analysis given the human involvement, and a considerable period of time elapses before a new file is classified as safe or as malware. Thus, the human analysis introduces a time delay where users are exposed and unprotected from the risks posed by previously unidentified malware. (*See* Exhibit 2, '389 Patent, 2:9-23, 2:63-67.)

55. By contrast, the methods and systems disclosed and claimed in the '250,'389, and '123 Patents perform automatic, sophisticated review (*e.g.*, "pattern analysis") of the actual attributes of a software object or process and the behavior engaged in by, or associated with, that object or process on computers connected to a network.

56. This review enables a determination of "the nature of the object," (*e.g.*, whether it is malicious or not based on review of the object, its behaviors or the activities associated with the object), without requiring a detailed manual analysis of the code of the object itself or relying exclusively on whether it has a signature that matches an extensive database of known malicious "signatures." (*See* Exhibit 2, '389 Patent, 3:14-24; Exhibit 1, '250 Patent, 3:7-18.) This provides a significant improvement to the operation of the computer network because monitoring behavior or other information about the object or process, rather than code or signature matching, allows the system to rapidly determine the nature of the object (*e.g.*, malware), without requiring a detailed

manual analysis of the code of the object itself as in conventional anti-virus software. (*See* Exhibit 1, '250 Patent, 3:11-18.)

57. The approaches in the Advanced Malware Detection Patents are generally focused on receiving *information about the behavior* of objects or processes on remote computers at a base computer. This information is analyzed automatically by, for example, mapping the behavior and attributes of objects known across the community in order to identify suspicious behavior and to identify malware at an early stage. This approach allows, among other advantages, the number of human analysts needed to be massively reduced. It also improves the computer network by reducing the latency involved with identifying new threats and responding to objects exhibiting new, potentially malevolent behavior. (WBRT001252 at WBRT001663-1664, '250 Patent Prosecution History, 2010-09-07 Amendment at 16-17.)

58. Each of the claimed inventions of the Advanced Malware Detection Patents is necessarily rooted in computer technology—in other words, the identification of malicious computer code in computer networks is fundamentally and inextricably a problem experienced with computer technology and networks—and addresses this fundamental computer technology problem with a computer technology solution. Furthermore, the Advanced Malware Detection Patents improve the technical functioning of the computer network using techniques—such as analyzing behavioral information about or associated with computer objects and processes—to improve network security by identifying malware more quickly and with less resources. These technical improvements address identified weaknesses in conventional systems and processes. (*See, e.g.*, Exhibit 1, '250 Patent, 2:5-3:18.)

59. In particular, the '250 Patent describes and claims methods and systems that include receiving *behavioral data about or associated with a computer object* from remote computers on

which the object or similar objects are stored; comparing in a base computer the data about the computer object received from the remote computers; and, classifying the computer object as malware on the basis of said comparison if the data indicates the computer object is malware. In effect, this process builds a central picture of objects and their interrelationships and activities across the entire community and allows automation of the process of identifying malware by aggregating and comparing the activity of objects running across the community (*i.e.*, on multiple remote computers).

60. The ’250 Patent further provides that a mask is automatically generated for an object that defines “acceptable behavior” for the object. The operation of the computer object is then monitored and if the actual monitored behavior extends beyond that permitted by the mask, the object is disallowed from running and reclassified as malware.

61. The claimed methods and systems of the ’250 Patent constitute technical improvements over the traditional anti-malware systems and provide numerous advantages to computer systems and the process of detecting malware. In addition to the advantages set forth above, the methods and systems claimed in the ’250 Patent provide additional advantages in dealing with objects that do not initially exhibit suspicious behavior, but later start to exhibit malevolent behavior. Traditional malware systems could only mark a computer object as good or bad (*i.e.*, a binary decision), and did so by examining the signature of the object itself against a database of “known bad” signatures. This approach does not permit the system to automatically deal with the case where an object does not initially exhibit suspicious behavior but starts to exhibit malevolent behavior in the future.

62. By contrast, the ’250 Patent improves these systems by generating an appropriate behavior mask for the object and then continuing to monitor the behavior of the object. If the object

operates out of bounds of the permitted behavior, then an appropriate action is taken, such as disallowing the computer object from running and reclassifying the object as malware. Thus, the systems and methods described and claimed further the operation and security of the network by stopping an object from running and changing the classification of an object in real-time when unacceptable behavior is identified. (*See* Exhibit 1, '250 Patent, 3:47-50; 4:19-30.)

63. Furthermore, the methods and systems claimed in the '250 Patent, including generating a "mask" of acceptable behavior, allowing an object to run, continuing to monitor the object, and disallowing/reclassifying the object if the behavior extends beyond that permitted by the mask, are not routine or conventional. For example, while a "safe," mask-permitted version of notepad.exe "would not be expected to perform a wide variety of events, such as transmitting data to another computer or running other programs or running other programs" a "modified" and potentially "malevolent" version of notepad.exe could perform those unexpected events. (*See* Exhibit 1, '250 Patent, 11:27-41.) Unlike traditional malware systems that would have already made a binary determination that the notepad.exe object is safe, the methods and systems of the '250 Patent re-classify that version of notepad.exe as malware when its behavior becomes unexpected and "extends beyond that permitted by the mask." (*Id.* at 4:19-30.)

64. The applicants provided another example illustrating the unconventional nature and technical advantages and improvements, offered by the claimed systems and methods during prosecution:

> As an example, suppose a new version of Internet Explorer appeared. This could be a legitimate update to Internet Explorer released by Microsoft or alternatively it could be a file infected with a virus. In the prior art, the new object would have an unknown signature, so an in-house analyst would laboriously analyse the new object and determine whether or not it was safe. Whilst this analysis is carried out, the object would either be blocked, which would cause huge inconvenience to users of the new object, or allowed to run, in which case there is a risk of the object performing malevolent acts. In contrast, the present invention would collect data at

> the base computer from remote computers running the new version of Internet Explorer. Using the information collected, the system could determine that the new object purports to be a new version of Internet Explorer. However, it may not be apparent at this point whether or not the new object is capable of malevolent behaviour. In this scenario the present invention generates an appropriate behavioural mask for the object, e.g. by using a profile of behaviour of previous versions of Internet Explorer that are known not to be malware, or by using a profile for the behaviour appropriate for a web browser. The remote computers are allowed to let the new version run whilst monitoring its behaviour against the mask. The instant the new object exhibits some new, malevolent behaviour, this can be stopped at the remote computer, as well as being flagged to the base computer and used at the base computer to change the classification of the object. Thus, the present invention allows an instant response to an object changing its behaviour to exhibit malevolent behaviour in the future. (*See* WBRT001252 at WBRT001665-1666, '250 Patent Prosecution History, 2010-09-07 Amendment at 18, 19.)

