UNIVERSIDAD DEL AZUAY

FACULTAD DE CIENCIA Y TECNOLOGÍA

ESCUELA DE INGENIERÍA ELECTRÓNICA

Implementación de un Honeypot mediante KIPPO para detectar acciones de un atacante al ganar acceso por SSH para mejorar la seguridad en la red de un servidor

Trabajo de graduación previo a la obtención del Título de Ingeniero Electrónico

Autor:

Carlos Alberto Castro Guerrero

Director:

Leopoldo Carlos Vázquez Rodríguez

CUENCA – ECUADOR

2011

DEDICATORIA:

A mis padres que siempre me han apoyado en todo y me han guiado durante todas las etapas de mi vida personal y académica.

AGRADECIMIENTO:

Ante todo a Dios, a mi familia y seres queridos que siempre me han apoyado para que cumpla mis metas, a la Universidad del Azuay y a sus docentes, de manera especial a mi Director de monografía Ing. Leopoldo Vázquez, y al Ing. Leonel Pérez, director de la Escuela de Electrónica, por el apoyo brindado.

Implementación de un Honeypot mediante KIPPO para detectar acciones de un atacante al ganar acceso por SSH para mejorar la seguridad en la red de un servidor.

Resumen:

Para la implementación de un tipo especial de Honeypot (trampa para atacantes) se utiliza KIPPO, que es un software diseñado para detectar acciones de los ataques, que son generados mediante máquinas virtuales en diferentes sistemas operativos a través de una conexión SSH. En el servidor se obtiene un registro detallado de ataques a la red en la que está conectado este servidor. Para construir la trampa de atacantes se implemento un Servidor Virtual en el sistema operativo Ubuntu y se emularon ataques ficticios empleando tres Maquinas Virtuales, con la finalidad de demostrar la efectividad del Honeypot diseñado.

Palabras claves:

Honeypot, KIPPO, SSH, seguridad, redes.

Ing. Leopoldo Carlos Vázquez Rodríguez

Carlos Alberto Castro Guerrero

KIPPO based Honeypot implementation to detect attacking actions via SSH access in order to improve server's network security.

Abstract:

For the implementation of a special type of Honeypot (attacker's tramp) is used KIPPO, which is a software designed to detect attack actions, which are generated by virtual machines on different operating systems through an SSH connection. In the server is obtained detailed log info of attacks to the network where it is connected. To build the attacker's tramp an Ubuntu based Virtual Server was implemented and fictitious attackers were emulated by three Virtual Machines; in order to prove the effectiveness of the designed Honeypot.

Keywords:

Honeypot, KIPPO, SSH, security, networks

Ing. Leopoldo Carlos Vázquez Rodríguez

Carlos Alberto Castro Guerrero

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

Castro Guerrero Carlos Alberto
Trabajo de Graduación
Ing. Leopoldo Vázquez.
Septiembre del 2011

Implementación de un Honeypot mediante KIPPO para detectar acciones de un atacante al ganar acceso por SSH para mejorar la seguridad en la red de un servidor.

INTRODUCCIÓN

Es muy común hoy en día el ataque a las redes de computadoras sean estas alámbricas o inalámbricas, en especial en donde los atacantes puedan encontrar información valiosa, y para esto debemos contar siquiera con un cierto grado de seguridad.

Existen mecanismos de defensa como firewalls, sistemas de detección de intrusos (IDS), redes privadas virtuales (VPNs), listas de control de acceso, etc. Los problemas con estos mecanismos de seguridad se producen cuando no están correctamente configurados, y pueden dar una falsa sensación de seguridad. El uso de una tecnología llamada Honeypots permite conocer con detalle los ataques y vulnerabilidades de las redes.

En la actualidad existen diferentes tipos de redes, estas están interconectadas entre sí mediante internet, teniendo cierta red o máquina acceso a cualquier otra, en especial en infraestructura inalámbricas, con lo cual existe un alto índice de que intrusos quieran entrar en una red con el fin de causar daño o el robo de información; por lo cual es estrictamente necesario incrementar la seguridad en estos sistemas.

Tener información detallada acerca de actividades de intrusos es esencial, ya que nos ayudara a tomar medidas sobre los ataques sufridos y evitar futuros incidentes. Así como también nos proporciona información detallada sobre vulnerabilidades que tenemos en nuestra red con el fin de corregirlos.

