3. Diseño y desarrollo

3.1 Introducción

En el proceso de desarrollo de este proyecto surgieron una variedad de problemas y una cantidad de decisiones por tomar. Resolver dichas dificultades resultó en la mejora de la herramienta utilizada originalmente. Así también, fue necesario elegir los lenguajes de programación a utilizar y las herramientas de análisis más adecuadas para el cumplimiento del objetivo. Al final la herramienta mantiene un funcionamiento estable, al mismo tiempo entrega información contundente y condensada.

Sin embargo, las capacidades de esta herramienta aún se pueden extender. Y llegar a un mejoramiento sustancial y más completo de la herramienta, de tal forma que permita su libre utilización en el trabajo de análisis de códigos maliciosos. Es importante y ampliamente recomendable que la herramienta esté en constante actualización y mantenimiento. En unos cuantos meses más podría llegar a ser necesaria la migración de Windows XP a Windows Seven.

En esta parte del informe se presenta una breve descripción de las diferentes herramientas mejoradas y algunas desarrolladas que componen al proyecto TRUMAN ahora ampliado.

3.2 Herramientas para la ejecución de binarios en Windows

La herramienta para la ejecución de los binarios en Windows tiene la intención de obtener el binario del código malicioso desde una ubicación de red, a través de un servicio HTTP ejecutándose en el servidor de TRUMAN, para que posteriormente el agente malicioso directamente se ejecute en el Microsoft Windows cliente. Todo lo anterior lo puede llevar a cabo a través de la funcionalidad del lenguaje de programación batch de Mircrosoft Windows.

El programa en batch utiliza la herramienta wget.exe únicamente para descargar el código malicioso. Lo realiza mediante la siguiente línea de código; donde la opción “-q” sirve para apagar su salida, la opción “-O” funciona enviando la salida a un archivo. %SERVER_IP% y %REPORT_CGI% son constantes definidas al principio del script.

```
wget -q -O C:\ok.txt http://%SERVER_IP%/cgi-bin/%REPORT_CGI%?res=booted >
nul
```

Posteriormente un simple comando de ejecución hace que podamos ver su comportamiento. La misma herramienta detecta el momento en que es ejecutado el código malicioso (escribe el archivo C:\flag.txt con un número "1") y se pone en un estado de cuenta regresiva, en el cual espera cinco minutos para reiniciar el equipo automáticamente (lo anterior lo lleva a cabo el script C:\restart.bat al leer el contenido del archivo C:\flag.txt, el cual si es 0 volverá a leerse en un segundo y si es 1 se ejecutará el comando para reiniciar el equipo: `wmic os where primary=true Call Reboot`). (Fig. 3.1)

Para consultar el código fuente completo del archivo C:\get.bat revisar el anexo 1. Y el anexo 2 para el caso de C:\restart.bat.

El script C:\get.bat fue reditado y ajustado a la necesitades actuales del proyecto respecto a los sistemas operativos contemporáneos (2009-2010), este archivo ya existía en el proyecto original, base de este informe. El script C:\restart.bat, fue creado para resolver los problemas de reinicio automático.

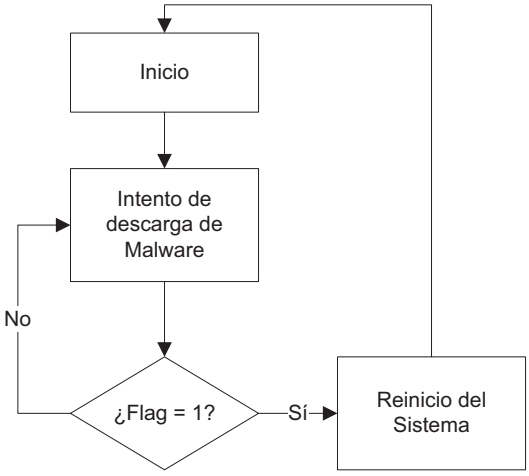

Fig. 3.1. Diagrama de flujo de la tarea desempeñada por ambos scripts.

3.3 Herramienta de análisis del sistema de archivos

La herramienta para el análisis del sistema de archivos es la encargada de obtener las bases de datos de los sistemas de archivos, primero la del Microsoft Windows XP no contaminado y segundo la del Microsoft Windows XP infectado. De cada uno de los archivos obtiene su ruta

absoluta y su firma MD5, dichos datos le sirven posteriormente para realizar las validaciones de creación, modificación y eliminación de archivos respecto de una imagen con la otra.

