1. Antecedentes teóricos

1.1 El concepto de código malicioso

De manera general y considerando los objetivos de los códigos maliciosos se puede emitir la siguiente definición:

"Un código malicioso es un conjunto de instrucciones que se ejecuta en una computadora y provoca que el sistema del usuario víctima haga algo que un atacante quiere que este sistema realice. En el idioma inglés se le conoce como 'malware', que proviene de la contracción de 'Malicious Software'".

El código malicioso instalado en una computadora brinda al atacante control parcial o total sobre el equipo. Si un atacante puede instalar un código malicioso sobre los equipos de cómputo o engañar a los usuarios para que ejecuten un programa malicioso, dichos equipos actuarán como subordinados del usuario malintencionado. Al mismo tiempo, los sistemas podrían no responder a sus comandos normales. Entonces esos sistemas de cómputo estarían comprometidos y actuando como dobles agentes obedeciendo las órdenes de sus autores.

¿Quién necesita un colaborador humano introducido en una organización cuando un atacante puede usar un código malicioso para ejecutar rutinas o procesos sobre los equipos del interior de la misma? Los seres humanos infiltrados en alguna organización podrían ser atrapados, arrestados e interrogados. Un código malicioso, por otro lado, posiblemente sólo sea descubierto, analizado y eliminado. Si la organización es de negocios comerciales, una institución educativa, una agencia de gobierno o una división militar, un código malicioso puede realizar algo realmente dañino; como extraer, alterar e incluso eliminar información.

Aunque muchos códigos maliciosos son implementados en programas ejecutables los atacantes tienen a su disposición una gran variedad lenguajes de scripting, macro lenguajes de procesadores de palabras y algún otro tipo de instrucciones para crear software malicioso.

En los siguientes renglones se mencionan las actividades que el software malicioso puede provocar que una computadora haga comúnmente:

- Borrar archivos de configuración sensibles del disco duro, dejando a las computadoras completamente inoperables.
- Infectar la computadora y ser como un punto de salto para propagarse a las computadoras de los amigos, vecinos y contantos en general del usuario víctima.
- Monitorear las teclas presionadas y dejar que el atacante vea todo lo que se captura.
- Recolectar información acerca del usuario víctima, sus hábitos en cómputo, los sitios web que visita, el tiempo que permanece conectado y demás cosas.
- Enviar un flujo de video de la pantalla de la computadora del usuario víctima al atacante, quien puede, prácticamente, mirar lo que el usuario observa.
- Grabar video desde una cámara adjunta o audio desde el micrófono del sistema y enviarlo al atacante a través de la red, viendo y escuchando todo lo que sucede.
- Ejecutar comandos del atacante en el sistema de la víctima, justo como si el usuario mismo los hubiera ejecutado.
- Robar archivos de la máquina de la víctima, especialmente algunos con contenido delicado como información personal, financiera, o crítica.
- Colocar archivos en el sistema de la víctima, así como código malicioso adicional, información robada, software no original, pornografía, convirtiendo el sistema atacado en un auténtico repositorio de archivos ilícitos para ser accedidos por otros.
- Usar el sistema comprometido como un punto de salto para atacar a otra máquina, ocultando la fuente real del ataque para evadir a la ley.
- Establecer el escenario para un crimen, haciendo que toda la evidencia de un delito cometido por un atacante parezca señalar a la víctima y a su computadora.
- Ocultar las actividades de un atacante en su sistema, enmascarando la presencia del mismo por medio de esconder archivos, procesos y conexiones de red.

La lista anterior es una pequeña parte de ejemplos de lo que puede llegar a realizar un atacante por medio de una infección por códigos maliciosos.

1.2 Comportamiento general de un código malicioso

El primer objetivo para un atacante que ha creado un software malicioso es lograr infectar equipos. Para ello se vale de numerosas técnicas de propagación. Las más significativas se podrían enumerar como sigue:

- Aprovechando vulnerabilidades remotas.
- Por correo electrónico.
- Por dispositivos extraíbles como memorias USB, discos de 3.5 pulgadas, discos compactos, entre otros.
- Por sitios en Internet.