Los Honeypots a través de KIPPO presentan una mejor alternativa a este problema. Definiremos Honeypot como un recurso de red destinado a ser atacado o comprometido. KIPPO es un tipo de Honeypot, destinado a capturar información detallada sobre ataques de intrusos, con sistemas, aplicaciones y servicios virtuales a ser comprometidos teniendo en cuenta que no son reales(los cuales no se encuentran en producción) y con costos muy por

debajo en comparación con otros tipos de Honeypots ya que no utiliza mucho hardware, beneficiando a todo servidor que este expuesto al mundo exterior.

CAPÍTULO I

INTRODUCCIÓN A LOS HONEYPOTS

1.1 ORIGEN DE SSH Y VULNERABILIDADES.

SSH (**S**ecure **SH**ell, en español: intérprete de órdenes segura) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, y también puede redirigir el tráfico de X para poder ejecutar programas gráficos si tenemos un Servidor X (en sistemas Unix y Windows) corriendo.

Además de la conexión a otros dispositivos, SSH nos permite copiar datos de forma segura (tanto ficheros sueltos como simular sesiones FTP cifradas), gestionar claves RSA para no escribir claves al conectar a los dispositivos y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH.

SSH trabaja de forma similar a como se hace con telnet. La diferencia principal es que SSH usa técnicas de cifrado que hacen que la información que viaja por el medio de comunicación vaya de manera no legible y ninguna tercera persona pueda descubrir el usuario y contraseña de la conexión ni lo que se escribe durante toda la sesión.

La seguridad no es patrimonio exclusivo en internet. Pruebas a esto: Al desarrollarse este software, que gozaba de buena reputación entre las empresas de Finlandia, fue pedido por compañías de Europa y de los Estados Unidos para ser incluido a su vez en otros programas de seguridad. Sin embargo, Core SDI encontró agujeros en la seguridad de este software, aparentemente infalible: descubrió que permitía que, a través de una serie de instrucciones y comandos, un extraño manejara en forma remota una computadora dentro de una intranet.

Esta empresa no solucionó el problema de seguridad del programa, sino que puso un "parche" que detecta cualquier ataque de intrusos en la red, activa una alarma y hace que enseguida se corten todas las conexiones. Ese parche se puso en internet para bajar gratis de su sitio en la web. Pero no obstante no es un sistema tan seguro como para lo que fue creado.

Para las empresas es muy importante contar con un software de protección confiable porque cada vez utilizan más las redes del tipo intranet, Internet y el correo electrónico para transmitir información.

La primera versión del protocolo y el programa eran libres y los creó un finlandés llamado Tatu Ylönen, pero su licencia fue cambiando y terminó apareciendo la compañía SSH Communications Security, que lo ofrecía gratuitamente para uso doméstico y académico, pero exigía el pago a otras empresas. En el año 1997 (dos años después de que se creara la primera versión) se propuso como borrador en la IETF (Internet Engineering Task Force; en español, Grupo Especial sobre Ingeniería de Internet).

A principios de 1999 se empezó a escribir una versión que se convertiría en la implementación libre por excelencia, la de OpenBSD, llamada OpenSSH. SSH (Secure SHell), www.openssh.com, es la herramienta de conexión segura más usada en el mundo Linux, no hay nada como SSH para conectarse a servidores remotos Linux, ya sea desde Internet o dentro de una LAN (Local Area Network). Todo el tráfico se encripta de punto a punto haciendo la conexión sumamente segura. Pero aun así siempre hay riesgos en el salvaje Internet, hackers black hat, script kiddies, crackers, mafias cibernéticas, etc. que en cuanto detectan un servidor SSH tratan de atacarlo por todos los medios posibles. Además, dentro de una LAN relativamente grande también se corren riesgos como el famoso tipo de ataque "man in the middle".