Se ejecuta desde el sistema operativo GNU/Linux y se trata de un programa en perl, el cual está facultado para la construcción de la base de datos de archivos y la comparación de éstos mismos por medio del uso de las poderosas cualidades del lenguaje perl para el manejo de cadenas. La herramienta para el análisis del sistema de archivos fue desarrollada en su totalidad, ya que el método que utiliza TRUMAN original era muy simple y precario.

Este script está estructurado mediante el uso de funciones, definición de variables, utilización de sentencias de control como “if” y “foreach”, manejo de archivos, entre otros elementos. Recibe como argumentos el archivo de base de datos original “/forensics/orig/orig.md5” (sin infección) y el archivo de base de datos nuevo (con infección). Trabaja en dos modos; el modo “INICIAL”, el cual sólo generá la base de datos del sistema de archivos original o sin infección; y el modo “COMPARACIÓN”, en el cual compara el archivo original con el archivo nuevo o con infección. (Fig. 3.2)

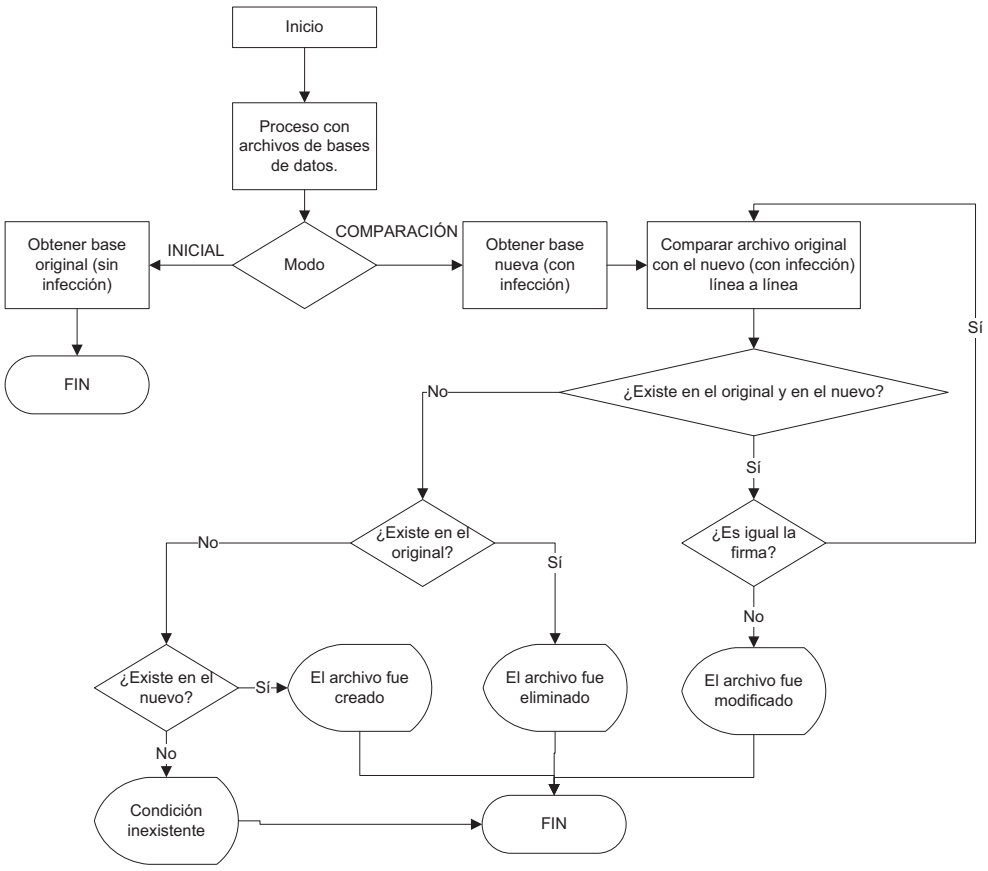

Fig. 3.2. Diagrama de flujo del proceso de comparación y obtención de la base de datos del sistema de archivos.