65. Similarly, the '389 Patent describes and claims deploying an unconventional "event" based model that classifies a particular object as malicious or safe by analyzing real-time data sent by remote computers on the events, or actions, that a particular software "object," and other objects deemed similar to it, initiate or perform on those computers. (*See* Exhibit 2, '389 Patent, 3:14-55.) This information is collected from across the network, correlated and used for subsequent comparisons to new or unknown computer objects to identify relationships between the correlated data and the new or unknown computer objects. The objects may be classified as malware based on this comparison.

66. Through continuous aggregate analysis of events involving computer objects as they occur across network endpoints, the methods and systems described and claimed in the '389 Patent maintain up-to-date information about computer objects (including malicious objects) seen across the network, identify relationships between those previously identified objects and any new or unknown objects, and make malware determinations based on those relationships. "For example, a new object that purports to be a version of notepad.exe can have its behavior compared with the behav[io]r of one or more other objects that are also known as notepad.exe … In this way,

new patterns of behav[io]r can be identified for the new object." (*Id.* at 10:58-65.)

67. The methods and systems described and claimed in the '389 Patent can rapidly determine "the nature of the object," (*e.g.*, whether it is malicious or not) based on information such as the behavior of the object or effects the object has, without requiring "detailed analysis of the object itself as such" (manually reviewing the object's code) or reliance on matching an extensive database of known malicious "signatures." (*Id.* at 3:14-24; Exhibit 1, '250 Patent, 3:7-18.)

68. The '123 Patent generally is directed to the real-time and cloud-based determination of whether a particular computer is vulnerable to a particular malware process. The '123 Patent bases this determination on the security products it has installed and the products' vulnerabilities.

69. The methods of the '123 Patent therefore are directed to solutions to specific problems that arose in, and are necessarily rooted in, computer technology. Prior art models performed simple comparisons between the versions of software installed (*e.g.*, an internet browser) on a given computer and their "known vulnerabilities" in a static database. (Exhibit 7, '123 Patent, 2:52-57.) These models were limited in that they only performed simple comparisons of what software applications a particular computer had installed (*e.g.*, a particular internet browser), to the vulnerabilities with which that application was installed in a static database.

70. By contrast, the methods described in the '123 Patent target vulnerabilities in an end user's computer caused by blind spots in its locally installed security products. The described methods then identify those vulnerabilities in real time by marshalling a continuously updating "community database." The "community database" holds historical information, continually updating vulnerability data from "many, possibly millions, of users' computers," including the "configuration of security products" and operating systems each of those millions of computers

had installed, and which such combinations were "susceptible or vulnerable to any particular malware object." (Exhibit 7, '123 Patent 16:22-50, 17:5-15.)

71. To reduce "the data quantity for storage and transmission," the community database receives this configuration information in the form of a highly compressed "signature or key" that encapsulates all "the details of all local security products, versions, signature files, firewall settings, etc." (Exhibit 7, '123 Patent, 16:24-31.)

72. Storing "keys" in such a form enables quick searches of the community database to find the configuration "key" of any particular computer and associate that key with the vulnerabilities to which that computer is exposed. (Exhibit 7, '123 Patent, 16:22-39.) The search, diagnosis and resulting message to the user (*e.g.*, "directing the user to a website" from which she can download a particular security update) are performed "virtually in real-time." (Exhibit 7, '123 Patent, 17:5-15.)

73. The Advanced Malware Detection Patents provide systems and methods that necessarily address issues unique to computer networks and computer network operation; namely the identification of "bad" software (*e.g.*, malware, viruses, etc.). These patents all provide unique network security enhancement that solves the technical problem of rapidly identifying newly arising and emerging malware by reviewing information about the object and processes (*e.g.*, the behaviors and events associated with software objects and processes running on computers within the network).

74. The systems and methods claimed in the Advanced Malware Detection Patents improve the operation of computer networks by identifying malicious objects in real-time and taking action to remove or eliminate the threat posed by the malware object or process once it has been identified. The claimed inventions in these patents provide a technological solution to a

technological problem—the inability of conventional code or signature matching solutions to identify new or unknown malware objects or processes at or near the runtime of the objects or processes themselves without the extensive delay and resource use associated with traditional systems.

Forensic Visibility Patents
U.S. Patent No. 9,578,045 and U.S. Patent No. 10,257,224

75. The ’045 and ’224 Patents are part of the same patent family and are each generally directed to providing forensic visibility into computing devices in a communication network by analyzing network events and creating audit trails. Plaintiff Webroot owns by assignment the entire right, title, and interest in and to the ’045 and ‘224 Patents. Webroot has granted OpenText an exclusive license to the ’045 and ’224 Patents.

76. The ’045 Patent is entitled “Method and Apparatus for Providing Forensic Visibility into Systems and Networks,” was filed on May 5, 2014, and was duly and legally issued by the USPTO on February 21, 2017. The ’045 Patent claims priority to provisional application 61/819,470 filed on May 3, 2013. A true and correct copy of the ’045 Patent is attached as Exhibit 3.

77. The ’224 Patent is also entitled “Method and Apparatus for Providing Forensic Visibility into Systems and Networks,” was filed on February 20, 2017, and was duly and legally issued by the USPTO on April 9, 2019. The ’224 Patent claims priority to the ’045 Patent and also to provisional application 61/819,470 filed on May 3, 2013. A true and correct copy of the ’224 Patent is attached as Exhibit 4.

78. The ’045 and ’224 Patents describe and claim inventive and patentable subject matter that significantly improves on traditional network forensic tools used to discover or identify security issues on computer networks. Network forensics generally relates to intercepting and

analyzing network events to discover the source of security attacks. (*See* Exhibit 3, '045 Patent, 1:22-24; Exhibit 4, '224 Patent, 1:24-26.)

