

Exhibit A-14

Invalidity of U.S. Patent No. 10,601,780

by Jose Nazario, DEFENSE AND DETECTION STRATEGIES AGAINST INTERNET WORMS (Artech House, 2004) (“Nazario”)

As shown in the claim chart below, the asserted claims of the '780 patent are invalid under 35 U.S.C. § 102(a) and/or (b)¹ as anticipated by Jose Nazario, DEFENSE AND DETECTION STRATEGIES AGAINST INTERNET WORMS (Artech House, 2004) (“Nazario”), and/or is invalid under 35 U.S.C. § 103 as obvious in view of Nazario alone or in combination with the reference(s) specifically identified in the following claim chart or in one or more other reference identified in Defendant’s Preliminary Invalidity Contentions (“Defendant’s Invalidity Contentions”).

To the extent a finder of fact determines that the references cited herein do not teach certain limitations in the asserted claims, such limitations would have been inherent and/or obvious. These claims are also invalid as obvious in view of Nazario alone or in combination with other prior art references, including, but not limited to, the prior art identified in the Cover Pleading of Defendants’ Invalidity Contentions and the prior art described in the claim charts attached in Appendix A. Amazon reserves the right to supplement its preliminary invalidity contentions by identifying additional prior art references that render these claims invalid in combination with Nazario, including, for example, the prior art identified in Appendix B.

Defendant’s Invalidity Contentions are based, in part, upon Defendant’s present understanding of the asserted claims and Croga’s apparent interpretations of the asserted claims in its September 20, 2024 Preliminary Infringement Contentions (Croga’s “Infringement Contentions”) and Defendant’s investigation to date. Defendant is not adopting Croga’s constructions or apparent constructions, nor is Defendant admitting to the accuracy of any particular contention or construction. The citations provided in the charts below are exemplary rather than exhaustive and Defendant reserves the right to rely upon additional references uncovered through further searching, other portions of the cited references, and/or other portions of references cited within Defendant’s Invalidity Contentions. Defendant further incorporates by reference the reservation of rights identified in the cover pleading to Defendants’ Invalidity Contentions as though fully set forth herein.

¹ Nazario was publicly available at least as early as 2004, which is more than one year before January 27, 2011, the earliest priority date of the '780 patent asserted by Croga. See Croga’s Infringement Contentions at 7.

Claim	'780 Patent Claim Language	Disclosure
1.pre	A networked computer system comprising:	To the extent that the preamble is held to be limiting, Nazario discloses a networked computer system. <i>See, e.g.,</i> Nazario at p. 162. 10.1 Honeypots A network honeypot is simply a system you expect to get probed or attacked so that you can analyze these data later. As defined by Spitzner, a honeynet differs from a honeypot in that it is a network of honeypots made of full production systems. This network can be logically and geographically dispersed. Because of their nature, worms will indiscriminately attack any available host on the network, including honeypots. Nazario discloses this preamble as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
1.a	a network;	Nazario discloses a network. <i>See, e.g.,</i> Nazario at p. 162. 10.1 Honeypots A network honeypot is simply a system you expect to get probed or attacked so that you can analyze these data later. As defined by

Claim	'780 Patent Claim Language	Disclosure
		Spitzner, a honeynet differs from a honeypot in that it is a network of honeypots made of full production systems. This network can be logically and geographically dispersed. Because of their nature, worms will indiscriminately attack any available host on the network, including honeypots. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
1.b	at least one computer system configured to connect to the network, the computer system comprising a host system and a virtual system,	Nazario discloses at least one computer system configured to connect to the network, the computer system comprising a host system and a virtual system. <i>See, e.g.,</i> Nazario at p. 162. 10.1 Honeypots A network honeypot is simply a system you expect to get probed or attacked so that you can analyze these data later. As defined by Spitzner, a honeynet differs from a honeypot in that it is a network of honeypots made of full production systems. This network can be logically and geographically dispersed. Because of their nature, worms will indiscriminately attack any available host on the network, including honeypots. The value of this approach is that you can analyze the attack after it has happened and

