Cuadernos de Ingeniería. Nueva Serie.
Revista de la Facultad de Ingeniería de la
Universidad Católica de Salta, núm. 11, 2019
ISSN 2545-692X (papel) / 2542-7012 (on line): 7-20

Ares Galaxy: análisis del comportamiento y del código fuente

Ares Galaxy: analysis of the behavior and the source code

Matías Amor[1], Iván Maccio[1], Ignacio Occhipinti[1], María Lorena Talamé[1] y Alejandra Cardoso[1]

Resumen

Ares Galaxy es uno de los programas más conocidos utilizados para la descarga de archivos de música y videos, entre otros. Estos tipos de programas se basan en la arquitectura de comunicación Peer to Peer que permite el intercambio de información entre las computadoras de la red. Los programas de descarga de archivos son muy populares y fáciles de usar. Sin embargo, muchas veces los usuarios desconocen que este tipo de intercambio tiene riesgos, como, por ejemplo, favorecer el tráfico de pornografía infantil. Por ello, resulta importante investigar si Ares Galaxy realiza modificaciones en el sistema operativo sin el consentimiento del usuario.

Para detectar posibles modificaciones en el registro del sistema operativo y su configuración, se siguieron tres caminos. Se observaron los archivos generados y modificados durante la instalación y uso de Ares Galaxy, y luego, se analizaron con software de forensia informática para comprender el contenido de archivos encriptados. También se examinó el comportamiento de Ares Galaxy desde su código fuente.

En este proyecto de la cátedra Compiladores, se utilizaron herramientas como la gramática de Delphi, expresiones regulares para la detección de elementos de interés dentro del programa y otras herramientas relacionadas con compiladores y traductores.

Palabras clave: Ares Galaxy, código fuente, informática forense, pornografía infantil, P2P.

Abstract

Ares Galaxy is one of the most popular programs used to download music files and videos, among others. These types of programs are based on the architecture of communication Peer to Peer that allows the exchange of information among the computers of the network. File download programs are very popular and easy to use. However, many times users are unaware that this type of information exchange has risks, such as favoring the trafficking of child pornography. Therefore, it is important to investigate whether Ares Galaxy makes modifications to the operating system without the user's consent.

To detect possible modifications to the operating system registry and its configuration, three paths were followed. The files generated and modified during the installation and use of Ares Galaxy were observed, and then, analyzed with forensics software to understand the content of encrypted files. The behavior of Ares Galaxy was also examined from its source code.

Citar: Matías Amor et al. (2019). Ares Galaxy: análisis del comportamiento y del código fuente. *Cuadernos de Ingeniería. Nueva Serie*. [Salta - Argentina], núm. 11: 7-20

[1] Facultad de Ingeniería – Universidad Católica de Salta (UCASAL)

Some of the tools used in this project of the Compiladores subject were: Delphi grammar, regular expressions for the detection of elements of interest within the program and other tools related to compilers and translators.

Key words: Ares Galaxy, source code, computer forensic, child pornography, P2P.

1. Introducción

Ares Galaxy es uno de los programas más conocidos para la descarga de archivos de música y videos, entre otros. Estos tipos de programas se basan en la arquitectura de comunicación Peer to Peer (P2P), que permite el intercambio de información entre las computadoras de la red.

Los programas de descarga de archivos son muy populares y fáciles de usar. Sin embargo, muchas veces los usuarios desconocen que este tipo de intercambio de información tiene riesgos, como, por ejemplo, favorecer el tráfico de pornografía infantil.

El proyecto tiene como objetivo comprobar si Ares Galaxy realiza modificaciones en el sistema operativo sin el consentimiento del usuario, especialmente en puertos y en carpetas compartidas. El análisis se realiza examinando el código fuente del programa escrito en el lenguaje de programación Delphi, y analizando las modificaciones en el registro de Windows producidas durante la instalación y uso del software.

Este proyecto surge en la cátedra Compiladores, de la carrera Ingeniería en Informática de la Universidad Católica de Salta. Se utiliza la gramática de Delphi, expresiones regulares para la detección de elementos de interés dentro del programa y otras herramientas relacionadas con compiladores y traductores.

En la sección 2 de este trabajo se presenta el marco legislativo argentino sobre pornografía infantil. En la sección 3 se define la red de intercambio de archivos P2P, y luego, en la sección 4, se describen las principales características de Ares Galaxy. La sección 5 presenta los pasos desarrollados en esta investigación, detallándolos en las secciones siguientes. En la última parte de este trabajo se enuncian las conclusiones arribadas.

2. Legislación argentina sobre pornografía infantil

Antes de abordar la descripción técnica de Ares Galaxy, se enuncia el contexto legal que se tiene en cuenta para el análisis de esta herramienta. Ares Galaxy permite el intercambio de material digital, eventualmente material pornográfico, y ante la posibilidad de que realice acciones de intromisión indebida en el equipo del usuario, conviene describir la normativa argentina respecto a delitos informáticos, en particular sobre pornografía infantil.