En el anexo 3 puede observarse la estructura completa del archivo /forensics/compare.pl. Como sigue, en la muestra 3.1, es la forma en que luce una de las bases de datos de archivos generadas por esta herramienta:

```
C:/Archivos de programa/MSN Gaming Zone/Windows/chkrzm.exe|f4a230d27eb0f4203ead0fee10c08d9b
C:/Archivos de programa/MSN Gaming Zone/Windows/Cmnclim.dll|5a5aef13fb00786d7b4d3ceb0aaac1c4
C:/Archivos de programa/MSN Gaming Zone/Windows/Cmnresm.dll|ce0a4954a515cd20a68f91c2b5c22804
C:/Archivos de programa/MSN Gaming Zone/Windows/hrtz.dll|2a0045ad010b014819c32d3e851714a3
C:/Archivos de programa/MSN Gaming Zone/Windows/Hrtzres.dll|1f1df9d64422cae806cf97bcb93efc50
C:/Archivos de programa/MSN Gaming Zone/Windows/hrtzzm.exe|6c83027a211213ddac15ce41305ce6cc
C:/Archivos de programa/MSN Gaming Zone/Windows/rvse.dll|2837511173d1de0ff3d5822ef14ad6db
C:/Archivos de programa/MSN Gaming Zone/Windows/Rvseres.dll|aac55cd7242f5c8bf7657c8c050a1aef
C:/Archivos de programa/MSN Gaming Zone/Windows/Rvsezm.exe|3f54b446356d3679b66df92350877646
C:/Archivos de programa/MSN Gaming Zone/Windows/shvl.dll|d029db42a62a72a2f63e18632d3ceec8
C:/Archivos de programa/MSN Gaming Zone/Windows/Shvlres.dll|8cdc125e9644f8a1285ce3454d03a18c
C:/Archivos de programa/MSN Gaming Zone/Windows/shvlzm.exe|16ad25828c946bb9b22da8f5981cd759
C:/Archivos de programa/MSN Gaming Zone/Windows/UniAnsi.dll|091bf3f1ea7888bb5615ed5f2f48a7ae
```

Muestra. 3.1. Fragmento de la base de datos del sistema de archivos.

3.4 Herramienta de análisis del registro de Windws

La herramienta para el análisis del registro tiene como objetivo encontrar los cambios realizados en las llaves del registro bajo las acciones de creación, modificación y eliminación.

Los scripts descritos en este apartado fueron totalmente desarrollados y similares a los del apartado anterior. Fue necesaria su total edición porque el método que empleaba el trabajo original era prácticamente no funcional.

La herramienta “dumphive” obtiene un volcado del registro de Windows desde el GNU/Linux, sin embargo, los archivos no son de lo más manejables. Es por ello, que esta herramienta de análisis del registro se ayuda de otro pequeño programa que vuelve a los volcados del registro más amigables y entendibles. El programa que desempeña la función anterior se llama /forensics/ajustar.sh y se basa en una serie de expresiones regulares para la obtención de un resultado adecuado. Su estructura está detallada en el anexo 4. La forma en que se encuentra el archivo antes de la aplicación del anterior script es la siguiente (muestra 3.2).

```
[default\AppEvents\EventLabels\Maximize]
@="Maximizar"
"DispFileName"="@mmsys.cpl,-5833"

[default\AppEvents\EventLabels\MenuCommand]
@="Comando de menú"
"DispFileName"="@mmsys.cpl,-5834"

[default\AppEvents\EventLabels\MenuPopup]
@="Menú emergente"
"DispFileName"="@mmsys.cpl,-5835"

[default\AppEvents\EventLabels\Minimize]
@="Minimizar"
"DispFileName"="@mmsys.cpl,-5836"
```

Muestra. 3.2. Fragmento del volcado del registro de Microsoft Windows.

De manera resumida realiza las siguientes acciones:

- Duplica el archivo original y trabaja las transformaciones sobre la copia.
- Realiza una sustitución de las líneas en blanco por el conjunto “####”.
- Sustituye los cadenas “]\r” por “]####”.
- Elimina los retornos de carro “\r”.
- Elimina los saltos de línea de los registros con valores hexadecimales, para que sean de una sola línea.
- Sustituye el conjunto “,\s\s” por “,”. Donde “\s” representa un espacio.
- Elimina una línea en específico, la cual contiene el patrón “REDEDIT4”.
- Sustituye todos los saltos de línea “\n” por “||”.
- Sustituye el patrón “#####||” por un salto de línea “\n”.
- Sustituye el patrón “####||” por “###”.
- Finalmente elimina las líneas en blanco.