Claim	'780 Patent Claim Language	Disclosure
		learn about the methods used by the attacking agent. Honeypots come in three basic varieties: • Full dedicated systems, which are typically nonhardened installations of an operating system. These are installed with a minimum amount of setup in an attempt to mirror a default installation and then placed on the network. External monitors are typically used to capture the network traffic to and from the host. • Service-level honeypots are hosts that have one or more services installed in logical "jails," areas of protected process and memory space. An attacker can probe and attack the service, but any compromise is contained to the virtual machine running on the host. Commercial as well as open-source versions of these tools are available. • Virtual hosts and networks, which provide the illusion of a host and its associated services to an attacker. This is typically housed in a single host on the network, spoofing other hosts. Each of these approaches offers varying degrees of accessibility and value, along with associated risk. <i>See, also Nazario at p. 220.</i> A related concept, and one that is common for both practical uses as well as security, is the use of virtual hosts for exposed services. Popular in the hosting provider service market, virtual hosts are full system images that reside in memory partitioned from the host operating system. In this way, a single large server can act as the host operating system to several guest installations of the same or alternate operating systems. Fast becoming popular in the security

Claim	'780 Patent Claim Language	Disclosure
		marketplace, this provides a similar advantage to the use of the chroot() system call. The advantage, however, is that the exposed operating system that is visible to the service appears to be a full operating system. It is significantly harder to determine that the underlying system is only a guest system on a larger host, and no attacks at this time have been successful in breaking out of this system. Several commercial and free implementations of this system exist. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
1.c	wherein the virtual system is a separate operating system or a software module operating on the computer system,	Nazario discloses wherein the virtual system is a separate operating system or a software module operating on the computer system. <i>See, e.g.,</i> Nazario at p. 220. A related concept, and one that is common for both practical uses as well as security, is the use of virtual hosts for exposed services. Popular in the hosting provider service market, virtual hosts are full system images that reside in memory partitioned from the host operating system. In this way, a single large server can act as the host operating system to several guest installations of the same or alternate operating systems. Fast becoming popular in the security marketplace, this provides a similar advantage to the use of the chroot() system call. The advantage, however, is that the exposed operating system that is visible to the service appears to be a full operating system. It is significantly harder to determine that the underlying system is only a guest system

Claim	'780 Patent Claim Language	Disclosure
		on a larger host, and no attacks at this time have been successful in breaking out of this system. Several commercial and free implementations of this system exist. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
1.d	wherein an internal firewall is configured to separate the host system from the virtual system in the computer system, and	Nazario discloses wherein an internal firewall is configured to separate the host system from the virtual system in the computer system. <i>See, e.g.,</i> Nazario at p. 212. This chapter focuses on host-based defense measures. These include host-based firewalls and antivirus software used to detect the installation of worms on desktops. Additional strategies discussed will include application sandboxing, the practice of confining an application to a small area of a host with minimal privileges, and some information on the protection of systems via patching and service minimization. <i>See also</i> Nazario at p. 216. 12. 5 Partitioned privileges Multiuser systems, typically found on server systems, usually have the rights and authorized actions for users partitioned into groups. One group may be able to read most of the system., for example, but not modify system files. Another may be able to read sensitive files but not modify them.

Claim	'780 Patent Claim Language	Disclosure
		<i>See also Nazario at p. 217.</i> However, these access rights do not need to be maintained over the life-time of a program, such as with a Web server. Any such system that does not need to repeatedly access sensitive files can discard the elevated privileges it began with once restricted operations are performed. This can be achieved in several ways. The first is through access controls that allow for access to what would normally be restricted operations to certain processes or users. These can include the binding of a reserved listening socket to accept inbound connections [2]. This would allow a network server program to be run in a limited privilege space, using only what would be needed to begin launch and handling of inbound requests. Any compromise of the server process would be limited in the additional actions it can take on the basis of the process's capabilities. Such capability systems are increasingly found in commercial software, including Windows NT and 2000 systems and many popular forms of UNIX. <i>See also Nazario at pp. 219-220.</i> 12.6 Sandboxing of applications Similar to the concept of partitioning privileges for processes is the principle of partitioned system access for applications. In this approach, the root file system is restricted to a subset of the real file system. Processes are then executed in this confined space with minimal setup. This typically makes use of the chroot() system call in UNIX systems. In doing so, the hope is to minimize any damage an attacker will make to a small subset of the file system.