79. The '045 and '224 Patents improved on these prior art network forensic tools by providing a technical solution to a technical problem experienced by computer networks and computer network operation. Unlike traditional network forensic tools, these patents create forensic visibility into the computing devices on the communication network to identify malware or other security issues in operation of those devices. (*See* Exhibit 3, '045 Patent, 2:36-38; Exhibit 4, '224 Patent, 2:38-40.)

80. In particular, the Forensic Visibility Patents improve network security by gathering an "event," generating "contextual state information," obtaining a "global perspective" for the event in comparison to other events, and generating/transmitting an "event line" that includes information for the event. (*See* Exhibit 3, '045 Patent, cl. 1; Exhibit 4, '224 Patent, cl. 1.) The described and claimed systems and methods intercept network events, create audit trails, or contextual states, for each individual event by correlating the event to objects such as their originating processes, devices, and/or users, and establishing a global perspective of the objects. The claimed systems and methods of the Forensic Visibility Patents address an identified weakness in conventional systems and processes; namely the ability to monitor, capture and/or analyze what is occurring *at* computing devices on a computer network, thereby providing an improved way to address the technical problem of discovering security attacks or security problems within a computer network.

81. In addition to analyzing the behavior of an object to identify those that are potentially malicious, malware detection is further improved by understanding the context of the event and computer objects of interest. (*See* Exhibit 3, '045 Patent, 2:39-45 ("The system filters

may be built upon the same or similar technology related to behavior monitoring and collection, as discussed in U.S. application Ser. No. 13/372,375 filed Feb. 13, 2012, 'Methods and Apparatus for Dealing with Malware.'").) In particular, in many cases a potentially malicious object is identified by the system as a result of other events that provide information as to whether the code is malicious. For example, if an object or event under investigation originated from an object or event that is known to be malicious or have malicious behaviors or characteristics, the presence of the known, malicious object provides a further indication that the potentially malicious object or event is malicious as well.

82. The patents further explain that in addition to context information, the systems and techniques can also use information from the network to obtain a global perspective of the network operation. The combination of contextual information and global perspective enables detection of new zero-day threats, including objects created from objects (or similar objects) that have been identified previously as malicious. Indeed, in the context of modern computers and network systems that generate tens of millions of events every minute, the use of a global perspective and contextual information to correlate an event or object under investigation with prior, related events and objects—including the originating object—significantly improves the ability of the system to identify potential threats.

83. The patents further disclose technical improvements to forensic systems by "assembling" or "generating" an "event line" based on the contextual information—including the correlation to the originating object—and global perspective. (*See, e.g.*, Exhibit 3, '045 Patent, 9:50-58.) The generation of the event line makes it easier for end users to "identify events, and/or instances of malware, that require more immediate attention"—thereby improving the accuracy and efficiency of identifying additional malicious code, as well as enabling administrators to more

readily analyze malware, assess vulnerabilities, and correct damage done by the originating objects (and other objects in the event chain). (*See* Exhibit 3, '045 Patent, 9:45-49.) The generation and use of an event line itself was, at the time, an unconventional way in which event information, contextual state information, and global perspectives are generated, communicated, and/or potentially displayed to, and interacted with by, an administrator or end user.

84. Thus, the '224 and '045 Patents describe and claim systems and methods that provide technical advantages and improvements over traditional network security and forensic systems, including more efficient and accurate identification of malware (*e.g.*, the contextual and global perspective information reduced false negative and positives for malware detection). The patented systems and methods also improved the identification of other malware (and corresponding events) that might otherwise go undetected in prior systems, thereby improving system performance and reducing the number of resources required.

85. Indeed, the patented systems and methods provide end-to-end forensic visibility into event occurrences across a networked environment and from the bottom of the stack to the top, thereby improving upon conventional network forensic products. (*See* Exhibit 3, '045 Patent, 2:31-38, 3:49-55; Exhibit 4, '224 Patent, 2:33-40, 3:52-59; *see also* Exhibit 3, '045 Patent, 4:36-41; Exhibit 4, '224 Patent, 4:39-44.)

86. Applicant further explained during prosecution how the generation of contextual state information and obtaining a global perspective—including for objects and events other than those that were detected, such as the originating object—are unconventional steps in the areas of malware detection and network forensics. For example, Applicant explained how the described systems and methods improves the system performance of computing devices:

> In this case, the claimed invention provides for determining correlations between events and objects and creating an audit trail for each individual

> event. For example, a context analyzer may correlate an actor, victim, and/or event type to one or more originating processes, devices, and users. After the analysis is complete, a sensor agent may use the correlated data to generate a global perspective for each event such that an administrator is able to forensically track back any event which occurs to what triggered it. Thus, the global perspective represents a drastic transformation of raw event data into a comprehensive, system-wide forensic audit trail. (WBRT005145 at WBRT005274-5275, '045 Patent Prosecution History, 2016-03-16 Amendment at 11-12.)
>
> In this case, examples of the claimed systems and methods provide low level system filters which intercept system events "in a manner such that the operation of the system filter does not impact system performance." Specification, ¶ [0008]. For example, on an average system, because tens of millions of events take place every minute, the noise ratio can prevent forensic solutions from being able to provide sufficient value to the end consumer of their data due to the inability to quickly find important events. A product which impacts system performance will have considerably diminished value to an administrator and can negatively affect the results of an analysis undertaken. Examples of the present systems and methods address this shortcoming by providing a system filter that substantially improves the system performance of the computing devices in the system. (*See* WBRT005145 at WBRT005275, '045 Patent Prosecution History, 2016-03-16 Amendment at 12.)