En conclusión, en esta normalización de los volcados del registro de Microsoft Windows XP se logra colocar en una sola línea cada llave de registro y enseguida sus registros que dicha llave contenga, quedando la estructura de la siguiente manera: [Nombre_Llave]###"valor1"||"valor2"||...||"valorN"||. Siendo el resultado fácilmente manipulable para las comparaciones posteriores. A continuación en la muestra 3.3 se aprecia el resultado final.

```
[default\AppEvents\EventLabels\Maximize]###@="Maximizar"||"DispFileName"="@mmsys.cpl,-5833"||
[default\AppEvents\EventLabels\MenuCommand]###@="Comando de menú"||"DispFileName"="@mmsys.cpl,-5834"||
[default\AppEvents\EventLabels\MenuPopup]###@="Menú emergente"||"DispFileName"="@mmsys.cpl,-5835"||
[default\AppEvents\EventLabels\Minimize]###@="Minimizar"||"DispFileName"="@mmsys.cpl,-5836"||
[default\AppEvents\EventLabels\Navigating]###@="Iniciar exploración"||"DispFileName"="@mmsys.cpl,-5838"||
[default\AppEvents\EventLabels\Open]###@="Abrir programa"||"DispFileName"="@mmsys.cpl,-5839"||
[default\AppEvents\EventLabels\SystemAsterisk]###@="Asterisco"||"DispFileName"="@mmsys.cpl,-5843"||
[default\AppEvents\EventLabels\SystemExclamation]###@="Exclamación"||"DispFileName"="@mmsys.cpl,-5845"||
[default\AppEvents\EventLabels\SystemExit]###@="Salir de Windows"||"DispFileName"="@mmsys.cpl,-5846"||
[default\AppEvents\EventLabels\SystemHand]###@="Parada crítica"||"DispFileName"="@mmsys.cpl,-5847"||
```

Muestra. 3.3. Fragmento del volcado del registro después del tratamiento.

El código completo del archivo /forensics/comparer.pl se encuentra en el anexo 5.

3.5 Herramienta de recopilación de información

La herramienta “/forensics/forensics.sh” es un programa escrito en bash, encargado de establecer todas las llamadas a los diversos programas de análisis, con lo que se puede decir que centraliza la ejecución del análisis por completo. Su labor es fundamental y de no existir no se generarían lo

reportes. TRUMAN original contaba con un script del mismo nombre, similar estructura y mismo proósito, sin embargo, por la agregación de las demás funcionalidades y herramientas fue necesaria su re-edición en un 80% aproximadamente. Las actividades que realiza son las siguientes:

- Ejecutar el scritp /fauxservers/stop.sh
- Detener el servicio de apache2.
- Maneja el archivo /tmp/go.txt, el cual contiene el nombre del último malware ejecutado.
- Monta la imagen del sistema operativo que ha sido contaminado para después extraer la información del sistema de archivos, del registro de Windows, de los procesos y de las conexiones.
- Obtiene las firmas MD5 y SHA1 del binario malicioso.
- Invoca los scripts y comandos necesarios para las comparaciones. Con ello logra ejecutar la comparación del estado antes y después de la infección.
- Invoca el script para el análisis de tráfico.
- General el reporte final en formato “txt”.
- Lo envía a una ubicación de red y por correo electrónico.
- Desmonta la imagen contaminada y levanta nuevamente el servicio de apache2.

En resumen este programa activa las ejecuciones de la comparación de los sistemas de archivos, la comparación de los registros, la comparación de procesos y conexiones de red, el análisis de tráfico, la generación del reporte, la difusión de los mismos por sus distintos métodos y demás funciones de control que sirven para que TRUMAN siga su flujo de ejecución. Su código fuente se puede ver en el anexo 6. (Fig. 3.3)