Claim	'780 Patent Claim Language	Disclosure
		Furthermore, access to the tools and libraries typically needed to be leveraged to elevate privileges on a compromised system are missing, making this task difficult, if not impossible. While originally designed to study attackers, it works equally as well for worms. As the worm compromises the system, it cannot take full control of the host and therefore cannot cause further damage. In some circumstances it cannot even begin to operate, as the required pieces (such as a compiler or script interpreter) are missing. The correct method of calling the chroot() function in the C language is to first change to the new directory root and then set it as the root directory... This prevents attacks that are able to break out of the new root directory structure and gain full access to the system. Last, the user ID of the process being executed is changed to a nonprivileged user ID. This sequence of system calls helps to stop many attacks on the system from succeeding. By changing to an unprivileged user, the full view on the file system by the process is limited, helping to reduce any effort to break out of the process's file system area. <i>See also Nazario at p. 220.</i> A related concept, and one that is common for both practical uses as well as security, is the use of virtual hosts for exposed services. Popular in the hosting provider service market, virtual hosts are full system images that reside in memory partitioned from the host operating system. In this way, a single large server can act as the host operating system to several guest installations of the same or alternate operating systems. Fast becoming popular in the security marketplace, this provides a similar advantage to the use of the chroot() system call. The advantage, however, is that the exposed

Claim	'780 Patent Claim Language	Disclosure
		operating system that is visible to the service appears to be a full operating system. It is significantly harder to determine that the underlying system is only a guest system on a larger host, and no attacks at this time have been successful in breaking out of this system. Several commercial and free implementations of this system exist. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
1.e	a host-based firewall executed on the computer system is configured to implement network isolation between the computer system and the network; and	Nazario discloses a host-based firewall executed on the computer system is configured to implement network isolation between the computer system and the network. <i>See, e.g.,</i> Nazario at p. 212. This chapter focuses on host-based defense measures. These include host-based firewalls and antivirus software used to detect the installation of worms on desktops. Additional strategies discussed will include application sandboxing, the practice of confining an application to a small area of a host with minimal privileges, and some information on the protection of systems via patching and service minimization. <i>See also</i> Nazario at p. 213-214. 12.3 Host firewalls

Claim	'780 Patent Claim Language	Disclosure
		Host-based firewalls are a complement to a network-based firewall. While most systems do not run host-based firewalls, instead relying on the network's firewall to provide them with protection, at the host level more fine-grained control can be applied. This method also acts as a failover protection for the network-based firewall should any attack bypass that mechanism. These situations can include the penetration of a worm behind the perimeter firewall or a difference between the policy enforced by the perimeter defenses and those required by the host. There are several example situations where host-based firewalls may be an appropriate solution to defending a set of hosts. These include situations where the default network security policy is absent but the security requirements for the host are more demanding. Alternatively, a system may wish to dynamically add addresses to its list of blocked hosts that would have otherwise been permitted into the network. The fact that any host-based firewall cannot, without some convolutions, be more liberal than the perimeter firewall between it and the Internet at large is a design issue. Host-level firewalls are available in two major types. The first is a traditional firewall with statically configured rules. In this type of firewall a set of rules is established that enforces a policy. This can include coarse-grained rules such as the network ports and their associated services that are allowed to be accessed. Finer grained rules would enforce rules about which hosts are allowed to connect to these services. This type of firewall would also work well for a system with a well-defined and narrow network role, such as a network server. The second type of popular host-based firewall is one that dynamically adapts to the user's network use. Often called the