El Convenio de Ciberdelincuencia del Consejo de Europa define a los delitos informáticos como “los actos dirigidos contra la confidencialidad, la integridad y la disponibilidad de los sistemas informáticos, redes y datos informáticos, así como el abuso de dichos sistemas, redes y datos” (Consejo de Europa, 2001).

El Convenio de Budapest entró en vigor en noviembre de 2001. A través de la estandarización de conceptos y formas de actuar, tiene como fin la cooperación internacional para combatir los delitos informáticos como estafas, pornografía infantil, y todo tipo de acción vinculada a la

propiedad intelectual. Años más tarde, en 2017, Argentina se adhiere oficialmente al Convenio de Budapest, con la sanción de la Ley 27.411.

En la República Argentina, en el año 2008 fue sancionada y promulgada la Ley 26.388 que modifica el Código Penal incorporando algunas figuras de delitos informáticos. En el artículo 2, se establecen las penas de prisión en lo referido a la publicación y distribución de material de contenido sexual con participación de menores de edad. Las penas de prisión van desde seis meses a cuatro años a quien produjere, publicare o divulgare por cualquier medio, cualquier representación de menores de dieciocho años dedicado a actividades sexuales explícitas o representación de sus partes sexuales, y hasta dos años a quien tuviere en su poder esas representaciones para fines de comercialización o distribución (Honorable Congreso de la Nación Argentina, 2008).

En el mes de marzo del año 2018, a partir de la incorporación de Argentina al Convenio de Budapest, el Senado de la Nación promulgó la Ley 27.436 que incorpora una mejora de la Ley 26.388, que también castiga a quien tuviere en su poder material sexual infantil (Honorable Congreso de la Nación Argentina, 2018).

Temperini (2013) realiza un estudio comparativo de las leyes que regulan delitos informáticos en América Latina y en el cual se observa que la República Argentina es uno de los países que en los últimos años adhirió a las leyes internacionales y además, legisló recientemente sobre pornografía infantil.

3. Intercambio de archivos

El intercambio de todo tipo de material en Internet es cada vez más frecuente. Se puede intercambiar información almacenada en audios, videos, libros electrónicos y otros tipos de archivos. Existen diversas formas de compartir información. Una de las más utilizadas es mediante el uso de redes P2P.

En una red P2P todas las computadoras conectadas funcionan como clientes y como servidores (Figura 1). Este tipo de redes hace un buen uso del ancho de banda disponible entre los usuarios para el intercambio de archivos, lo que implica una mejor velocidad de transferencias. Sin embargo, el mal uso de este tipo de redes, contribuye y facilita actividades ilegales online (Wilson y Bazli, 2016), por ejemplo la distribución de material de contenido sexual infantil.

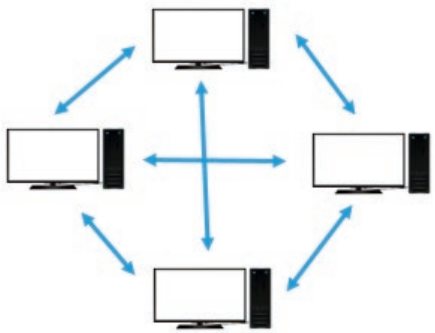

Figura 1. *Red Peer to Peer*

Típicamente, los programas de descarga P2P funcionan de la siguiente manera: cuando un usuario descarga un archivo, generalmente lo descarga en varias pequeñas partes distribuidas en los nodos de la red. Esto permite que la descarga se realice rápidamente. A su vez, estas partes descargadas pueden ser compartidas con otros usuarios de la red. Entre los programas P2P más populares se encuentran: Ares Galaxy, Emule, uTorrent o Vuze.

4. Ares Galaxy

Ares Galaxy[2], conocido popularmente como Ares[3], es uno de los programas P2P más utilizados. Su última versión es la 2.4.8, publicada en mayo de 2018. Es un software de código abierto, desarrollado en el lenguaje de programación Delphi para Windows.

Sus características principales son:

- Previsualización de archivos multimedia como audio o video
- Breves colas de espera
- Biblioteca de gestión de archivos compartidos
- Múltiples ventanas de búsqueda

Al iniciar el proceso de instalación de Ares se deben aceptar las condiciones de uso[4]. En ese documento se advierte que el uso del software para actividades ilegales está prohibido, incluso la infracción de las leyes de propiedad intelectual y que el usuario puede ser objeto de sanciones civiles y/o penales.

También señala que algunos archivos descargados podrían contener virus o spyware, y que las "carpetas compartidas" permiten que otros usuarios accedan a la información contenida en ellas, por lo cual sugiere no tener información personal. Respecto a la pornografía, se menciona que algunos archivos de contenido pornográfico podrían ser deliberadamente mal etiquetados y atraer a personas jóvenes o desprevenidas; y que los usuarios cuyas carpetas compartidas tuviesen material pornográfico ilegal, en particular pornografía infantil, podrían sufrir enjuiciamiento penal.