87. During prosecution, Applicant further explained how the claims are directed to solving a technical problem and a specific improvement in computer functionality relating to computer security:

> ***[T]he claims are directed to solving a technical problem***. Typically, network forensic systems use network forensic tools (e.g., network sniffers and packet capture tools) to detect and capture information associated with communication sessions. Although such network forensic tools are operable to passively collect network traffic, the tools reside at a network edge (e.g., outside of a system or hosts). As a result, the network forensic tools have no ability to obtain useful information within a host or to establish any sort of context from within a host that is generating and/or receiving network events. To address this, aspects of the present disclosure enable methods for providing forensic visibility into systems and networks. For example, a local aggregator/interpreter, context analyzer and sensor agent may provide visibility into occurrences across an environment to ensure that a user (e.g., an administrator) is aware of any system change and data communications in and out of the computing devices residing on the network. During this process, identified events may be correlated to

> objects, thus creating an audit trial [sic] for each individual event. (*See* WBRT005145 at WBRT005272-5273, '045 Patent Prosecution History, 2016-03-16 Amendment at 9-10 (emphasis added).)
>
> Here, ***the claims are directed to a specific improvement in computer functionality relating to computer security, and more specifically to providing end-to-end visibility of events within a system and/or network***. (*See* WBRT006334 at WBRT006791-6792, '224 Patent Prosecution History, 2018-08-29 Amendment at 10-11 (citing '224 Patent specification) (emphasis added).)
>
> The Specification subsequently discusses a variety of ways in which the claimed subject matter solves the above-described problem. For example: "It is, therefore, one aspect of the present disclosure to provide a system and method whereby events occurring within a computing device are captured and additional context and a global perspective is provided for each capture event. For example, a sensor agent may provide visibility into occurrences across an environment, such as a networked environment, to ensure that an administrator is aware of any system changes and data communication in and out of computing devices residing on the network." (*See* WBRT006334 at WBRT006793-6794, '224 Patent Prosecution History, 2018-08-29 Amendment at 11-12 (citing '224 Patent specification).)

88. In response to these arguments, the Examiner withdrew a rejection based on 35 U.S.C. §101 and allowed the claims of the Forensic Visibility Patents to issue. As recognized by the USPTO Examiner, the claimed inventions of the '045 and '224 Patents provide a technical solution to the technical problem of forensic visibility regarding events in a computer network.

US. Patent No. 10,284,591

89. The '591 Patent is entitled "Detecting and Preventing Execution of Software Exploits," was filed on January 27, 2015, and was duly and legally issued by the USPTO on May 7, 2019. The '591 Patent claims priority to provisional application 61/931,772 filed January 27, 2014. A true and correct copy of the '591 Patent is attached as Exhibit 5. Plaintiff Webroot owns by assignment the entire right, title, and interest in and to the '591 Patent. Webroot has granted Plaintiff OpenText an exclusive license to the '591 Patent.

90. The '591 Patent describes and claims an "anti-exploit" technique to prevent

undesirable software and/or other computer exploits from executing. (*See* Exhibit 5, '591 Patent, 1:13-28, 1:32-33.) Computer "exploits" include code, software, data, or commands that take advantage of a bug, glitch, or vulnerability in a computer system. To accomplish this goal, the novel anti-exploit techniques described and claimed in the '591 Patent monitor memory space of a process for execution of functions and performs "stack walk processing" upon invocation of a function in the monitored memory space. (*Id.* at 1:33-39.) During that stack walk processing, a memory check may be performed to detect suspicious behavior. (*Id.*) If the memory check detects certain types of suspicious behavior, an alert may be triggered and that prevents the execution of a payload for the invoked function. (*Id.* at 1:39-48.)

91. The '591 Patent describes and claims unconventional "stack walk processing" techniques for detecting and preventing unwanted software exploits during which memory checks are performed before an address of an originating caller function is reached. The anti-exploit techniques can include performing "[m]emory checks performed during the stack walk processing once an address is reached for an originating caller function." (*Id.* at 8:6-7.) In one embodiment, "memory checks from the lowest level user function of the hooked function down through the address of the originating caller function" may be performed to detect and identify suspicious behavior. (*Id.* at 6:7-11.)

92. The "stack walking" and "memory checks" described and claimed in the '591 Patent are fundamentally rooted in computer technology—in fact, they are processes only performed within a computer context. The techniques described and claimed in the '591 Patent addresses a problem that specifically arises in the realm of computer technology (namely, computer exploit identification) by, *inter alia*, performing memory checks and detection specified behavior during stack walking.

93. The ’591 Patent further describes and claims unconventional techniques that address identified weaknesses in conventional exploit prevention technologies. For example, unlike exploit prevention technologies that try to prevent an exploit from ever starting its own shellcode to execute a malicious payload, the ’591 Patent describes and claims techniques that prevent shellcode from executing a malicious payload even if the shellcode has been started. (*See id.* at 6:24-30; *see also id.* at 7:56-62.) Thus, these unconventional techniques address an identified weakness in conventional exploit prevention systems and provide technical advantages including enhanced security protection, improved detection of potential security exploits, reduction in error rate identifying and marking suspicious behavior (*e.g.*, false positives), and improved usability and interaction for users who are not required to continuously monitor for security exploits. (*Id.* at 2:44-51.) As such, the ’591 Patent describes and claims specific computer-related technological steps to accomplish an improvement in computer security and functionality and is directed to a specific technological solution to a problem unique to computers.

U.S. Patent Nos. 10,599,844 and 11,409,869

94. The ’844 Patent is entitled “Automatic Threat Detection of Executable Files Based on Static Data Analysis,” was filed May 12, 2015, and was duly and legally issued by the USPTO on March 24, 2020. A true and correct copy of the ’844 Patent is attached as Exhibit 6. Plaintiff Webroot owns by assignment the entire right, title, and interest in and to the ’844 Patent. Webroot has granted Plaintiff OpenText an exclusive license to the ’844 Patent.

95. The ’869 Patent, entitled “Automatic Threat Detection of Executable Files Based on Static Data Analysis,” is a continuation of the ’844 Patent and claims priority to the application of the ’844 Patent. The ’869 Patent was filed on February 14, 2020, and was duly and legally issued by the USPTO on August 9, 2022. A true and correct copy of the ’869 Patent is attached as

Exhibit 8. Plaintiff Webroot owns by assignment the entire right, title, and interest in and to the ’869 Patent. Webroot has granted Plaintiff OpenText an exclusive license to the ’869 Patent.

96. The ’844 and ’869 Patents address and improve upon conventional approaches to malware detection in computer networks and computer network operation. Every day, an uncountable number of new executable files are created and distributed across computer networks. Some of those files are unknown, and malicious. It is, thus, vital to accurately and immediately diagnose those files for any potential threat, while also efficiently using resources (*e.g.*, processing power). (*See* Exhibit 6, ’844 Patent, 1:7-13.)