Claim	'780 Patent Claim Language	Disclosure
		personal firewall, these systems query the user to determine what applications are in use on the system. Sources are associated with applications, giving the user an easy-to-use secure Internet workstation. Combined with a default deny policy, a personal firewall on a workstation can help prevent a network worm from entering a system via a previously unauthorized network path. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
1.g	wherein the device comprises a processor and a memory configured to implement network isolation between one or more untrusted network destinations and the networked computer system.	Nazario discloses wherein the device comprises a processor and a memory configured to implement network isolation between one or more untrusted network destinations and the networked computer system. <i>See, e.g.</i>, Nazario at p. 235. Cisco IOS-based routers have had filtering capabilities for several of their versions as of this writing. IOS uses access-list (ACL) statements, access-group statements and rules to manage traffic decisions. An example collection of several IOS access-list statements in a configuration would appear as follows: <pre>access-list 100 deny icmp any any fragments access-list 100 permit icmp any any echo access-list 100 permit tcp 192.168.1.0 0.0.0.255 any eq 22</pre> <i>See also</i> Nazario at p. 235.

Claim	'780 Patent Claim Language	Disclosure
		The Cisco PIX product, a dedicated firewall device, features a filtering statement in addition to the access-list and access-group statements found in IOS. The shun statement provides a coarse-grained filtering capability for filtering networks, as shown below: <pre>shun 10.1.1.27 10.2.2.89 555 666 tcp</pre> This statement would block any TCP traffic from 10.1.1.27 with a source port of 555 to the host 10.2.2.89 with a destination port of 666. <i>See also Nazario at p. 236.</i> Lastly, the popular and freely available IP Filter (IPF) tool from Darren Reed can also be used to build a filtering host (http://coombs.anu.edu/~avalon). IPF is available as a module for many popular operating systems, both freely available and commercially supported. <i>See also Nazario at p. 251.</i> 14.4 Web-based proxies The next type of common application gateway is the proxy for Web services. Again, the application gateway device serves as an intermediary for Web clients as they make requests to external servers. Just as is done for incoming electronic mail, the payload in a server's reply to a request can be screened for content with the offending bits removed. Web proxies are popular not only for content screening but also for performance reasons. By storing a local cache of the content and then recycling it for other clients, the load on the external link can be reduced.

Claim	'780 Patent Claim Language	Disclosure
		See also Nazario at p. 253. 14.5.I Strengths of proxy-based defenses ... Unlike a packet filter, which can only understand the contents of a packet, a proxy device offers true application-layer filtering. This can give the advantage of content specific filtering. As described above, this also gives the advantage of normalizing the communications stream, removing the ambiguity for signature matching and content-inspection or application handling. This gives the network administrators full control over the content of the communications. Application gateways default to a block policy during failure. Because no other gateway for communications is enabled, if the application gateway fails due to attack or an error, all internetwork communications will cease. No open window for an attack is created by the failure of the network filtering device. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
3.pre	The networked computer system of claim 1,	To the extent that the preamble is held to be limiting Nazario discloses the networked computer system of claim 1. See <i>supra</i> claim 1.

Claim	'780 Patent Claim Language	Disclosure
		Nazario discloses this preamble as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
3.a	wherein the host system is configured to store data in a host memory space and	Nazario discloses wherein the host system is configured to store data in a host memory space. <i>See, e.g.,</i> Nazario at p. 220. A related concept, and one that is common for both practical uses as well as security, is the use of virtual hosts for exposed services. Popular in the hosting provider service market, virtual hosts are full system images that reside in memory partitioned from the host operating system. In this way, a single large server can act as the host operating system to several guest installations of the same or alternate operating systems. Fast becoming popular in the security marketplace, this provides a similar advantage to the use of the chroot() system call. The advantage, however, is that the exposed operating system that is visible to the service appears to be a full operating system. It is significantly harder to determine that the underlying system is only a guest system on a larger host, and no attacks at this time have been successful in breaking out of this system. Several commercial and free implementations of this system exist. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or

Claim	'780 Patent Claim Language	Disclosure
		more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
3.b	the virtual system is configured to store data in a virtual memory space that is segregated from the host memory space.	Nazario discloses the virtual system is configured to store data in a virtual memory space that is segregated from the host memory space. <i>See supra</i> limitation 3.a. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
7.	The networked computer system of claim 1, wherein the device is configured to prevent unauthorized communication between the computer system and the one or more untrusted network destinations.	Nazario discloses the networked computer system of claim 1, wherein the device is configured to prevent unauthorized communication between the computer system and the one or more untrusted network destinations. <i>See supra</i> claim 1. <i>See, e.g.</i>, Nazario at p. 235. Cisco IOS-based routers have had filtering capabilities for several of their versions as of this writing. IOS uses access-list (ACL) statements, access-group statements and rules to manage traffic decisions. An example collection of several IOS access-list statements in a configuration would appear as follows: access-list 100 deny icmp any any fragments