5. Desarrollo

La investigación sobre el estudio del comportamiento de Ares se dividió en tres partes principales:

1. Análisis de la instalación de Ares
   En esta fase se analizaron cambios en el registro del sistema operativo haciendo uso de RegShot, una aplicación destinada a tal fin.
2. Análisis de archivos .dat generados por el software
   En este punto, el objetivo fue examinar el contenido de estos archivos que pudieran tener información sobre carpetas no compartidas o modificaciones de puertos no autorizados. Se utilizaron dos herramientas de forensia digital: Magnet AXIOM y Ares Decrypter.
3. Análisis del código fuente
   A fin de encontrar sentencias y variables que pudieran realizar modificaciones no establecidas por el usuario, se analizó el código del programa. En primer lugar, se exami-

2 https://www.ares.com.es/

3 De aquí en adelante se mencionará Ares para referenciar a Ares Galaxy.

4 https://www.ares.com.es/eula.html

naron las instrucciones que se disparan con las acciones del usuario desde la interfaz de Ares, y luego se extendió el estudio al resto del programa.

Todas las pruebas se realizaron sobre el sistema operativo Windows 7 Pro de 32 bits, instalado en una máquina virtual VMWare. La versión de Ares utilizada en este proyecto fue la 2.1.8.

5.1 Análisis de la instalación de Ares

Toda instalación de software genera ciertos cambios en los sistemas operativos que garantizan su correcto funcionamiento. Si bien durante el proceso de instalación, el setup solicita que el usuario se involucre en la configuración, no todos los parámetros son modificables, ni tampoco visibles al usuario durante el proceso.

RegShot es una utilidad de comparación de registros de código abierto (LGPL) que permite tomar rápidamente una instantánea o foto del registro de Windows y luego compararla con una segunda, hecha después de modificar el sistema o instalar un nuevo producto de software. En este trabajo se utilizó Regshot 1.9.0[5].

Previo a la instalación de Ares, se ejecutó Regshot y se capturó la primera instantánea (Figura 2). Luego se instaló Ares en la PC, con los parámetros por defecto. Se ejecutó Ares y se realizaron búsqueda de archivos, descargas, carga de archivos para compartir, lo que simularía una ejecución típica de cualquier usuario, con el fin de observar luego si estas acciones impactaban en los archivos .dat que se iban a analizar. Se observó una variedad de archivos con nombres de películas de Disney (por ejemplo, "El rey león") por lo cual se optó por descargar estos archivos.

Figura 2: Primera captura de Regshot.

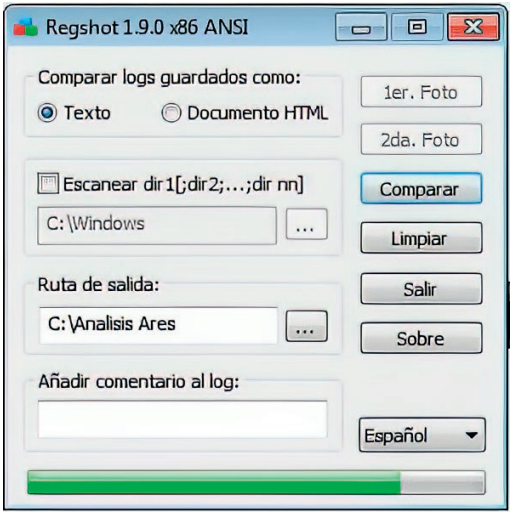

Figura 3: Comparación de 1ra y 2da imagen del registro

5 https://regshot.uptodown.com/windows

Se ejecutó nuevamente *Regshot*, capturando la segunda imagen del registro y se realizó la comparación de ambas fotos (Figura 3), generando un archivo de texto como resultado. En la Figura 4 se observa un fragmento del archivo obtenido.

```
Regshot 1.9.0 x86 ANSI
Comentarios:
Computador:7PRO32 , 7PRO32
Usuario:Administrador , Administrador

----------------------------------
Claves añadidas:93
----------------------------------
HKLM\SOFTWARE\Classes\CLSID\{083863F1-70DE-11D0-BD40-00A0C911CE86}\Instance\{3E0FA044-
926C-42D9-BA12-EF16E980913B}
HKLM\SOFTWARE\Classes\CLSID\{083863F1-70DE-11D0-BD40-00A0C911CE86}\Instance\{422A3AF6-
0B1D-42CB-AAF9-7DFD8EB2FCEF}
HKLM\SOFTWARE\Classes\CLSID\{3E0FA044-926C-42D9-BA12-EF16E980913B}
HKLM\SOFTWARE\Classes\CLSID\{3E0FA044-926C-42D9-BA12-EF16E980913B}\InprocServer32
HKLM\SOFTWARE\Classes\CLSID\{422A3AF6-0B1D-42CB-AAF9-7DFD8EB2FCEF}
HKLM\SOFTWARE\Classes\CLSID\{422A3AF6-0B1D-42CB-AAF9-7DFD8EB2FCEF}\InprocServer32
HKLM\SOFTWARE\Classes\.arescol
HKLM\SOFTWARE\Classes\.arlnk
HKLM\SOFTWARE\Classes\.pls
HKLM\SOFTWARE\Classes\.torrent
HKLM\SOFTWARE\Classes\Ares.Arlnk
HKLM\SOFTWARE\Classes\Ares.Arlnk\shell
```

Figura 4: Fragmento de los resultados de la comparación de Regshot.