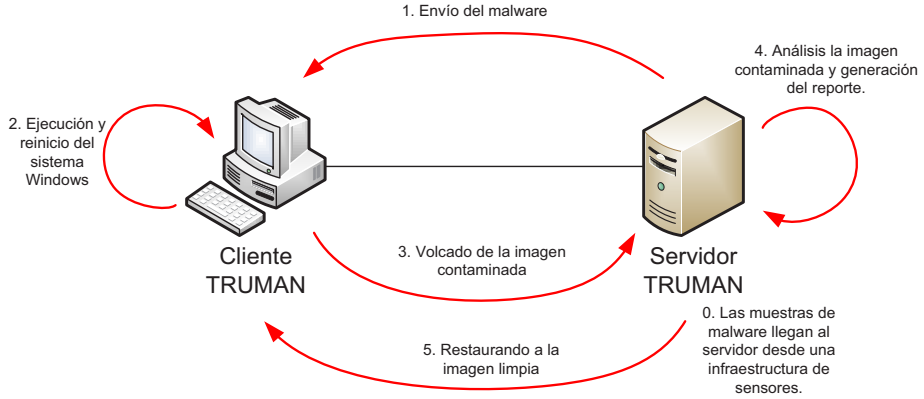

Fig. 3.3. Proceso gráfico de la ejecución y análisis de una muestra de malware.

3.6 Herramienta para el análisis de tráfico de red

Es una herramienta especializada en el análisis del tráfico de red generado por los códigos maliciosos y hasta el momento entrega información de los servidores HTTP y FTP consultados, si es que los hubiera. Se localiza en /forensics/traffic.sh. Al igual entrega un despliegue de conexiones a servidores IRC junto con sus datos de conexión, si es que existiesen. Finalmente entrega un reporte de otro tipo de conexiones dejándolas en un apartado para otras conexiones. Su ejecución requiere la previa instalación de Snort en su versión 2.8.4.1 y tcpflow. Este script fue desarrollado completamente; no existía en la implementación original. El código fuente de esta herramienta se halla en el anexo 7. A continuación un diagrama de flujo que describe este proceso. (Fig. 3.4)

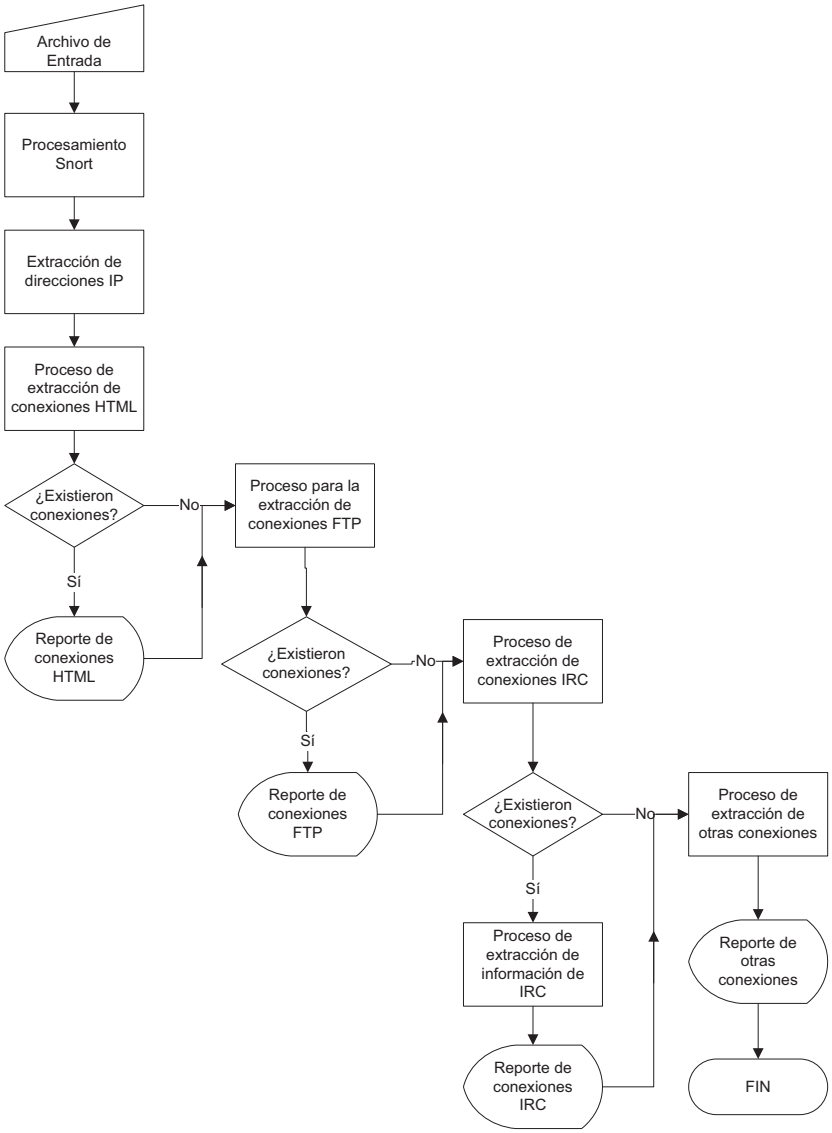

Fig. 3.4. Diagrama de flujo del proceso de análisis de la captura de tráfico de red de la herramienta en la ejecución de un malware.