97. Conventional approaches for diagnosing potential malware threats were costly and time consuming, making it difficult to realistically address zero-day threats for all of the files entering a system. These “[a]pproaches to detecting threats typically focus[ed] on finding malicious code blocks within a file and analyzing the behavior of the file.” (*See* Exhibit 6, ’844 Patent, 2:15-17.) Encrypted files would be decrypted then disassembled to extract the code for analysis, typically by traditional anti-virus software based on signature matching. (*Id.* at 2:15-20) If the code was malware, investigating its behavior involved running the code on the system, which put the system at risk. (*Id.* at 2:20-23.)

98. Another approach for protecting against potential threats from unknown executable files involved wavelet decomposition to determine software entropy. (*See* WBRT007115 at WBRT007491, ’844 Patent Prosecution History, April 24, 2019 Applicant Remarks, at 8.) Wavelet decomposition is a process where an original image is decomposed into a sequence of new images, usually called wavelet planes. (*Id.*) In this method, each data file in a set of data files is split into random, non-overlapping file chunks of a fixed length. (*Id.*) Those file chunks are then represented as an entropy time-series, which measures the time it takes for each chunk to

decompose. (*Id.*) Said differently, this approach measured how much time it took a data file to decompose. (*Id.*) Once the file decomposition rate, or entropy time-series, had been calculated, that rate would be compared to decomposition rates of "known bad" files to identify files that contain malware. (*Id.* at WBRT007492.) This process required significant computing resources—typically taking hours to complete—and was not sufficiently accurate in identifying malware.

99. The '844 and '869 Patents significantly improve upon and address shortcomings associated with these prior approaches. The Patents describe and claim methods and systems that detect threats in executable files without the need to decrypt or unpack those executable files by extracting "static data points inside of the executable file without decrypting or executing the file," generating "feature vectors" from those static data points, selectively turning on or off features of the feature vector, and then evaluating the feature vector to determine if the file is malicious. (*See, e.g.*, Exhibit 6, '844 Patent, 1:20-21; cl. 1.) The described systems and methods enable accurate and efficient identification of malware without the need to distinguish between encrypted files and non-encrypted files (*id*. at 6:58-59), thereby significantly increasing efficiency and reducing processing resources required to analyze each potentially malicious computer object. By using this unconventional approach to determine whether a file executable on a computer poses a threat, the '844 and '869 Patents improve on the operation of the computer network associated with the computer by enhancing security, including by increasing detection of new threats, reducing the error rates in identifying suspicious files, and improving efficiency in detecting malicious files. (*See* Exhibit 6, '844 Patent, 2:46-56.)

100. The '844 and '869 Patents describe and claim techniques that employ a learning classifier (*e.g.*, a machine-learning classifier) to determine whether an executable file is malicious, for example by using the classifier to classify data into subgroups and identify and analyze specific

data points to which those subgroups correspond. (*See* Exhibit 6, '844 Patent, 4:33-41, 7:40-8:1.) The described and claimed techniques also selectively turn on or off features for evaluation by the learning classifier. (*See id*. at 7:57-66.) Doing so accelerates analysis and reduces false positives by testing those features of a file likely to be relevant to a determination of its maliciousness. For example, the learning classifier "may detect that the file does not contain 'legal information'," such as "timestamp data, licensing information, copyright information, etc." (*See id.* at 7:66-8:5.) In this example, given the lack of legal protection information in the file, the learning classifier would "adaptively check" the file for additional features that might be indicative of a threat," while "turn[ing] off," and thus not use processing time unnecessarily checking features related to an evaluation of "legal information." (*Id*. at 8:5-10.)

101. Second, the '844 and '869 Patents describe and claim techniques that use character strings extracted from within the executable file to generate a feature vector and then evaluate that feature vector using support vector processing to classify executable files. (*See* Exhibit 6, '844 Patent, 9:2-11.) The classifier provides, for example, the ability to leverage the indicia of "benign" files, which use "meaningful words" in certain data fields, versus "malicious" files, which leave such fields empty or full of "random characters," to build meaningful feature vectors that are analyzed to make faster and more identifications of malware (*See, e.g.*, Exhibit 6, '844 Patent, 9:2-18.)

102. The '844 and '869 Patents are thus directed to specific solutions to problems necessarily rooted in computer technology, namely, the determination whether a file executable on a computer poses a threat. The '844 and '869 Patents improved upon the accuracy and efficiency of malware detection. (*See* Exhibit 6, '844 Patent, 2:15-45.)

103. By using some or all of the unconventional techniques described above to

determine whether a file executable on a computer poses a threat, the '844 and '869 Patents address a problem necessarily involving computers and improve upon the operation of computer networks. In particular, the '844 and '869 Patents achieve a number of technical advantages over conventional approaches to malware detection including, for example:

- enhanced security protection including automatic detection of threats, reduction or minimization of error rates in identification and marking of suspicious behavior or files (*e.g.*, cut down on the number of false positives),
- ability to adapt over time to continuously and quickly detect new threats or potentially unwanted files/applications,
- improved efficiency in detection of malicious files, and
- improved usability and interaction for users by eliminating the need to continuously check for security threats.

(*See* Exhibit 6, '844 Patent, 2:15-57.)

U.S. Patent No. 8,856,505

104. The '505 Patent, entitled "Malware Management Through Kernel Detection During a Boot Sequence," was filed on April 30, 2012, and was duly and legally issued by the USPTO on October 7, 2014. The '505 Patent claims priority to, and is a continuation of, U.S. Patent No. 8,190,868 filed August 7, 2006. A true and correct copy of the '505 Patent is attached as Exhibit 9. Plaintiff Webroot owns by assignment the entire right, title, and interest in and to the '505 Patent.

105. The '505 Patent describes and claims techniques to prevent rootkits, spyware, and/or other undesirable software from executing. (*See* Exhibit 9, '505 Patent, 2:17-29, 4:3-5.). In general, periodic scanning of a computing system often fails to identify pestware, much less stop

pestware from executing. (*Id*. at 1:61-67.) The intervening time between periodic scans permits pestware to cloak and execute, leaving the infected computing system exposed, and ultimately vulnerable to the collection and reporting of information held thereon to the malicious third-party. (*Id*.) And pestware that loads relatively early in the boot sequence may be generally undetectable (*e.g.*, cloaked) when scanned later in the boot sequence or otherwise. (*Id*.)