5.2 Análisis de los archivos .dat

Durante el proceso de instalación y ejecución, Ares genera archivos que almacenan información importante para su uso. Esta información es guardada en archivos de extensión .dat ubicados en la ruta por defecto: AppData\Local\Ares\Data.

Los archivos .dat se utilizan en muchas aplicaciones y contienen datos genéricos que sirven como parámetros de configuración o información para aplicativos y suelen estar encriptados. No existe un programa específico que permita abrir estos archivos, pero se encuentran disponibles una variedad de aplicaciones que permiten inspeccionarlos.

En el caso de Ares, los archivos generados no se pueden abrir con cualquier programa que los desencripte, debido a que Ares los cifra con un algoritmo único. Para poder leer estos archivos y comprender cuál es su función, se utilizaron dos programas de carácter forense: *Magnet AXIOM* y *AresDecrypter*.

Estos programas pueden entenderse como "traductores". Un traductor es un programa que toma como entrada un programa o texto escrito en un lenguaje (lenguaje fuente) y produce como salida un texto equivalente en otro lenguaje (lenguaje objeto). En el caso de los programas forenses usados, la entrada es el contenido de cada archivo .dat y el lenguaje objeto o salida es texto entendible por cualquier persona.

Magnet AXIOM

Magnet AXIOM es un software forense que permite el análisis de "artefactos"[6]. Con esta herramienta se pudo comprobar la información relacionada con los archivos compartidos desde

[6] Los SO cuentan con procesos o mecanismos que dejan algún rastro del uso de aplicaciones, de la conexión, de los accesos por parte de los usuarios, descargas, etc., comúnmente llamados artefactos

Ares. En la Figura 5 se observan los tres archivos que se estaban compartiendo durante el análisis. En el cuadro inferior de la derecha se detallan los datos del archivo seleccionado: el valor de hash, el peso y tamaño, y el origen sharel.dat. Los archivos *shareh.data* y *sharel.dat* almacenan información relacionada a los archivos que se comparten o descargan.

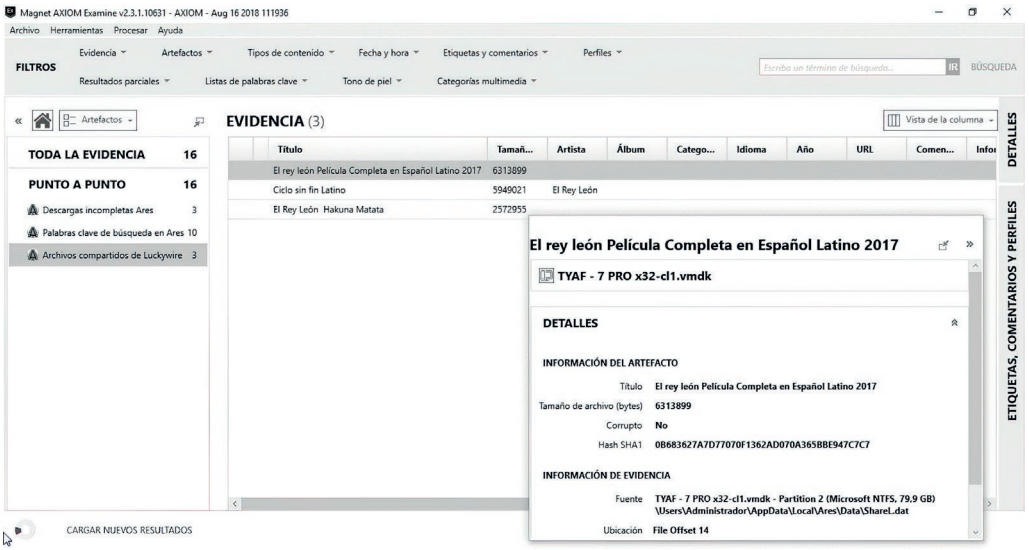

Figura 5: Archivos compartidos – Magnet AXIOM

AresDecrypter

AresDecrypter es un programa desarrollado por Fekruna Forensic[7], con el cual se pudo explorar más a fondo el contenido de los archivos *sharel.dat*, *shareh.dat*, *ntuser.dat*, y los registros propios de Ares que se crean o modifican en el registro de Windows.

El archivo *shareh.dat* guarda un registro por cada archivo descargado de Ares que a su vez es compartido. Este archivo mantiene el historial de archivos compartidos. A diferencia del archivo *sharel.dat* (también guarda un registro por cada archivo descargado y compartido), los registros del primero no son eliminados cuando el archivo es eliminado de la carpeta de descarga (por default MySharedFolder).

Las pruebas que se realizaron se hicieron pensando en las posibles acciones de un usuario de programas P2P: buscar un archivo; comenzar la descarga; descargar archivos en su totalidad; interrumpir la descarga en algún momento; compartir archivos; a los archivos descargados cambiarlos de carpeta y verificar si Ares lo sigue usando de semilla. En cada alternativa, se analizaron los archivos *.dat* antes y después de cada acción.

7 http://www.fekruna.com/

Durante las pruebas, se observó que si bien todos los archivos descargados o agregados a la biblioteca generaban registros en el *sharel.dat*, no siempre lo hacían en el *shareh*. Además, para que el archivo genere un registro en el *shareh*, debía contener algún metadata de Ares.