3.7 Herramientas para la automatización de procesos

Los programas /etc/init.d/services.sh, /fauxservers/start.sh, /fauxservers/stop.sh y /usr/local/apache2/cgi-bin/truman.cgi son las herramientas que llevan a cabo las tareas de automatización de los procesos. De los últimos cuatro scripts services.sh fue totalmente desarrollado; start.sh y stop.sh fueron reeditados; mientras que truman.cgi solo sufrió algunas modificaciones respecto a su original aplicación.

El programa /etc/init.d/services.sh se encarga de monitorear el cambio del archivo /fauxservers/start.flag, el cual es modificado por /usr/local/apache2/cgi-bin/truman.cgi al descargarse un código malicioso en el equipo cliente. Al manifestarse un cambio en el archivo /fauxservers/start.flag, el programa /etc/init.d/services.sh dispara a /fauxservers/start.sh, el cual ejecuta un comando para activar la captura de tráfico y activa el firewall con NAT ubicado en /etc/init.d/nat. El /etc/init.d/services.sh también invoca a /fauxservers/stop.sh cuando ha finalizado el proceso de ejecución del software malicioso. Consultar anexo 8 para revisar el código fuente de cada una de estas herramientas. Enseguida un diagrama del funcionamiento de este proceso. (Fig. 3.5)

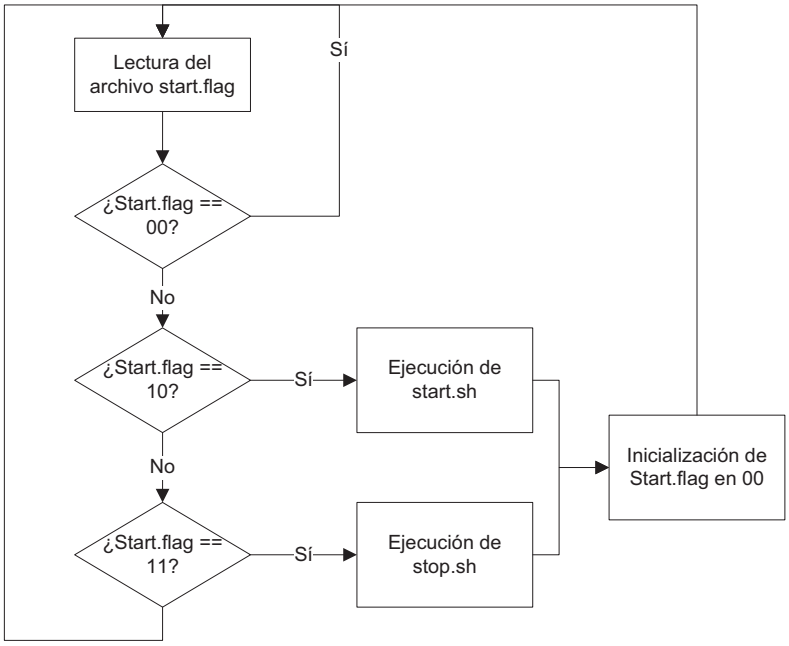

Fig. 3.5. Diagrama de flujo del funcionamiento del script /etc/init.d/services.sh.

3.8 Herramientas para la administración de reportes

Las herramientas de administración de reportes son, básicamente, para la difusión de éstos a través del uso de medios que conviene al DSC/UNAM-CERT. Es decir, fueron requerimientos del proyecto que los reportes de difundieran por correo electrónico y que se respalden en una ubicación de red; todo de manera automática y transparente. Estas herramientas se desarrollaron completamente, ya que no se contaba con algo similar antes de comenzar a realizarlo. (Fig. 3.6)