106. The pestware management techniques of the ’505 Patent mitigates these, and other challenges associated with pestware detection and mitigation, including pestware that executes early in the boot sequence and/or that may be undetectable or cloaked once a computing system runs native applications. (*Id*. at 2:17-29, 3:39-46.) The ’505 Patent describes and claims techniques that can monitor and identify pestware events early in the boot sequence of a computing system. (*Id*. at 3:19-31.) As additional drivers and applications are loaded in the course of the boot sequence, techniques that are described and claimed can monitor the loadings and other events and facilitate management of the pestware. (*Id*. at 4:19-31, 4:53-62). Thus, the pestware management techniques may operate to stop pestware before it cloaks and executes, including stopping pestware which may otherwise be undetectable by conventional, periodic, post-boot-sequence scans.

107. The ’505 Patent describes and claims an unconventional “kernel-level monitor” to facilitate the management of pestware early in the boot sequence of a computing system. (*Id*. at 3:1-14.) The kernel-level monitor “begins early in the boot and operating system loading process,” (*Id*. at 3:25-27), and “is responsible for detecting pestware or pestware activity on a protected computer or system,” (*Id*. at 3:19-21). Attempts by pestware to modify the operating system are monitored and intercepted by the kernel-level monitor, for example, by using “driver hooks and monitoring mechanisms,” which may be placed into the operating system kernel during an early portion of the boot sequence. (*Id*. at 3:32-39, 4: 53-62.) Accordingly, the kernel-level monitor can

be configured to identify changes at the kernel-level (*e.g.*, such as those changes which may be attempted by a rootkit or other malicious component), prior to the time when those changes may become generally undetectable. (*Id*. at 4:53-62, 5:1-12.)

108. The ’505 Patent further describes and claims unconventional techniques that permit scanning of the “registry” during the boot sequence “before most services are loaded and executed.” (*Id*. at 6: 4:10.) Performing this type scanning early in the boot process provides new and unexpected benefits, including providing additional information that may be utilized in the identification of malicious programs. For example, scanning the protected parts of the computer such as the registry early in the boot process enables later “examin[ation] and utiliz[ation]” of that information to, for example, “generate new behavior rules for the kernel-level monitor.” (*Id*. at 6:4-10.)

109. The ’505 Patent describes and claims unconventional techniques that address technical weaknesses in conventional mitigation techniques for pestware. (*Id*. at 1:46-66, 2:1-29.) The ’505 Patent describes and claims techniques that permit pestware management at the kernel level, early in the boot sequence before the malicious code can cause damage or conceal itself. (*Id*. at 2:17-29, 4:63-67, 5:1-12.) Thus, these novel and unconventional techniques provide technical advantages such as enhanced security protection, improved detection of pestware, and adaptability to evolving pestware threats. (*Id*. at 1:46-66, 2:1-29.) As such, the ’505 Patent describes and claims specific computer-related technological steps to accomplish an improvement in computer security and functionality and is directed to a specific technological solution to a problem unique to computers.

Kernel-Level API Monitoring Patents
U.S. Patent Nos. 8,201,243 and 8,719,932

110. The ’243 and ’932 Patents are part of the same patent family. Plaintiff Webroot

owns by assignment the entire right, title, and interest in and to the ’243 and ’932 Patents.

111. The ’243 Patent, entitled “Backwards Researching Activity Indicative of Pestware,” was filed on April 20, 2006, and was duly and legally issued by the USPTO on June 12, 2012. A true and correct copy of the ’243 Patent is attached as Exhibit 10.

112. The ’932 Patent, also entitled “Backwards Researching Activity Indicative of Pestware,” was filed on June 6, 2012, and was duly and legally issued by USPTO on May 6, 2014. The ’932 Patent claims priority to the ’243 Patent. A true and correct copy of the ’932 Patent is attached as Exhibit 11.

113. At the time the ’243 and ’932 Patents were filed, conventional pestware-detection-and-removal solutions struggled to stay ahead of evolving pestware because they relied on “definitions of known pestware to search for and remove files on a protected system.” (Exhibit 10, ’243 Patent, 1:63-65.) “Th[os]e definitions were often slow and cumbersome to create,” and “it was often difficult to locate the pestware in order to create the definitions,” in the first place. (*Id.*at 1:65-67.) Furthermore, relying on definitions made it difficult to distinguish wanted pestware from unwanted pestware. (*Id.* at 1:60-62.)

114. To overcome these and other shortcomings, the ’243 and ’932 Patents describe and claim novel “systems and methods for discovering the source of activity that is indicative of pestware.” (*Id*. at 2:17-19.) For example, the ’243 and ’932 Patents describe and claim the use of a kernel-mode driver to monitor the behavior of processes running on a computer, for example monitoring API calls, to detect pestware. (*Id.* at 4:24-29.) The inventors found that this approach provided technical advantages, such as the ability to “intercept” “when a process (*e.g.*, the pestware process) attempt[ed] to spawn … another pestware process or alter a registry entry, … before it [wa]s carried out by an operating system of the protected computer.” (*Id.* at 5:46-50.)

115. Moreover, the ’243 and ’932 Patents describe and claim techniques that enable the identification of suspected pestware as well as externally networked sources that, for example, may be the source of the pestware on the system. This approach provides benefits, such as identifying the source of malicious malware, which can be used to identify new malware (*e.g.*, based on a known source of malware) as well as to block potential malware based on a known malicious source.

U.S. Patent No. 8,181,244

116. The ’244 Patent, entitled “Backwards Researching Activity Indicative of Pestware,” was filed on April 20, 2006, and was duly and legally issued by USPTO on May 15, 2012. A true and correct copy of the ’244 Patent is attached as Exhibit 12. Plaintiff Webroot owns by assignment the entire right, title, and interest in and to the ’244 Patent.

117. At the time the ’244 Patent was filed, prior-art pestware-detection-and-removal solutions struggled to stay ahead of evolving pestware. Prior-art pestware-detection-and-removal solutions relied on “definitions of known pestware to search for and remove files on a protected system.” (Exhibit 12, ’244 Patent, 1:63-65.) “Th[os]e definitions were often slow and cumbersome to create,” and “it was often difficult to locate the pestware in order to create the definitions,” in the first place. (*Id.* at 1:65-67.) Furthermore, relying on definitions made it difficult to distinguish wanted pestware from unwanted pestware. (*Id.* at 1:60-62.)