Aquellos archivos descargados y pertenecientes a Ares generan registros. Sin embargo, a aquellos archivos agregados manualmente había que modificarles la metadata desde la biblioteca para que se graben en el *shareh.dat*. Por otro lado, se observó que *shareh.dat* guarda un registro único por cada tipo archivo, y se estima que está basado en el valor del hash sha1[8] (Figura 6).

De esta forma, se observa que Ares sólo agrega las entradas necesarias para su funcionamiento sin modificar otras preexistentes. A su vez, los puertos que habilita son los informados en el archivo de configuración.

El mecanismo que utiliza para compartir contenido es a través del uso de los archivos *shareh.dat* y *sharel.dat*. Como ambos ficheros se encuentran encriptados, es necesario contar con programas de forensia para su alteración.

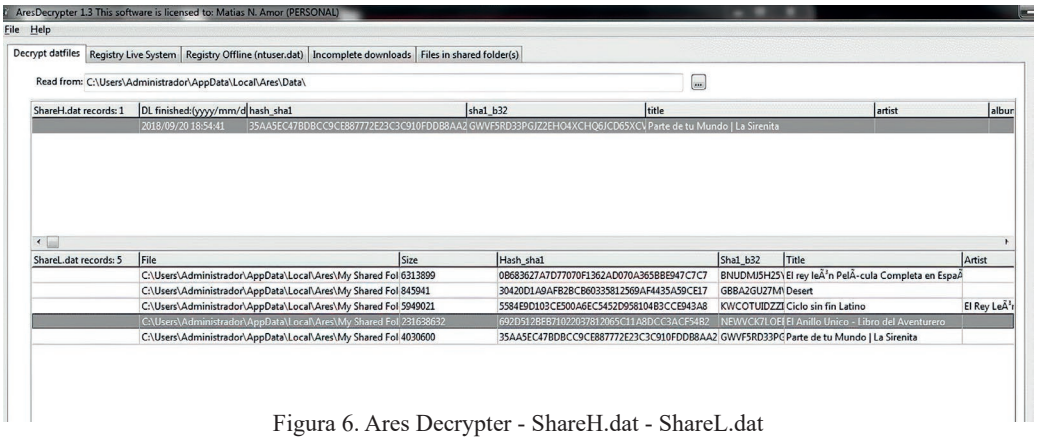

Figura 6. Ares Decrypter - ShareH.dat - ShareL.dat

5.3 Análisis del código fuente

La versión Ares 2.1.8 está compuesta por más de 100 módulos (archivos con extensión *pas*[9]), 5MB aproximadamente y unas 124.500 líneas de código. El código fuente se obtuvo de *SourceForge*[10], un sitio para la colaboración de proyectos de software de código libre. Para el análisis, se descartaron los módulos que contienen funciones de ayuda, reproductor de música, etc., centrándolo en aquellos relacionados a tareas de interconexión.

El objetivo en esta etapa fue identificar instrucciones que pudieran modificar puertos o carpetas compartidas desde módulos que no fuesen los de configuración, o a partir de las acciones realizadas durante la búsqueda y/o descarga de archivos.

Para el estudio del código fuente se dividieron las tareas en dos partes:

[8] Algoritmo matemático que, a partir de una entrada, genera una salida alfanumérica de longitud fija que representa toda la información de entrada.

[9] pas es la extensión típica de un archivo de programas que contienen código fuente escrito en Pascal o Delphi.

[10] https://sourceforge.net

1. Se exploraron las acciones que realiza Ares cuando, desde la interfaz, el usuario presiona alguno de los botones disponibles para buscar o descargar archivos.
2. Se examinaron las sentencias de asignación de variables relacionadas a puertos y carpetas compartidas de los módulos de interés.

Se comenzó explorando las acciones que realiza Ares cuando el usuario presiona alguno de los botones disponibles para búsqueda o descarga de archivos. Luego, se revisó el código fuente, capturando las sentencias de asignación de variables relacionadas a puertos y carpetas compartidas. Para este trabajo, se utilizó la gramática de Delphi y se armaron expresiones regulares para seleccionar las sentencias buscadas.

Análisis desde la interfaz

Con el objetivo de constatar si Ares pone a disposición de los usuarios archivos que no fueron intencionalmente compartidos, se analizó el código de los procedimientos que se ejecutan al realizar una búsqueda.