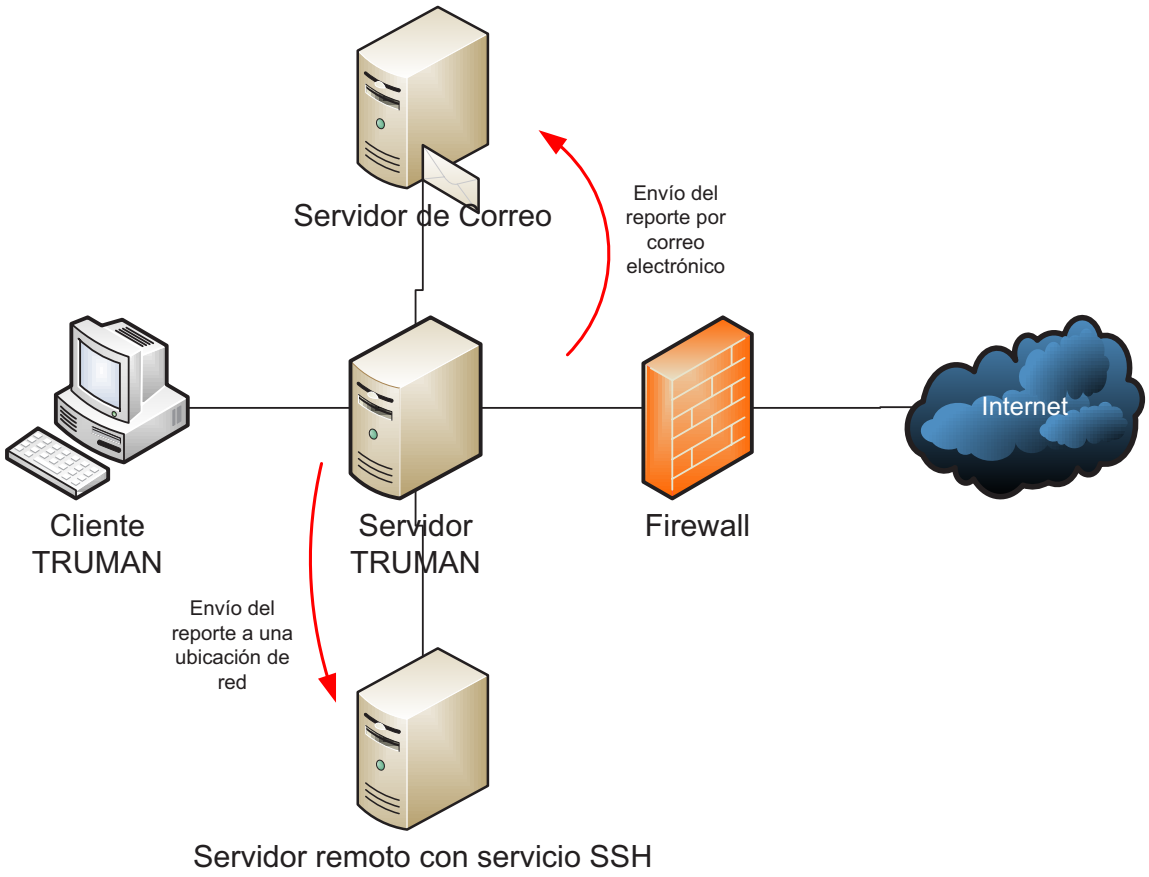

Fig. 3.6. Proceso gráfico de la administración de reportes.

3.8.1 Difusión por correo electrónico

Con el uso del servidor de correo electrónico sendmail y el lenguaje de programación Perl se ha logrado la difusión del reporte vía correo electrónico. El programa que lo hace se llama /forensics/mailreport.pl y su código está disponible en el anexo 9.

3.8.2 Difusión en ubicaciones de red

La difusión en una ubicación de red conlleva ciertas configuraciones tanto del lado del servidor TRUMAN como de la ubicación a donde se requieran mandar. De tal manera, que cuando se

necesite enviar un reporte a dicha ubicación no sea necesario el ingreso de credenciales de acceso. Se sugiere consultar el tutorial del anexo 10 para establecer las configuraciones necesarias.

Y finalmente la línea de código utilizada para este propósito es como sigue:

```
scp -r -q $FPATH/ malware-unam@quimera.seguridad.unam.mx:/home/malware-unam/TRUMAN/Reportes/
```

Donde "$FPATH/" es la carpeta a copiar, "malware-unam@quimera.seguridad.unam.mx" es el usuario y dominio del equipo (también puede ser dirección IP) y ":/home/malware-unam/TRUMAN/Reportes/" es el directorio donde se copiarán.