EXHIBIT A-8

scale to tens of millions of checks per day." Talmon Marco, in his declaration and recent deposition testimony, discusses the use of Audible Magic's technology in the iMesh P2P software. Marco says, "Deployment of these filtering tools has been extremely successful in blocking files unauthorized for distribution without impairing noninfringing use of the iMesh service." Acoustic fingerprinting can also be used to filter video files, such as music videos and television shows, by comparing the fingerprint of the audio content of video files with a database of such fingerprints.

ii. Possible Filtering Systems

109. Lime Wire LLC could adopt acoustic filtering by applying it to files a user is attempting to download. In one scenario, a user enters a query and receives search results; the user then clicks on one of the search results, initiating a download. If available, LimeWire would send the file's hash to Audible Magic for checking against their database of hashes. If the hash is not in their database, the file is downloaded by LimeWire. LimeWire would then compute the file's fingerprint using Audible Magic's software. This fingerprint is then transmitted to Audible Magic's server, which tries to match it, and depending upon the result the downloaded file is blocked or not. If the file is blocked, its hash can be added to Audible Magic's database for future blocking. Over time, Lime Wire LLC could build its own database of hashes, possibly making the check for unauthorized content even faster. This system could also be supplemented with a keyword filter that would be populated with song title and artist terms provided by a third party.

110. Lime Wire LLC could adopt acoustic filtering by applying it to files a user is attempting to upload. In one scenario, when LimeWire receives a request to upload a file, it would first compute the file's fingerprints (it may have already done so) and send

them to the Audible Magic server for checking. If Audible Magic does not find the file in its database, it can signal the LimeWire client to initiate the upload.

111. Another additional variation to the above strategy would be to apply filtering when a user *joins* the network rather than on a download. Lime Wire LLC could send the fingerprints of all files in the shared folder to an Audible Magic server. The server would check the file information against a database of unauthorized content. Any unauthorized files would be signaled back to the LimeWire client, which could mark the file in such a way that it cannot be uploaded.[44]

112. I understand that Lime Wire LLC has considered implementing a filter based on acoustic fingerprinting and has done some development work in that regard.[45]

F. *Lime Wire LLC can communicate with installed clients and can control certain settings*

i. SIMPP

113. Lime Wire LLC is able to communicate with its running LimeWire file sharing applications by using the so-called Signed Message Parameter Passing (SIMPP) mechanism. Using this method, Lime Wire LLC is able to control remotely various settings of the LimeWire file sharing application. It accomplishes this by using the following mechanism: when a LimeWire client starts, it reads a set of properties from the file limewire.props. It then connects to one or more ultrapeers to receive any updates; each simpp.xml file includes a string by which LimeWire can verify that the file comes from Lime

[44] LimeWire already runs servers to check for unauthorized content, e.g. fserv1.limewire.com.

[45] Sam Berlin Deposition Transcript 225:1-230:25.

Wire LLC. This guarantees that the messages were actually created by Lime Wire LLC and prevents entities other than Lime Wire LLC from successfully distributing simpp.xml files. After confirming that a message is legitimate, the LimeWire client copies the update values into the file simpp.xml and uses those values while it is running. Eventually, when the program exits, the corresponding properties in simpp.xml are copied into limewire.props, and hence they will be used the next time LimeWire is started. The entire process is started when Lime Wire LLC creates a simpp.xml file with a new version number and seeds it to one or more ultrapeers.

114. Two sample simpp.xml files are contained in Exhibit D. The first line contains a base32-encoded signature followed by a version number. Following that, there is a set of lines of the form "name=value," each containing a setting for some LimeWire property. A number of these settings are of special interest. For example, "evil_hosts=Bearshare 5.2" and "FilterSettings.hostileIps=128.108.*.*; 208.109.*.*; ...." (a long list of IP addresses). The evil_hosts property defines a (list of) names and version numbers of file sharing application(s) that LimeWire ultrapeers will de-preference in their determination of whether to make a connection to another ultrapeer.[46] The FilterSettings.hostileIps property defines a list of IP addresses of users with which the LimeWire application will not communicate.[47] The LimeWire application is, however,

[46] See https://www.limewire.org/fisheye/browse/limecvs/core/com/limegroup/gnutella/ConnectionManagerImpl.java?r=1.10.