Al haber logrado ejecutarse en los equipos víctimas, después buscan garantizar su permanencia en el mismo (copiarse, clonarse, propagarse, alterar archivos y llaves de registro). Los directorios en donde buscarán alojarse son, por lo regular, directorios donde se encuentran archivos del sistema operativos como C:\WINDOWS\ o C:\WINDOWS\System32\ y también usarán otras ubicaciones muy frecuentemente. Las llaves de registro que serán comúnmente alteradas serán \HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run y \HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services; aunque ésas dos serían las más comunes, también podrían verse afectadas más.

1.3 Clasificación de códigos maliciosos

En las décadas de los ochentas y noventas comenzarón a aparecer códigos maliciosos importantes. Podían llevar a cabo las tareas de ejecución de programas en contra de aplicaciones web, explotación de vulnerabilidades de desbordamiento de pila, envío de programas por correo electrónico, sobrescritura del sistema operativo, escalación de privilegios, entre otras. Y cada forma y técnica empleada era distinta y no menos creativa que las otras.

Hablar de una clasificación es algo sensato, pues no todos los códigos maliciosos se comportan de la misma manera, tal y como se vio en el parrafo anterior. Existen unos que llevan una sola actividad, otros desempeñan dos o más y las combinan a discreción. A continuación se ofrece una clasificación de manera general:

- Botnets.
- Bots.

- Caballo de Troya (Trojan horse).
- Combinación de dos dos más tipos de códigos maliciosos.
- Exploit.
- Gusano (Worm).
- Inundadores (Flooders).
- Keyloggers.
- Mailers y Mass-Mailers.
- Pharming.
- Phishings.
- Programas generadores de correo spam.
- Puerta trasera (Backdoor).
- RootKit a nivel de núcleo.
- RootKit a nivel usuario.
- Virus.

No son todos los tipos de códigos maliciosos existentes, sin embargo, son los más significativos históricamente y que en la actualidad se manifiestan con más frecuencia. Si se desea conocer un panorama más amplio de estas clasificaciones, así como sus definiciones se deben consultar las secciones de referencias y glosario respectivamente.

1.4 El concepto de caja de arena

Una definición formal habla de un ambiente restringido en el cual ciertas funciones están prohibidas. Por ejemplo, borrado de archivos y modificación de información del sistema como pueden ser parámetros del registro, además de otras funciones del panel de control, las cuales pueden estar prohibidas. Las cajas de arena son utilizadas cuando un código ejecutable ha venido desde una fuente externa que no es completamente confiable, los cuales pueden tratarse de códigos maliciosos.

De acuerdo con la seguridad en cómputo, una caja de arena es un mecanismo de seguridad para separar programas en ejecución. Es comúnmente usada para ejecutar código no probado, o programas no confiables provenientes de terceras partes no verificadas, proveedores y usuarios no confiables.

La caja de arena típicamente proporciona un amplio conjunto controlado de recursos para que programas ajenos se ejecuten en él, semejante a un espacio dedicado en disco, memoria y procesador. El acceso a red, la habilidad para inspeccionar el sistema host o la acción de leer desde dispositivos de entrada son, usualmente, capacidades anuladas o altamente restringidas. En este sentido, las cajas de arena pueden verse como un ejemplo de virtualización.

Algunos ejemplos de cajas de arena disponibles en Internet son:

- Anubis. Se trata de un proyecto patrocinado por Secure Business Austria y desarrollado por el Internacional Secure Systems Laboratory. Es un pequeño equipo de profesionales de la seguridad haciendo investigación en el campo de la seguridad en cómputo y análisis de malware. Su objetivo es proporcionar a los usuarios interesados y avanzados una herramienta que ayude a combatir los códigos maliciosos. Ésa es la razón por la que brindan el servicio gratuitamente. La dirección donde se puede obtener más información y llegar a utilizar la herramienta es: http://anubis.iseclab.org/?action=home
- Joebox. Es una simple aplicación de caja de arena con un único y especial concepto. Está diseñada para el análisis de comportamiento automático de códigos maliciosos en sistemas operativos basados en Windows. Promete amplias características, reportes sustanciales y tiene un costo si quieres tu propia infraestructura, sin embargo, se ofrece una versión de demostración o bien la utilización en línea. Su página para obtener más información es: http://www.joebox.org/index.php
- Threatexpert. Se autocalifica como un avanzado y automatizado sistema de análisis de amenazas (ATAS) diseñado para analizar e informar el comportamiento de virus, gusanos, troyanos, adware, spyware y otros riesgos de seguridad relacionados mediante un modo completamente automático. Está disponible en línea y no cuesta utilizarla. Para más información se puede consultar la liga: http://www.threatexpert.com/
- Norman sandbox. Es una tecnología que esta compañía incluye en sus soluciones de antivirus. Consiste en un ambiente virtual donde los programas pueden ejecutarse en un plano controlado sin interferir con los procesos reales, archivos de programa o entorno de red. Si un programa ejecuta acciones que la solución califica como sospechosa, ésta misma lo etiqueta como un programa malicioso. Se trata de una firma noruega, la cual vende este tipo de soluciones entre otras. Mayor información consultando: http://www.norman.com/about_norman/es

- CWsandbox. Es una herramienta para el análisis de código malicioso que logra reunir la automatización, efectividad y corrección en una sola entidad funcional y precisa. Puede accederse gratuitamente vía web y se puede comprar una licencia para tenerla en el lugar de trabajo. Se pueden ver más detalles en: http://cwsandbox.org/?site=1&page=home

Si bien es cierto, el tener este tipo de soluciones es de gran ayuda a la investigación en el campo del análisis de códigos maliciosos, sin embargo, se apegan a la idea de una red de arena con ciertas limitantes y las implementaciones físicas son discutibles. Puede existir malware que detecte estos ambientes y se inhiba. Podría ser un amplio proyecto de investigación y desarrollo, el cual implica costos y esfuerzos. Es por eso que en este informe se propone una idea más básica pero completa. La red de arena, lo cual es asunto del siguiente punto.

1.5 El concepto de red de arena

Una red de arena es una infraestructura de red mínima que trabaja en el modelo TCP/IP y con la cual se pueden realizar análisis de códigos maliciosos, de tal forma que éstos encuentren un ambiente completamente real y exista mucho menor riesgo de inhibición. Por lo que sería posible obtener los cambios en el sistema de archivos, el registro, procesos, conexiones e incluso una captura de tráfico.

Entonces enumerando los elementos (infraestructura mínima) es necesario tener un servidor para proporcionar la salida a Internet, por lo menos un equipo cliente; herramientas de monitoreo y análisis de archivos, registro, procesos, conexiones y tráfico de red; sobretodo y lo más importante, muestras de códigos maliciosos para analizar.

Para comprender el concepto de la red de arena se debe regresar a la concepción del escenario donde, normalmente, los códigos maliciosos se ejecutan. Entonces se prentende simular la infraestructura de una red doméstica donde se cuenta con un dispositivo de comunicaciones, el cual brinda el servicio de conexión a Internet y detrás de él están los equipos de los usuarios, los cuales utilizan este servicio.

Una situación ejemplo es cuando un usuario recibe un correo electrónico que avisa sobre una tarjeta digital disponible para él en alguna ubicación de la red mundial en Internet. Al entrar a la página web el usuario mira la tarjeta, pero también y en segundo plano una vulnerabilidad de su navegador es explotada y convierte su equipo en parte de una red robot (botnet) internacional mediante el levantamiento de un cliente de protocolo IRC que envía información al atacante.

1.6 Infraestructura mínima de la red de arena

La infraestructura que se propone para construir en este informe es la básica. Es decir, se debe contar con ese servidor que se menciona en el apartado anterior y al menos poseer un cliente en la red de arena. También se requiere acceso a Internet en el servidor.