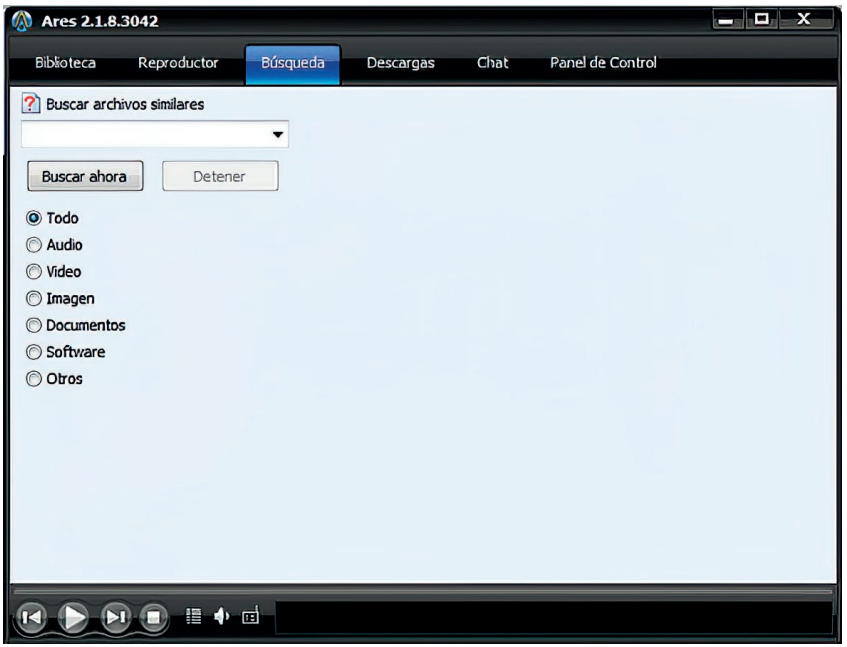

Figura 7: Búsqueda de archivos

En la ventana de búsqueda de archivos existe un cuadro para ingresar las palabras a buscar. Al hacer click en el botón "Buscar ahora" (Figura 7), se ejecuta el procedimiento *Tares_frmmain.Btn_start_searchClick* (Figura 8) que se encuentra en la unidad ufrmmain.pas y llama al procedimiento *gui_start_search*. Dicho procedimiento (que se encuentra en *helper_search_gui.pas*), únicamente se encarga de acciones relacionadas con la interfaz. Entre ellas, se puede mencionar la desactivación del botón "Buscar ahora", la habilitación del botón "Detener" o el inicio de un temporizador.

```
procedure Tares_frmmain.Btn_start_searchClick(Sender: TObject);
begin
gui_start_search;
end;
```

Figura 8: Procedimiento que se ejecuta al comenzar la búsqueda

Cuando se detiene la búsqueda (botón Detener) se ejecuta el procedimiento *Tares_frmmain.btn_stop_searchClick* (Figura 9) de la unidad *ufrmmain.pas*. El procedimiento *gui_stop_search* habilita el botón "Buscar ahora" y deshabilita el botón "Detener".

```
procedure Tares_frmmain.btn_stop_searchClick(Sender: TObject);
begin
gui_stop_search;
end;
```

Figura 9: Procedimiento que se ejecuta al detener la búsqueda

En la interfaz, una vez recuperados aquellos archivos que coinciden con los criterios de búsqueda, se pueden seleccionar los que se van a descargar. Estos archivos se descargan con el botón "Descarga" (Figura 10). Con esta acción, se ejecuta el procedimiento *Tares_frmmain.Download1Click* (Figura 11) que se encuentra en la unidad *ufrmmain.pas*. En el código de este procedimiento no se observan instrucciones que impacten en algún puerto o carpeta del equipo.