Claim	'780 Patent Claim Language	Disclosure
		access-list 100 permit icmp any any echo access-list 100 permit tcp 192.168.1.0 0.0.0.255 any eq 22 <i>See also</i> Nazario at p. 235. The Cisco PIX product, a dedicated firewall device, features a filtering statement in addition to the access-list and access-group statements found in IOS. The shun statement provides a coarse-grained filtering capability for filtering networks, as shown below: shun 10.1.1.27 10.2.2.89 555 666 tcp This statement would block any TCP traffic from 10.1.1.27 with a source port of 555 to the host 10.2.2.89 with a destination port of 666. <i>See also</i> Nazario at p. 236. Lastly, the popular and freely available IP Filter (IPF) tool from Darren Reed can also be used to build a filtering host (http://coombs.anu.edu/~avalon). IPF is available as a module for many popular operating systems, both freely available and commercially supported. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.

Claim	'780 Patent Claim Language	Disclosure
10.pre	The networked computer system of claim 1,	To the extent that the preamble is held to be limiting Nazario discloses the networked computer system of claim 1. <i>See supra</i> claim 1. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
10.a	wherein one or more applications or processes are configured to run in the host system, and	Nazario discloses wherein one or more applications or processes are configured to run in the host system. <i>See, e.g.</i>, Nazario at p. 211. Host-Based Defenses 12.1 Part overview ... The easiest way to defend against network-based worms coming from the Internet is to remove any links to the outside world. This would leave only the internal network vulnerable to attacks that originated inside. Obviously, this is not a viable solution for many, because the Internet's communications links are important for business, research, and even our personal lives. This means that this avenue cannot be explored, though it has been used as a temporary measure by many network administrators during especially heavy onslaughts of worm attacks.

Claim	'780 Patent Claim Language	Disclosure
		The second major line of defense is to move all exposed services from well-known ports to uncommonly used ports. This would mean, for example, running a Web server on a port that is different than the normal port 80/TCP port used. The major drawback to this approach is that the outside world, which needs to communicate with your site, will be unable to do so without assistance on your part. With that assistance, it is possible that worms could similarly use that information to exploit the vulnerabilities that still may reside on your servers but on different ports. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
10.b	wherein the one or more applications or processes running in the host system are configured to communicate with one or more devices on the network.	Nazario discloses wherein the one or more applications or processes running in the host system are configured to communicate with one or more devices on the network. <i>See, e.g.</i>, Nazario at p. 211. Host-Based Defenses 12.1 Part overview ... The easiest way to defend against network-based worms coming from the Internet is to remove any links to the outside world. This would leave only the internal network vulnerable to attacks that originated inside. Obviously, this is not a viable solution for many, because the Internet's communications links are important for business, research, and even our personal lives. This means that this avenue cannot be

Claim	'780 Patent Claim Language	Disclosure
		explored, though it has been used as a temporary measure by many network administrators during especially heavy onslaughts of worm attacks. The second major line of defense is to move all exposed services from well-known ports to uncommonly used ports. This would mean, for example, running a Web server on a port that is different than the normal port 80/TCP port used. The major drawback to this approach is that the outside world, which needs to communicate with your site, will be unable to do so without assistance on your part. With that assistance, it is possible that worms could similarly use that information to exploit the vulnerabilities that still may reside on your servers but on different ports. <i>See also Nazario at p. 223.</i> 12.7.2 Features within a service The second step in the protection of a network at the host and service level is to ensure that the services are properly configured. Misconfiguration of services can expose the host to new vulnerabilities that would otherwise be absent. If the software itself is secure, this effort may be in vain. Many of the Web servers affected by the Code Red worm were not known to be vulnerable to the worm due to a poor understanding of the features in the software. This is based on the demographics of many of the Code Red sources [5, 6]. The vulnerable component of the server software, an indexing utility enabled by default, can be shut off by reconfiguring the server [7, 8). This effectively removes the exposed risk of the Web server without requiring an upgrade or reinstallation, which may cause downtime. By using such a strategy,