El servidor deberá contar con dos interfaces de red físicas Ethernet; una para conectarse directamente al cliente y la otra para conectarse a la red interna o Internet. Un disco duro mínimo de 10 Gb; de tal forma que al menos tenga la capacidad de almacenar dos imágenes del sistema cliente (imagen limpia e imagen contaminada) independientemente de su propio sistema operativo. Memoria RAM considerable, al menos 512 Mb, aunque es recomendable 1 Gb. Un procesador de cualquiera de las tecnologías Core, Core Duo y Core2 Duo; ya que desempeñará varios procesos que requieren de memoria y buena capacidad de procesamiento. En el caso exclusivamente del servidor, se puede implementar desde máquina virtual, aunque su desempeño se verá afectado, por lo que se recomienda que sea real. En el ámbito lógico se puede decir que tendrá un sistema operativo GNU/Linux (Debian 4 Etch en particular, porque los programas fueron diseñados bajo esta plataforma originalmente) además de requerir la instalación de diversas herramientas para su completo funcionamiento, sin embargo, ese asunto será ampliamente explicado en la sección de apéndices de este informe.

El cliente debe ser un equipo estrictamente real con sistema Windows XP SP2 (aunque podría ser otra versión, es necesario que se verifique la compatibilidad de algunos componentes). Dicho sistema operativo deberá ser instalado sobre una partición de disco primaria con una magnitud que no rebase los 2 Gb de memoria de disco duro; cabe señalar que sí se puede destinar más memoria de disco duro, sin embargo, las imágenes que se almacenarán en el servidor serán más grandes y entonces también será necesario contar con más espacio en el disco duro del servidor. La memoria RAM puede ir desde 128 Mb hasta 256 Mb; no es necesario contar con más memoria RAM, lo cual deja abierta la posibilidad de que no se emplee un equipo nuevo para esta función. Aunque se asegure que debe ser una instalación en máquina real no se descarta la posibilidad de que sea virtual, sin embargo, se somete a las limitaciones que se explican en la justificación de este informe.

A continuación, en la figura 1.1, se ofrece un diagrama de cómo se vería una implementación mínima.

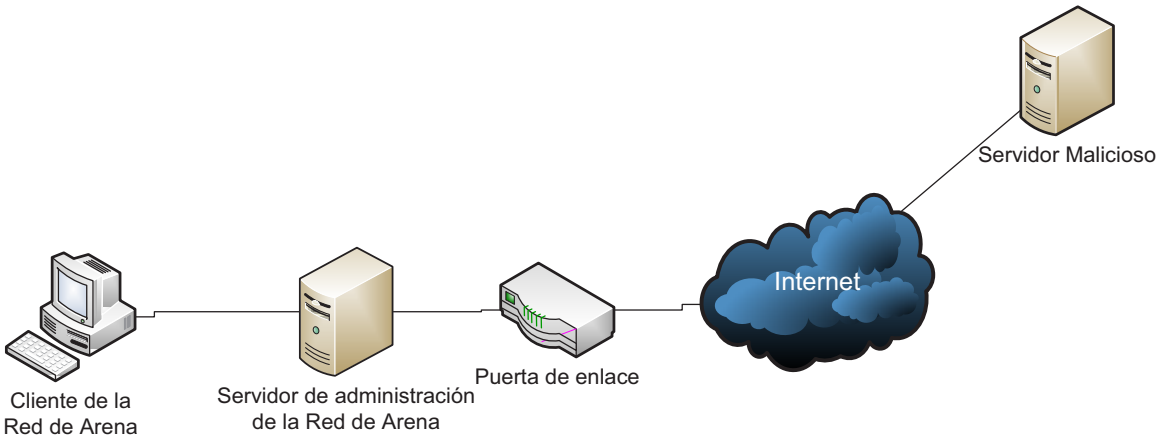

Fig. 1.1. Infraestructura mínima de una red de arena.

1.7 Introducción a las herramientas de análisis de códigos maliciosos

Existe una lista extensa de herramientas que pueden ser utilizadas para el análisis de códigos maliciosos. Y no es propósito de este informe explicar cada una de ellas, aunque sería sumamente interesante. Sin embargo, a continuación se describen las herramientas que se consideran más utilizadas en el análisis de muestras de software malicioso.