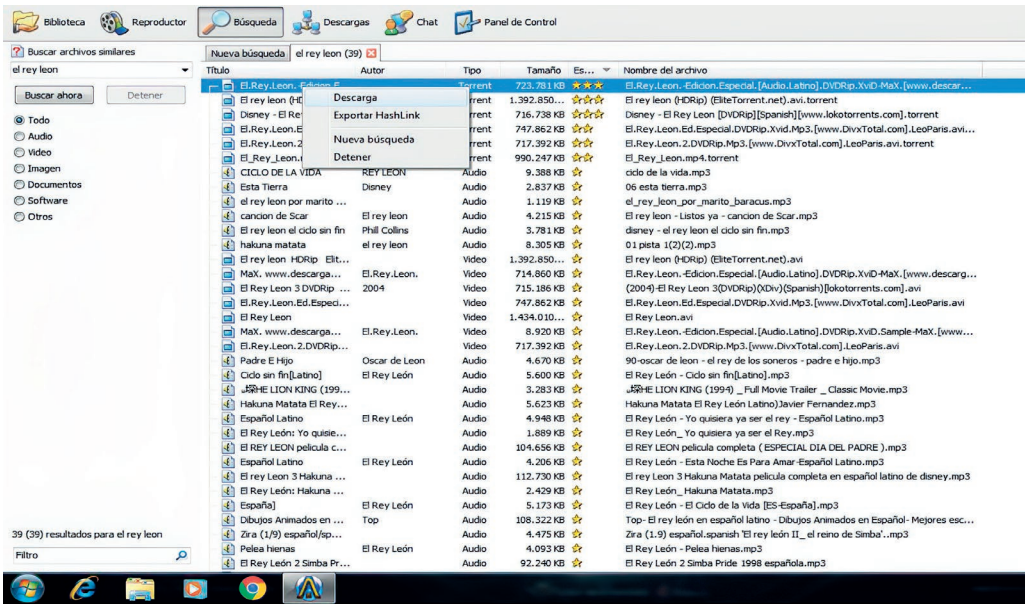

Figura 10: Descarga de archivos

```
procedure Tares_frmmain.Download1Click(Sender: TObject);
var
node,node_child,selected_node:PCmtVNode;
datao,data_child:precord_search_result;
down:tdownload;
hi:integer;
src:precord_panel_search;
begin

try

for hi:=0 to src_panel_list.count-1 do begin
 src:=src_panel_list[hi];
 if src^.containerPanel<>pagesrc.activepanel then continue;

with src^.listview do begin
node:=GetFirstSelected;
while (node<>nil) do begin
 if getnodelevel(node)>0 then selected_node:=node.parent
  else selected_node:=node;
 datao:=getdata(selected_node);

 if datao^.downloaded then begin
  node:=getnextselected(node);
  continue;
 end;

 if is_in_progress_sha1(datao^.hash_sha1) then begin
messageboxW(self.handle,pwidechar(GetLangStringW(STR_TRANSFER_ALREADY_IN_PROGRESS)+CRLF+CRL
F+'('+extract_fnameW(utf8strtowidestr(datao^.filenameS))+'  )'+CRLF+CRLF+GetLangStringW(STR
_TAKE_A_LOOK_TO_TRANSFER_TAB)),pwidechar(appname+':
'+GetLangStringW(STR_DUPLICATE_REQUEST)),mb_ok+MB_ICONEXCLAMATION);
  exit;
 end;

 if is_in_lib_sha1(datao^.hash_sha1) then begin

messageboxW(self.handle,pwidechar(GetLangStringW(STR_FILE_ALREADY_IN_LIBRARY)+CRLF+CRLF+Get
LangStringW(STR_FILE)+':
'+extract_fnameW(utf8strtowidestr(datao^.filenameS))+CRLF+GetLangStringW(STR_SIZE)+':
'+format_currency(datao^.fsize)+chr(32)+STR_BYTES+CRLF+CRLF+GetLangStringW(STR_TAKE_A_LOOK_
TO_YOUR_LIBRARY)),pwidechar(appname+chr(58)+chr(32){':
'}+GetLangStringW(STR_DUPLICATE_FILE)),mb_ok+MB_ICONEXCLAMATION);
  exit;
 end;

down:=start_download(datao);
lista_down_temp.add(down);
 GUI_add_sources_ares(src^.listview,down,selected_node,datao);

 datao^.downloaded:=true;
  if node.childcount>0 then begin
     node_child:=getfirstchild(selected_node);
    while (node_child<>nil) do begin
      data_child:=getdata(node_child);
      data_child^.downloaded:=true;
       invalidatenode(node_child);
      node_child:=getnextsibling(node_child);
    end;
  end;
 invalidatenode(selected_node);
 put_backup_results_inprogress(src,datao);
```

Continúa en el siguiente bloque

```
 node:=getnextselected(node);
end;
end;
break;
end;
except
end;
end;
```

Figura 11. Procedimiento que se ejecuta en la descarga

Análisis del código fuente

En esta etapa se continuó con el análisis de código fuente. Se extrajeron las sentencias que pudieran modificar las variables de configuración como puertos y carpetas compartidas de todas las unidades del programa, con la suposición que se pudieran realizar modificaciones sin el consentimiento ni conocimiento del usuario.

Un lenguaje de programación se puede definir describiendo la estructura de sus programas (la *sintaxis* del lenguaje) y el significado de sus programas (la *semántica* del lenguaje) (Aho et al., 2008). La sintaxis de un lenguaje de programación describe la forma correcta en la cual las sentencias, expresiones y unidades de programa se deben escribir. Para especificar la sintaxis de un lenguaje se utiliza una notación llamada gramática, en particular *gramática independiente del contexto*. Una gramática está compuesta por símbolos y reglas de producción que describen la sintaxis.

Partiendo de la gramática del lenguaje de programación Delphi (Charte Ojeda, 2003), se buscaron aquellas sentencias que pudieran resultar interesantes para este estudio; principalmente, sentencias de asignación y definición de variables. Para esto se utilizó un programa escrito en el lenguaje de programación *Python*.

Para extraer las sentencias de asignación en todos los módulos seleccionados, se definió una expresión regular. Una expresión regular es una secuencia de caracteres que describe un patrón de texto, generalmente se la utiliza para ubicar, dentro de un archivo, cadenas que se equiparen con el patrón (Hopcroft et al., 2007). La *expresión regular* básica para representar el lado izquierdo de una sentencia de asignación se observa en la Figura 12.

```
([a-z][a-z0-9]+(\.([a-z][a-z0-9]+)?(:=)
```

Figura 12: Expresión regular

En la Figura 13 se puede observar el procedimiento que define la expresión regular y obtiene las sentencias de asignaciones que se equiparan con la misma. Obsérvese que en el procedimiento la variable "pal" contiene la palabra buscada. Por lo tanto, en la expresión regular básica se reemplazó la subexpresión [a-z][a-z0-9]+. Se obtuvieron alrededor de 19000 sentencias de asignación, de las cuales se seleccionaron aquellas que tuvieran, en el lado izquierdo de la asignación, variables con nombres que hicieran alusión a carpetas o términos relacionados con las redes P2P.