118. To overcome these shortcomings, the ’244 Patent introduced novel “systems and methods for discovering the source of activity that is indicative of pestware.” (*Id.* at 2:17-19.) The ’244 Patent employed a kernel-mode driver to monitor the behavior of processes running on a computer, for example monitoring API calls, to detect pestware. (*Id.* at 4:24-29.) Once the ’244 Patent detected “pestware activity,” the ’244 Patent issued a timestamp and assembled a log of

events that occurred at or around that time. (*Id.* at 6:2-9.) Then, using the log of events, the '244 Patent was able to "trac[e] … the pestware activity to an origin of the pestware … associated with the activity." (*Id.* at 8:23-26.) Additionally, by using a kernel-mode driver, the '244 Patent was able to "intercept" "when a process (e.g., the pestware process) attempt[ed] to spawn … another pestware process or alter a registry entry, … before it [wa]s carried out by an operating system of the protected computer." (*Id.* at 5:46-50.)

119. The '244 Patent describes and claims technological solutions to a technological problem. For example, embodiments enable the identification of suspected pestware as well as externally networked sources that, for example, may be the source of the pestware on the system. This approach provides benefits, such as identifying the source of malicious malware, which can be used to identify new malware (*e.g.*, based on a known source of malware) as well as to block potential malware based on a known malicious source.

ACCUSED PRODUCTS

120. CrowdStrike offers, sells, and uses several products that provide and implement malware detection and endpoint protection platforms for individuals and enterprises and incorporate Plaintiffs' patented technologies.

121. Those products include the CrowdStrike Falcon Platform. The Falcon Platform is a cloud-based endpoint protection platform that integrates anti-malware technologies, risk management, and attack forensics to protect remotely connected computers. (*See* WBR_CSK000451 (https://www.crowdstrike.com/endpoint-security-products/); WBR_CSK000134 (CrowdStrike 2021 Annual Report Form 10-K) at 144-148.)

122. CrowdStrike's Falcon Platform is installed on endpoint devices at least by downloading the Falcon agent. (*See* WBR_CSK000090 (https://www.crowdstrike.com/blog/tech-

center/install-falcon-sensor/.) On information and belief, the Falcon Platform operates on multiple devices using the Falcon agent including workstations, desktops, laptops, and other traditional end user computer devices, servers, virtual machines, cloud containers, cloud networks, mobile computer devices such as smartphones, and Internet of Things devices.

123. CrowdStrike's Falcon Platform includes multiple modules or functionalities that are integrated in the Falcon Platform. All of these modules are functionalities of the Falcon Platform and operate on endpoint devices and through the cloud using the "lightweight [Falcon] agent." These modules are part of and can be added to the base Falcon Platform. Examples of these modules are discussed further below.

(*See* WBR_CSK000455 (https://www.crowdstrike.com/endpoint-security-products/falcon-platform/) at 456.)

124. CrowdStrike's Falcon Prevent is a cloud-native Next-Generation Antivirus ("NGAV") software solution that detects and prevents known and unknown malware using tools including machine learning, artificial intelligence, and behavior-based indicators of attack

filed. Defendants had knowledge of the ’224 Patent and of the specific conduct that constitutes infringement of the ’224 Patent at least based on the original Complaint, yet have continued to engage in the infringing activity, including by selling, making, configuring, and installing the Accused Products and performing the accused functionality, and by engaging in activities that constitute inducing infringement and contributory infringement as described above.

292. On information and belief, the infringing actions of each partner, customer, and/or end-user of the Accused Products are attributable to Defendants. For example, on information and belief, Defendants direct and control the activities or actions of their partners or others in connection with the Accused Products by contractual agreement or otherwise requiring partners or others to provide information and instructions to customers who acquire the Accused Products which, when followed, results in infringement. Defendants further direct and control the operation of devices executing the Accused Products by programming the software which, when executed by a customer or end user, perform the claimed method of at least claim 1 of the ’224 Patent.

293. Plaintiffs have suffered and continue to suffer damages, including lost profits, as a result of Defendants’ infringement of the ’224 Patent. Defendants are therefore liable to Plaintiffs under 35 U.S.C. § 284 for damages in an amount that adequately compensates Plaintiffs for Defendants’ infringement, but no less than a reasonable royalty.

294. Plaintiffs will continue to suffer irreparable harm unless this Court preliminarily and permanently enjoins Defendants, their agents, employees, representatives, and all others acting in concert with Defendants from infringing the ’224 Patent.

295. Defendants’ infringement of the ’224 Patent is knowing and willful. Defendants acquired actual knowledge of the ’224 Patent at least when Plaintiffs filed the original Complaint and had constructive knowledge of the ’224 Patent from at least the date Plaintiffs marked their

products with the ’224 Patent and/or provided notice of the ’224 Patent on their website.

296. On information and belief, despite Defendants’ knowledge of the Asserted Patents and Plaintiffs’ patented technology, Defendants made the deliberate decision to sell products and services that they knew infringe these patents. Defendants’ continued infringement of the ’224 Patent with knowledge of the ’224 Patent constitutes willful infringement.

297. Plaintiffs’ allegations of infringement, indirect infringement, and willful infringement with respect to this patent are further set forth in Plaintiffs’ Disclosure of Asserted Claims and Preliminary Infringement Contentions Pursuant to the Court’s Standing Order Governing Patent Proceedings served July 12, 2022.

FIFTH CAUSE OF ACTION
(INFRINGEMENT OF THE ’591 PATENT)

298. Plaintiffs reallege and incorporate by reference the allegations of the preceding paragraphs of this Complaint.

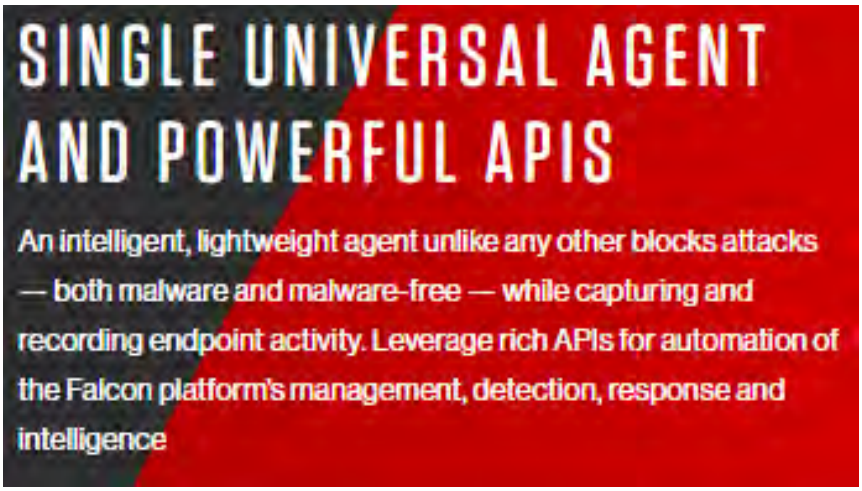
299. Defendants have infringed and continue to infringe one or more claims of the ’591 Patent in violation of 35 U.S.C. § 271 in this District and elsewhere in the United States and will continue to do so unless enjoined by this Court. The Accused Products, including features including features of the Falcon Platform such as Falcon Prevent, at least when used for their ordinary and customary purposes, practice each element of at least claim 1 of the ’591 Patent as described below.