Sysanalyzer. Es una aplicación automática de análisis de códigos maliciosos en tiempo de ejecución que monitorea varios aspectos del sistema y estados de procesos. Fue diseñada para construir y mostrar un reporte con el detalle de las acciones llevadas a cabo en el sistema operativo por parte del código malicioso en cuestión. Puede monitorear y comparar procesos en ejecución, puertos abiertos, dispositivos cargados, bibliotecas inyectadas, cambios en llaves de registro, modificación de archivos y tráfico HTTP, IRC y DNS. Aunque parece una herramienta completa tiene ciertos detalles de funcionalidad, sin mencionar que no se ha actualizado últimamente. (Fig. 1.2)

Fig. 1.2. Vista Sysnalizer.

Autoruns. Es una utilería que mantiene el conocimiento de las locaciones de auto inicio como cualquier otro monitor de arranque, muestra qué programas están configurados para ejecutarse

durante el arranque del sistema o en el proceso de inicio de sesión, y muestra las entradas en el orden en que Windows las procesa. Estos programas incluyen binarios en la carpeta de arranque, Run, RunOnce, y otras llaves de registro. Es funcional con Windows ME, Windows XP y Windows Vista. (Fig. 1.3)

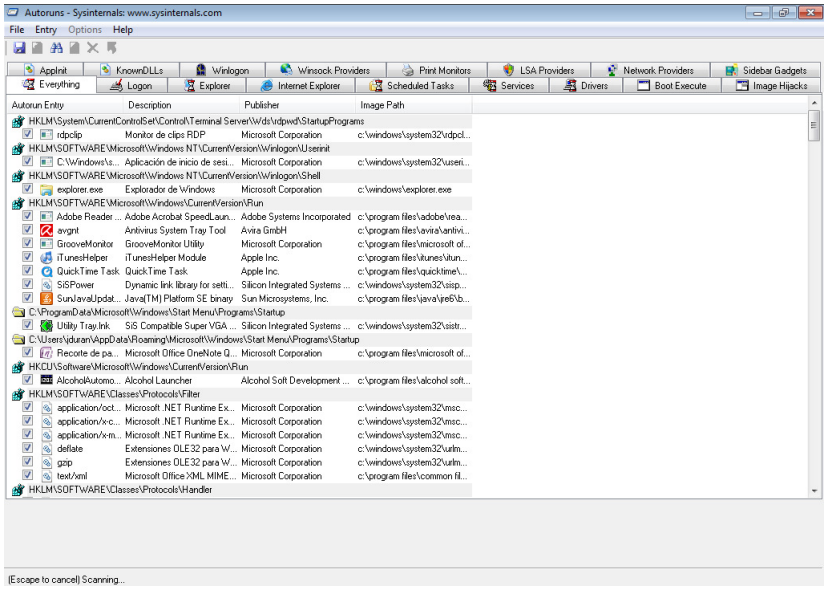

Fig. 1.3. Vista Autoruns.

Process Explorer. Es una herramienta que muestra una lista de los procesos activos actualmente, incluyendo los nombres de las cuentas a las que pertenecen. Posee dos modos para mostrar las cosas. También permite ver los manejadores de los procesos y el mapeo de memoria de las DLLs de los procesos en ejecución. También es válida para múltiples versiones de Windows. (Fig. 1.4)