Básicamente, los ataques a SSH están basados en una situación (muy frecuente) de un servidor o demonio SSHD mal configurado o que no esté actualizado. Entonces, el objetivo es atacarlo mediante alguna vulnerabilidad descubierta a través de un escaneo de puertos o de ataques de login mediante fuerza bruta. Por ejemplo, una mala configuración sería permitir que el todopoderoso usuario root tuviera permiso de acceso al servidor SSH, esto será relativamente fácil de descubrir y la siguiente parte es lanzar un ataque de fuerza bruta con el objeto de adivinar la contraseña, esto claro mediante un script automático que realice esta función. La siguiente parte de la mala configuración sería no tener un número máximo de intentos de conexión pudiendo el atacante entonces lanzar hasta cientos de sesiones simultaneas de solicitud de ingreso y en cada una lanzar el script de fuerza bruta. Otro mal elemento de configuración sería el no limitar el número de intentos fallidos por conexión, etc. Y si a todo esto agregamos una contraseña débil de root (por ejemplo menos de 8 caracteres y solo minúsculas) será cuestión de posiblemente solo minutos o unas cuantas horas para lograr el objetivo.

1.2 HISTORIA ACERCA DE LOS HONEYPOTS.

Definición.

Se denomina Honeypot al software o conjunto de computadores cuya intención es atraer a atacantes, simulando ser sistemas vulnerables o débiles a los ataques. Es una herramienta de seguridad informática utilizada para recoger información sobre los atacantes y sus técnicas. Los Honeypots pueden distraer a los atacantes de las máquinas más importantes del sistema, y advertir rápidamente al administrador del sistema de un ataque, además de permitir un examen en profundidad del atacante, durante y después del ataque al Honeypot. Algunos Honeypots son programas que se limitan a simular sistemas operativos no existentes en la realidad y se les conoce como Honeypots de baja interacción y son usados fundamentalmente como medida de seguridad. Otros sin embargo trabajan sobre sistemas operativos reales y son capaces de reunir mucha más información; sus fines suelen ser de investigación y se los conoce como Honeypots de alta interacción.

Existen muchas definiciones para el término Honeypots, dependiendo de sus autores y usos. Un Honeypot es un sistema pasivo pero altamente dinámico que cambia de acuerdo a su utilización. Lance Spitzner en su libro “Honeypot: Tracking Hackers” define a los Honeypots generalizando sus características como sigue: "Un Honeypot es un recurso computacional altamente monitoreado, el cual se desea que sea probado, atacado o comprometido". En forma más precisa es definido como "recurso de un sistema de información, cuyo valor reside en el uso no autorizado o lícito del mismo".

Honeypots: la Historia

Los primeros conceptos que constituyeron la base de lo que actualmente conocemos como Honeypots se dieron a la luz a finales de los 80's e inicio de los 90's, publicaciones como “The Cuckoo‘s Egg” de Cliff Stoll y “An Evening with Berferd” *de Bill Cheswick, son las dos más importantes de esa época, y que incluyen conceptos sobre Honeypot”.*

Cliff Stoll fue un astrofísico que trabajó como administrador de sistemas en un laboratorio de California, que notó la discrepancia de 75 centavos en la facturación del uso de tiempos de computadora y gracias a su búsqueda de la razón de este error logra rastrear a un hacker que está intentando acceder a las redes de computadoras de América, usando sus computadoras intentaba hackear centenas de computadoras militares, industriales y académicas. The Cuckoo‘s Egg” fue publicado en 1988 y detalla la experiencia de Stoll a

través de los 3 años que duró el incidente en los cuales pudo observar al hacker y subsecuentemente obtener información que le permitió ayudar en su arresto.

El paper de Cheswick es una cronología de los movimientos de un hacker, describe los señuelos y trampas que utilizaron para detectar a un hacker, adicionalmente la construcción de una Cárcel Chroot que fue diseñada para monitorear las actividades del intruso.

En 1997, Fred Cohen publicó uno de los precursores de los actuales Honeypots de baja interacción, el "Deception Toolkit" (DTK). Consiste en una colección de scripts en PERL diseñados para sistemas UNIX, que emulan una variedad de conocidas vulnerabilidades. El concepto de "defensa engañosa" presentado por el DTK actualmente es el núcleo para la implementación de Honeypots. Usando un viejo sendmail (servidor de correos en Linux), con vulnerabilidad simulada, con falsos archivos de contraseñas, se buscaba atraer atacantes hacia el sistema, mientras perdía valioso tiempo e intentaba romper las contraseñas se protegía el verdadero sistema.

En 1998 sale a la luz el primer Honeypot comercial llamado "Cybercop Sting", corría bajo Microsoft Windows NT y simula un conjunto de diferentes dispositivos de red, tales como servidores Windows NT, servidores Unix y routers, con la capacidad de guardar y reportar cualquier actividad en la red a los administradores.