300. For example, claim 1 of the ’591 Patent recites:

> 1. A computer-implemented method comprising:
>
> monitoring a memory space of a process for execution of at least one monitored function of a plurality of functions, wherein monitoring the memory space comprises loading a component for evaluating the at least one monitored function in the memory space;

> invoking one of the plurality of functions as a result of receiving a call from an application programming instance;
>
> executing stack walk processing upon the invocation of one of the plurality of functions in the monitored memory space; and
>
> performing, during the executing of the stack walk processing before an address of an originating caller function is reached, a memory check for a plurality of stack entries identified during the stack walk processing to detect suspicious behavior, wherein an alert of suspicious behavior is triggered when the performing of the memory check detects at least one of the following:
>
> code execution is attempted from non-executable memory,
> a base pointer is identified as being invalid,
> an invalid stack return address is identified,
> attempted execution of a return-oriented programming technique is detected,
> the base pointer is detected as being outside a current thread stack, and
> a return address is detected as being inside a virtual memory area,
>
> wherein when an alert of suspicious behavior is triggered, preventing execution of a payload for the invoked function from operating.

301. The Accused Products perform the method of claim 1 of the ’591 Patent. To the extent the preamble is construed to be limiting, the Accused Products perform *a computer-implemented method*, as further explained below. For example, the Falcon Platform includes “[a]n intelligent, lightweight agent unlike any other [that] blocks attacks — both malware and malware-free — while capturing and recording endpoint activity. Leverage rich APIs for automation of the Falcon platform’s management, detection, response and intelligence.”



(*See* WBR_CSK000455 (https://www.crowdstrike.com/endpoint-security-products/falcon-platform/) at 459.)

302. On information and belief, the Accused Products perform a method that includes *monitoring a memory space of a process for execution of at least one monitored function of a plurality of functions, wherein monitoring the memory space comprises loading a component for evaluating the at least one monitored function in the memory space*. For example, the "Falcon Platform can detect a fileless attack using web shell" because the "Falcon Sensor sits in the kernel and CrowdStrike focuses on malicious patterns or indicators of attack" to detect hacking tools in which "no file is written to a disk." In another example, as shown below, the Accused Products display information for an event related to "HOST CS-WEBSERV1-TMM" and "USER NAME CS-WEBSERV1-TMM" and connected a series of events including "[root]," "smss.exe," another "smss.exe," "wininit.exe," "services.exe," "svchost.exe," "w3wp.exe," "cmd.exe," "ipconfig.exe," another "cmd.exe," "whoami.exe," another "cmd.exe," and "NETSTAT.EXE," including "w3wp.exe" using the command prompt "cmd.exe" to perform malicious actions. The Accused Products and their "indicators of attack…recognize that this series of events corresponds to a webshell exploit" and "see the commands entered in the command prompt—whoami, ipconfig,

acting on their behalf;

(ii) destroy or deliver all such infringing products to Plaintiffs;

(iii) revoke all licenses to all such infringing products;

(iv) disable all web pages offering or advertising all such infringing products;

(v) destroy all other marketing materials relating to all such infringing products;

(vi) disable all applications providing access to all such infringing software; and

(vii) destroy all infringing software that exists on hosted systems,

i) That this Court, if it declines to enjoin Defendants from infringing any of the Asserted Patents, award damages for future infringement in lieu of an injunction; and

j) That this Court award such other relief as the Court deems just and proper.

DEMAND FOR JURY TRIAL

Plaintiffs respectfully request a trial by jury on all issues triable thereby.

DATED: August 31, 2022

By:*/s/ Jeffrey D. Mills*

Jeffrey D. Mills
Texas Bar No. 24034203
KING & SPALDING LLP
500 West Second St., Suite 1800
Austin, Texas 78701
Telephone: (512) 457-2027
Facsimile: (512) 457-2100
jmills@kslaw.com

Mark D. Siegmund
STECKLER WAYNE CHERRY & LOVE, PLLC
8416 Old McGregor Rd.
Waco, Texas 76712
Telephone: (254) 651-3690
Facsimile: (254) 651-3689
mark@swclaw.com

Steve Sprinkle
Texas Bar No. 00794962
SPRINKLE IP LAW GROUP, P.C.
1301 W. 25th Street, Suite 408
Austin, Texas 78705
Telephone: 512-637-9220
ssprinkle@sprinklelaw.com

Christopher C. Campbell (DC Bar No. 444262)
Patrick M. Lafferty *(pro hac vice)*
KING & SPALDING LLP
1700 Pennsylvania Avenue, NW
Suite 200
Washington, DC 20006
Telephone: (202) 626-5578
Facsimile: (202) 626-3737
ccampbell@kslaw.com
plafferty@kslaw.com

Britton F. Davis *(pro hac vice)*
Brian Eutermoser (*pro hac vice)*
KING & SPALDING LLP
1401 Lawrence Street
Suite 1900
Denver, CO 80202
Telephone: (720) 535-2300
Facsimile: (720) 535-2400
bfdavis@kslaw.com
beutermoser@kslaw.com

Attorneys for Plaintiffs Open Text, Inc. and Webroot, Inc.

CERTIFICATE OF SERVICE

A true and correct copy of the foregoing instrument was served or delivered electronically via the U.S. District Court ECF filing system to all counsel of record on this 31st day of August, 2022.

/s/ Jeffrey D. Mills
Jeffrey D. Mills