Claim	'780 Patent Claim Language	Disclosure
		a more comprehensive solution can be developed and tested and implemented at a more convenient time, such as the weekend. Nazario discloses this limitation as identified above. This limitation is also obvious in view of Nazario combined with the knowledge of a person having ordinary skill in the art alone, or with any of the references identified as a combination in Section II.A. of the cover pleading to Defendant's Invalidity Contentions or with any one or more of the references as exemplified in the corresponding limitation of Exs. A-1 to A-20. A person having ordinary skill in the art would have been motivated to make these combinations based on one or more of the motivations to combine identified in Section II.A.d. of the cover pleading to Defendant's Invalidity Contentions.
11.pre	A method of network isolation in a networked computer system, the method comprising:	To the extent that the preamble is held to be limiting, Nazario discloses a method of network isolation in a networked computer system. <i>See supra</i> limitation 1.pre.
11.a	providing a network and	Nazario discloses providing a network. <i>See supra</i> limitation 1.a.
11.b	at least one computer system that is configured to connect to the network,	Nazario discloses at least one computer system that is configured to connect to the network. <i>See supra</i> limitation 1.b.
11.c	the computer system comprising a host system and a virtual system, wherein the virtual system is a separate operating system or a software module operating on the computer system;	Nazario discloses the computer system comprising a host system and a virtual system, wherein the virtual system is a separate operating system or a software module operating on the computer system. <i>See supra</i> limitation 1.c.
11.d	separating the host system from the virtual system using an internal	Nazario discloses separating the host system from the virtual system using an internal firewall executed on the computer system.

Claim	'780 Patent Claim Language	Disclosure
	firewall executed on the computer system;	<i>See supra</i> limitation 1.d.
11.e	implementing network isolation between the computer system and the network using a host-based firewall executed on the computer system;	Nazario discloses implementing network isolation between the computer system and the network using a host-based firewall executed on the computer system. <i>See supra</i> limitation 1.e.
11.f	providing at least one device configured to implement a network firewall or a web proxy; and	Nazario discloses providing at least one device configured to implement a network firewall or a web proxy. <i>See supra</i> limitation 1.f.
11.g	implementing network isolation, between one or more untrusted network destinations and the networked computer system, via the at least one device.	Nazario discloses implementing network isolation, between one or more untrusted network destinations and the networked computer system, via the at least one device. <i>See supra</i> limitation 1.g.
13.pre	The method of claim 11, further comprising:	To the extent that the preamble is held to be limiting, Nazario discloses the method of claim 11. <i>See supra</i> claim 11. <i>See also supra</i> limitation 3.pre.
13.a	the host system storing data in a host data storage; and	Nazario discloses the host system storing data in a host data storage. <i>See supra</i> limitation 3.a.
13.b	the virtual system storing data in a virtual data storage.	Nazario discloses the virtual system storing data in a virtual data storage. <i>See supra</i> limitation 3.b.

Claim	'780 Patent Claim Language	Disclosure
17.	The method of claim 11, further comprising the at least one device preventing unauthorized communication between the computer system and the one or more untrusted network destinations.	Nazario discloses the method of claim 11, further comprising the at least one device preventing unauthorized communication between the computer system and the one or more untrusted network destinations. <i>See supra</i> claim 11. <i>See also supra</i> claim 7.
20.pre	The method of claim 11,	To the extent that the preamble is held to be limiting, Nazario discloses the method of claim 11. <i>See supra</i> claim 11. <i>See also supra</i> limitation 10.pre.
20.a	further comprising running one or more applications or processes in the host system	Nazario further comprising running one or more applications or processes in the host system. <i>See supra</i> limitation 10.a.
20.b	that are configured to communicate with one or more devices on the network.	Nazario discloses that are configured to communicate with one or more devices on the network. <i>See supra</i> limitation 10.b.