En 1998 también fue liberado "NetFacade" otro Honeypot comercial que podía simular toda una red de Clase C hasta 254 sistemas, además es capaz de simular 7 sistemas operativos diferentes con una gran variedad de servicios. "NetFacade" condujo al desarrollo de Snort IDS que actualmente juega un papel muy importante dentro de los Honeypots.

En 1999 un grupo de personas lideradas por Lance Spitzner fundaron "Honeynet Project", grupo sin fines de lucro dedicado a investigar la comunidad blackhat y compartir los resultados de sus investigaciones con otros.

En ese mismo año fue lanzado otro Honeypot comercial llamado "ManTrap" y ahora conocido como "Decoy Server", el cual simulaba una red con 4 diferentes máquinas para que el atacante pueda interactuar con ellas con la capacidad de generar tráfico y enviar e-mails entre los equipos simulados.

En el 2002, fue lanzado "Tiny Honeypot" por George Bakos, es un código simple en Perl que escucha en cada puerto TCP, registra toda la actividad de los mismos, y provee de respuestas a comandos que los atacantes emitan con el objetivo de obtener tiempo suficiente para que actúen los mecanismos de detección de intrusos. En ese mismo año se

lanza otro concepto en Honeypot por la compañía Google llamado “Google Hack Honeypot” GHH. El motor de Google indexa diariamente una cantidad enorme de sitios para que formen parte de las respuestas dentro de su exitoso buscador, pero en sitios web mal configurados Google puede llegar a indexar archivos muy sensibles y privados que pueden ser vistos por personas no autorizadas, pueden ser archivos de configuración, archivos de contraseñas, nombres de usuarios, números de tarjetas de crédito, etc. El GHH emula sitios vulnerables indexados por Google y recolecta información sobre los ataques a los portales web usando como herramienta este motor de búsqueda.

Desde 2007 la compañía Google viene trabajando en lo que se le conoce como “Google Project Hosting” la nueva versión de los trabajos realizados anteriormente y en una de sus actuales herramientas de “Google Code” para Honeypot denominado “KIPPO SSH Honeypot”.

1.3 INTRODUCCIÓN A KIPPO Y CARACTERÍSTICAS.

KIPPO es un Honeypot de interacción media a través de SSH diseñado para registrar los ataques de fuerza bruta y, sobre todo, la interacción realizada por el atacante. KIPPO es un proyecto de código abierto alojado en el Proyecto de Código de Google construido por Upi Tamminen. Este provee la estrategia de detectar los fallos y mejorar en la defensa cuando se usan en conjunto con otros mecanismos de seguridad. Al recoger información de las intrusiones y estudiarlas se puede conocer nuevas amenazas y herramientas aún no documentadas.

Características:

- Diseñado para registrar la capa de interacción entera realizada por el atacante.
- Basado en Python.
- Aparenta un sistema operativo parecido a Debian 5.
- Sistema de archivos falsos con la capacidad de añadir o eliminar archivos.
- Posibilidad de añadir contenido falso de archivos para el atacante utilizando "cat".
    - Por ejemplo: / etc / passwd, / etc / hosts, / proc / cpuinfo etc
- Posibilidad de añadir la salida del comando falsos.
    - Por ejemplo. / sbin / ifconfig, vi, ssh, useradd, apt-get, etc.

-Sesión TTY de registros almacenados para la reproducción fácil con los tiempos originales.

- Guarda los archivos descargados con wget para su posterior análisis.
- Tiene la opción de escribir los datos de ataque en SQL.
- Juega con trucos de la mente.

- Ssh pretende conectarse a algún lugar como salida.
- apt-get install pretende instalar cosas.
- Proporciona información sobre el atacante
    - agente SSH utilizado (putty, libssh, etc)
    - p0f-db (pasivo OS fingerprinting)
    - Ubicación del GEO Posible
    - Tiempos y conocimientos generales.