[47] A discussion of hostileIPs can be found at https://www.limewire.org/jira/browse/LWC-676.

programmed specifically to reject any list of hostile IPs that blocks more than 2.5% of all possible IPs.[48]

115. LimeWire LLC is able to control the *turning off* of its content filtering mechanism using the content.managementActive directive in the simpp.xml file. According to the LimeWire source code,[49] the content.managementActive directive must be set to "true" for content filtering to be enabled. The effect of setting the directive to "false" would be to remotely turn the content filter off for all users who had received that simpp.xml file, overriding a user's local setting of "on." The converse, however, is not the case. Lime Wire LLC's setting the directive to "true" merely permits a user to determine whether to enable or disable the content filter.[50]

116. Recently, LimeWire LLC has added a new directive to its simpp.xml file called thirdpartysearchresultssettings.searchdatabase. The value of the directive is a set of phrases, some of which appear to be names of musical groups that are offered for sale at LimeSpot.com, e.g. http://thelowlows.limespot.com/. A search for "low lows" brings up a search result that points to the LimeSpot website.

---

[48] See https://www.limewire.org/fisheye/browse/limecvs/core/com/limegroup/gnutella/filters/IPList.java?r=1.18; also see Balevsky revision 1.18 check-in comment at https://www.limewire.org/fisheye/browse/limecvs/core/com/limegroup/gnutella/filters/IPList.java.

[49] See code at https://www.limewire.org/fisheye/browse/limecvs/core/com/limegroup/gnutella/settings/ContentSettings.java.

[50] The file simpp.xml contains dozens of settings for various parameters. By setting these parameters to extreme values, one could affect search and download performance.

117. There is no technical reason why Lime Wire LLC could not remotely control additional functions of installed LimeWire clients by using the SIMPP mechanism.

ii. Update Notification

118. Lime Wire LLC is able to notify its running LimeWire file sharing applications that a new version is available. First, Lime Wire LLC can cause installed clients to display an announcement regarding the new version in a line near the bottom of the client's window. Second, as shown in Figure 22 below, LimeWire can be directed to display a popup window that appears when a user runs a particular search. The window alerts the user to the existence of a new version and provides a link to the limewire.com website so the new version can be downloaded. Lime Wire LLC is able to communicate the fact that a new version is available by placing this information within the new version and directing the new version to append the information when it communicates with other Gnutella clients. Each copy of LimeWire is programmed to look for special, so-called Vendor Messages within Gnutella communication with other LimeWire clients.[51] Upon receipt of such a message, and verifying that it is from a LimeWire file sharing application, the message containing information about the new version is extracted, and the window in Figure 22 below is produced.

[51] There is a standard mechanism in Gnutella for adding additional information in Gnutella messages. This is done using the Gnutella Generic Extensions Protocol (GGEP), specifically the provision for vendor-specific messages. The GGEP specification can be found at http://209.85.173.104/search?q=cache:JuHtbiA06_AJ:gnutella-specs.rakjar.de/index.php/Gnutella_Generic_Extension_Protocol+Gnutella+Generic+Extensions+Protocol+GGEP+0.5&hl=en&ct=clnk&cd=1&gl=us&client=safari.

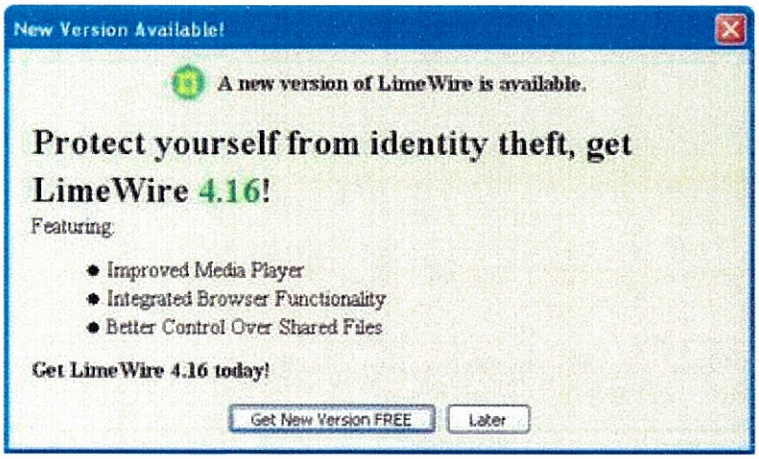

Figure 22: Popup Window Declaring a New Version of LimeWire Exists

(The button links to **www.limewire.com**)

April 18, 2008

Ellis Horowitz