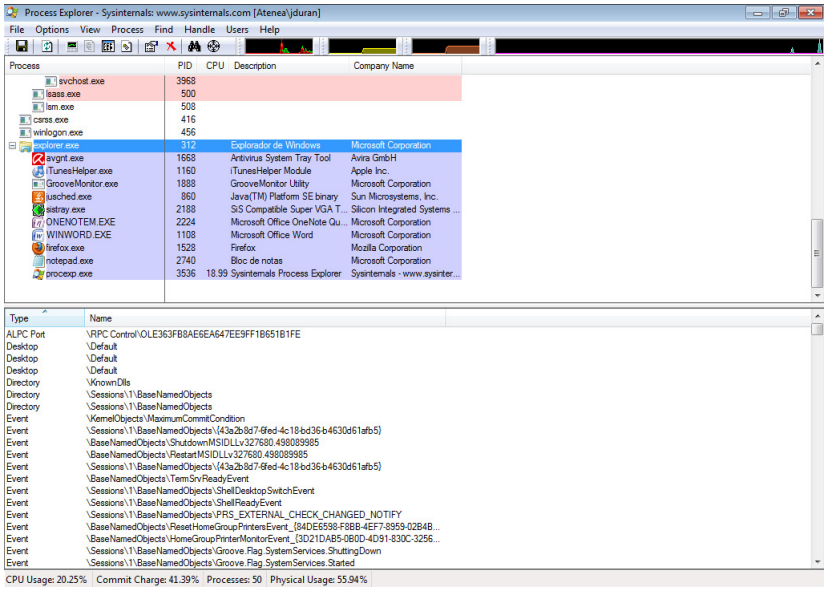

Fig. 1.4. Vista de Process Explorer.

Pslist. Forma parte del conjunto de herramientas pstools y su función es listar los procesos en ejecución del sistema a través de la línea de comando de Windows. (Fig. 1.5)

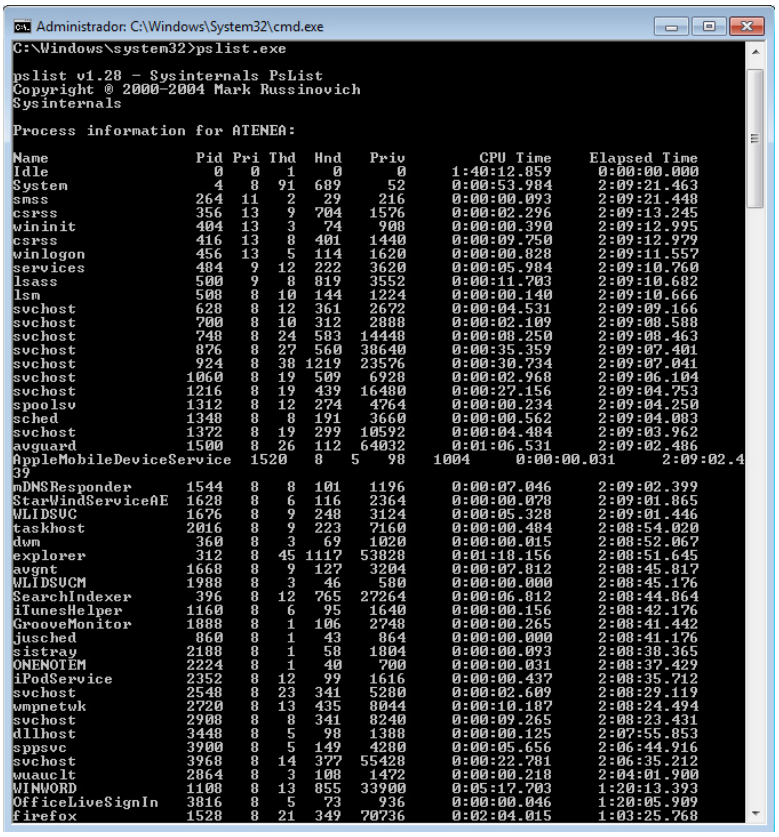

Fig. 1.5. Ejecución de pslist.exe en Windows Vista.

Tasklist. Se comporta como una alternativa a la herramienta pslist, pues lleva cabo la misma función. (Fig. 1.6)

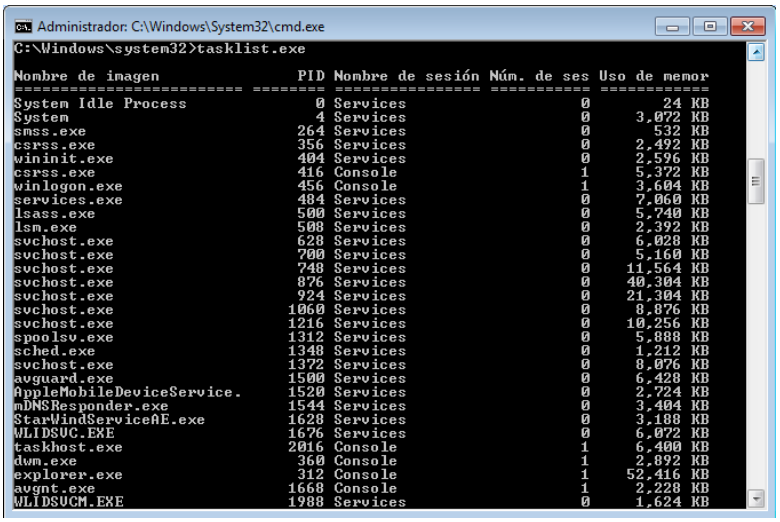

Fig. 1.6. Ejecución de tasklist.exe en Windows Vista.