CAPÍTULO II

ANÁLISIS PARA LA IMPLEMENTACIÓN DE KIPPO EN UN SERVIDOR

2.1 ANÁLISIS DEL PROBLEMA DE SEGURIDAD INFORMÁTICA EN GENERAL EN LAS EMPRESAS DEL ECUADOR.

En nuestro país, la seguridad informática que se practica en las empresas es débil, no tenemos una cultura desarrollada, dentro de la estructura del personal de informática; la mayoría de las empresas no consideran una persona dedicada a las funciones de seguridad, sino que mezcla sus funciones con otras como la administración de la infraestructura, base de datos, incluso soporte técnico, obligando que se olvide las funciones de seguridad por cubrir las del día a día. No se concibe la idea de un OSI (Oficial de Seguridad informática), ni su correcto posicionamiento dentro del organigrama de la empresa.

Una práctica común en nuestro país es ahorrar costos escatimando en seguridad informática, cuando la empresa crece considerablemente o sospecha que existe robo y/o fuga de información, recién entonces se piensa en implementaciones de seguridad informática.

Con la masificación del uso de Internet, las empresas están conectadas permanentemente, así como el incremento de equipos y aplicaciones móviles mantienen a los usuarios conectados con su empresa, pero los peligros se incrementan considerablemente, las empresas deben entender que no basta con tener un firewall (de hecho algunas no cuentan con uno), haberlo configurado una vez y dejarlo así por siempre aunque cambie las aplicaciones y la infraestructura de la empresa.

También es una realidad que en las medianas y grandes empresas que han logrado crear una área de Seguridad Informática, estas pierden su rumbo, centrándose solo en los accesos, y limitadas por no poder señalar fallas dentro del área Informática, ya que tratan de evitar conflictos internos. La concientización en los usuarios es un factor importante, si no aplican las políticas de seguridad informática de la empresa, estas vendrían a no tener sentido, es decir para que sirve tener una serie de políticas si el usuario cuando falta o está de vacaciones facilita su clave a sus compañeros con el fin de "no afectar la productividad", ya que la mayoría considera engorroso los procedimientos para un reemplazo "temporal".

La Seguridad Informática en una empresa no depende solo del área informática, sino de los usuarios y principalmente de que la Gerencia de la empresa apoya las acciones de seguridad, periódicamente se debe revisar y actualizar los planes de Seguridad.

La parte positiva es que cada vez hay más profesionales en el área de Seguridad Informática, más empresas que prestan servicios, y las pequeñas y medianas empresas ya comienzan a interesarse por la seguridad de sus datos. La existencia de amenazas que afectan la disponibilidad, integridad y confidencialidad de los datos es real. Es crítico para las organizaciones poder identificar esas amenazas y adoptar recomendaciones que permitan prevenir, detectar y protegerse de ellas. La diversidad y la heterogeneidad de los sistemas de información que requieren las organizaciones actuales, sumado a la globalización a la que se enfrentan al conectar esos sistemas al mundo de Internet, genera incertidumbres en lo referente a la Seguridad de la Información.

Todos los días, en todo el mundo, las computadoras, llámense servidores, estaciones de trabajo o simplemente PCs, son violados y con ello expuesta la información de sus usuarios. Esta información puede ser las finanzas de la empresa, números de tarjetas de crédito, planes estratégicos, información relacionada con la investigación y el desarrollo de nuevos productos o servicios, etc.

El ataque a los sistemas puede ser por fines económicos, por obtener cierto tipo de información, para sabotear las operaciones de la empresa, para desmeritar el prestigio de la empresa, por revanchismo o simplemente por curiosidad, entre otras muchas causas. En cualquier caso, el riesgo e impacto son altos para una empresa. La pérdida de información sensitiva, fraude, paro de operaciones además de las pérdidas en imagen por el impacto publicitario que genere el ataque, representan altos costos para la empresa.

En ocasiones, las intrusiones no son detectadas debido a su naturaleza y porque el atacante, como todo intruso, se cuida de "no dejar huella" para seguir explotando el medio por el cual ingresa a los sistemas atacados. No hay que olvidar que el atacante es un experto en temas de computación, conoce y prueba los productos de software para buscar huecos de seguridad y crear las herramientas específicas para su explotación. Se enfoca pues a vulnerar la seguridad de los sistemas.

Gran parte de los administradores de sistemas basan la seguridad informática en el uso de contraseñas, otros utilizan herramientas como detectores de intrusos, firewalls, software especializado en seguridad. Sin embargo, a menudo los administradores de sistemas no son conscientes del peligro existente por cualquier cosa más allá de los ataques más triviales además de que muchas empresas parecen no querer disponer de los recursos para

valorar qué nivel de protección es adecuada para sus sistemas. Por ello, las intrusiones no autorizadas siguen sucediendo y muchas veces los administradores de sistemas ni se enteran.