Estos nombres surgieron sabiendo que Ares Galaxy utiliza el protocolo torrent[11], con lo cual en las líneas de código podrían aparecer nombres de variables derivados de términos particularmente utilizados en este protocolo, tales como peers, leechers, seeders, trackers, node, supernode, hash, etc. También se escogieron nombres de variables que hicieran referencias a los puertos y a los protocolos de red TCP y UDP, por ejemplo, port, socket, etc. Por otro lado, se descartaron las sentencias que asignaban valores simples. Para esta tarea se utilizó otro programa en Python que, a partir de una lista con estos nombres, seleccionó las sentencias de asignación que las contuviera. Se obtuvieron 756 sentencias. Con estas pruebas y analizando cada una de estas líneas, no se observó que Ares habilite puertos, protocolos de red ni que modifique las carpetas compartidas puesto que en el código siempre se hace referencia a variables globales. Estas variables globales toman su valor inicial de los parámetros de configuración definidos por el usuario.

```
def procesar(archivo):

   with open(archivo,'r',encoding='cp850') as f:

      for k in range(len(vpal)):

          pal=vpal[k].lower().replace('\n','').strip()
          if len(pal)>0:
             regex = re.compile(r'(^[a-z]|\s*)'+pal+'(\s*(:=)|\.[a-z]+\s*(:=))')
             nrolin=1
             for linea in f.readlines():
                 lin=linea.lower().strip()
                 if (regex.search(lin)):
                     w=[archivo,str(nrolin),linea.lstrip()]
                     if (w not in vsale):
                         vsale.append(w)
                 nrolin+=1
             f.seek(0)
    f.close()
```

Figura 13: Procedimiento que recupera sentencias de asignación

6. Conclusiones

Este trabajo presentó varios caminos para intentar llegar al mismo fin: determinar si Ares era capaz de modificar variables de configuración sin intervención del usuario.

Se analizaron los archivos .dat generados y modificados durante la instalación y/o durante la ejecución del programa. Se utilizaron tres programas frecuentemente usados en forensia digital. Fue posible examinar el historial de descarga de archivos en Ares, los archivos compartidos y los valores de los campos que se pueden configurar.

De los módulos que componen el programa Ares, se analizaron las sentencias de asignación que contuvieran nombres de puertos y carpetas compartidas, de las cuales se observó que ninguna hacía modificaciones en puertos o carpetas compartidas.

[11] El protocolo torrent o BitTorrent consiste en disponer de varios servidores desde donde el usuario descarga archivos. Cuando el usuario sube un archivo, hace que esté disponible en la red a través de un nodo BitTorrent que actúa como semilla. Si otros usuarios quieren descargar el archivo, obtienen el archivo torrent y crean otro nodo que actúa como cliente, intercambiando partes del archivo con la semilla y con otros clientes.

Se puede concluir que Ares, en la versión 2.1.8, no realiza modificación alguna de las variables de configuración en lo que respecta a puertos, protocolos y carpetas compartidas sin intervención del usuario. Para modificar la carpeta de archivos compartidos o el puerto de comunicación, el usuario tiene que modificarlo en la sección establecida a tal fin.

Ares es una interfaz que permite al usuario abstraerse del funcionamiento del protocolo torrent. Los detalles del funcionamiento de la red P2P y del protocolo torrent no fueron modificados por los autores de Ares. Sin embargo, al ser Ares un programa de código abierto, no se puede descartar que cualquier persona con conocimientos en programación pueda modificarlo y, eventualmente, realizar acciones delictivas.

Referencias

Aho, A. V., Ravi Sethi, M. L., y Ullman, J. (2008). *Compiladores: Principios, técnicas y herramientas.* Mexico: Pearson Educación.

Charte Ojeda, F. (2003). *Delphi 7* (1 ed.). Madrid: Anaya Multimedia.

Consejo de Europa. (2001). *Informática Legal.* Recuperado el Septiembre de 2018, de http://www.informaticalegal.com.ar/2001/11/23/convencion-de-budapest-sobre-ciberdelincuencia/

Honorable Congreso de la Nación Argentina (2008). *Argentina.gob.ar.* Recuperado el Septiembre de 2018, de https://www.argentina.gob.ar/normativa/nacional/ley-26388-141790/texto

Honorable Congreso de la Nación Argentina (2018). *Argentina.gob.ar.* Recuperado el Septiembre de 2018, de https://www.argentina.gob.ar/normativa/nacional/ley-27436-309201/texto

Hopcroft, J., Motwani, R. & Ullman, J. (2007). *Teoría de autómatas, lenguajes y computación.* Madrid: Pearson Education.

Temperini, M. G. (2013). Delitos Informáticos en Latinoamérica: Un estudio de derecho comparado. 1ra. Parte. *1er. Congreso Nacional de Ingeniería Informática / Sistemas de Información.* Córdoba: CONAIISI.

Wilson, M. & Bazli, B. (2016). Forensic analysis of i2p activities. *Automation and Computing (ICAC) 22nd International Conference* (págs. 529-534). IEEE

Recibido: febrero de 2019
Aceptado: agosto de 2019