Netstat. Esta herramienta informa el estado de las conexiones de red que el sistema operativo mantiene. Es de especial utilidad cuando los códigos maliciosos descargan complementos de Internet, o bien, cuando establecen sesiones remotas por algún protocolo. (Fig. 1.7)

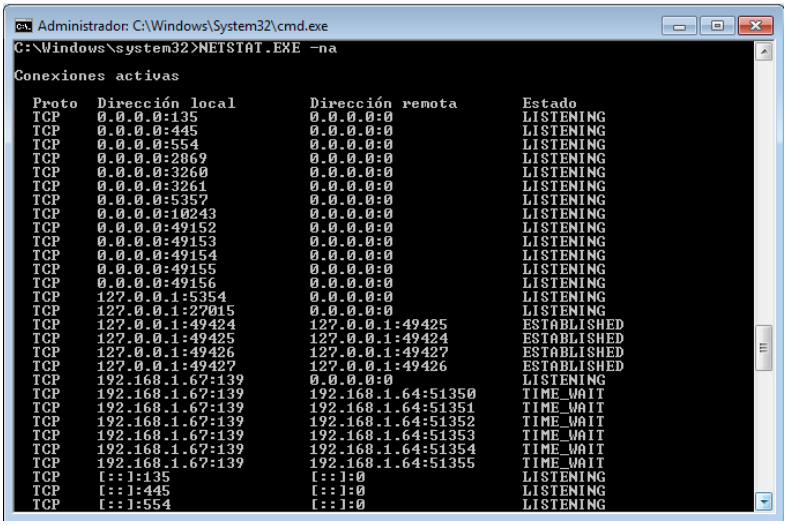

Fig. 1.7. Ejecución de netstat.exe -na en Windows Vista.

1.8 Introducción a la herramienta TRUMAN

The Reusable Unknown Malware Analysis Net proveniente de las siglas TRUMAN en inglés o La Red Reusable de Análisis de Malware Desconocido es un proyecto desarrollado por Joe Stewart y bajo los términos de la licencia GPL; la única versión que existe para el año 2010 es la 0.1. El proyecto TRUMAN es la parte medular de este informe, debido a que sin su desarrollo hubiera tomado mucho más tiempo obtener el resultado planeado.

Es un trabajo muy sencillo, pero funcional, de restauración del equipo infectado mediante el respaldo de imágenes del sistema operativo por red. También incluye herramientas básicas de análisis de códigos maliciosos, las cuales funcionan deficientemente.

Esta herramienta puede ser usada para construir una red de arena, para así analizar muestras de código malicioso en un ambiente aislado, incluso puede brindar un esquema simulado de Internet para la interacción con el código malicioso. Se ejecuta de manera nativa en un equipo, por lo que no es evadido por código malicioso que detecte entornos virtuales. TRUMAN automatiza parte del proceso de análisis permitiendo que el investigador sólo realice un mínimo de trabajo.

TRUMAN consiste en un servidor que contiene una imagen de un sistema operativo GNU/Linux, la cual puede ser capaz arracar desde un cliente a través de la red sin necesidad de hacer uso del

disco duro (originalmente basado en el trabajo de Chas Tomlins titulado “Imagen de Windows usando Linux”); también se compone de una colección de herramientas en forma de scripts. En el desarrollo posterior de este informe se observarán las mejoras sustanciales realizadas al proyecto original de TRUMAN y se verá que se logra una verdadera utilería para el campo del análisis de software malicioso. (Fig. 1.8)

Fig. 1.8. Pantalla inicial de arranque de TRUMAN.