Entre las causas comunes de la inseguridad de los sistemas se encuentran: los defectos potenciales de seguridad en la instalación del producto de software, en la configuración de los servicios de red, "huecos" típicos en las utilerías del sistema operativo y demás software base o de red, así como en la implantación de decisiones tácticas ignorantes de las condiciones mínimas de seguridad para los sistemas. Otros métodos que han resultado efectivos para un atacante son: la ingeniería social y el rompimiento de contraseñas.

2.2 PLANTEAMIENTO DE UNA SOLUCIÓN UTILIZANDO KIPPO.

Con el fin de tener un registro detallado de ataques a una red vamos a implementar un Honeypot especializado para esto que se llama KIPPO. Los Honeypots a través de KIPPO presentan una mejor alternativa a este problema. Definiremos Honeypot a un recurso informático, simulando ser un sistema en producción, su principal función es servir de carnada para monitorear todo uso ilícito del mismo.

KIPPO es un tipo de Honeypot, destinado a capturar información detallada sobre ataques de intrusos, con sistemas, aplicaciones y servicios virtuales a ser comprometidos teniendo en cuenta que no son reales(los cuales no se encuentran en producción) y con costos muy por debajo en comparación con otros tipos de Honeypots ya que no utiliza mucho hardware. Los ataques a los que puede estar expuesto un sistema pueden ser automatizados o realizados por humanos. En la prevención los Honeypots tienen más valor frente a los ataques automatizados, como gusanos (worms) o auto-rooters, los cuales se basan en herramientas de escaneo de redes enteras buscando vulnerabilidades para poder explotarlas de distintas maneras.

En un sistema ordinario con NIDS (Network Intrusion Detection Systems) es común que se presenten falsos positivos cuando los datos normales en el tráfico diario de la red pueden coincidir en el formato con algún ataque conocido y se generan alertas. Aunque no se produzcan alertas por parte del NIDS del Honeypot, si se analiza el tráfico registrado por el Honeypot será probable detectar nuevos ataques.

A medida que se pretende crear un servidor Linux hay que estar muy consciente en ver que muchos atacantes utilizan la funcionalidad de acceso remoto como una forma de ingresar

en este servidor, y como SSH es el factor estándar para el acceso de Linux como un objetivo prioritario para el ataque.

La instalación del programa no es tan difícil como aparenta, solo se necesita saber cada uno de los pasos necesarios para la instalación y funcionamiento; principalmente se basa en dos paquetes, el paquete de KIPPO y el paquete de phyton-twisted

Hay un par de archivos clave que se pueden editar para cambiar la sensación del sistema que se proporciona a los usuarios maliciosos:

- kippo.cfg contiene información de tiempo de ejecución incluida la ubicación de registro, nombre de host falso, etc.
- kippo.tac contiene los "usuarios" una matriz, que enumera el nombre de usuario y una contraseña que el inicio de sesión SSH emulado aceptará como "válidas".

El directorio “honeyfs /” va tan lejos como para que pueda crear un "verdadero" sistema de ficheros para que el usuario malicioso pueda interactuar con estos, posiblemente copiando el sistema de archivos en un servidor activo para ayudar a camuflar el sistema emulado.

Una interesante utilidad es “utils / playlog.py”. Esto le permite reproducir una sesión de terminal maliciosos en tiempo real, errores tipográficos y de todo, para proporcionar una verdadera idea de la interacción de los usuarios maliciosos con el período de sesiones. Siendo KIPPO un medio de interacción Honeypot SSH diseñado para registrar los ataques de fuerza bruta y, sobre todo, la interacción realizada por el atacante, se puede obtener la estrategia de detectar los fallos y mejorar en la defensa cuando se usan en conjunto con otros mecanismos de seguridad. Al recoger información de las intrusiones y estudiarlas se puede conocer nuevas amenazas y herramientas aún no documentadas.

CAPÍTULO III

IMPLEMENTACIÓN DE KIPPO EN UN SERVIDOR

3.1 CONFIGURACIÓN DE UN SERVIDOR LINUX.

Para construir una red virtual con un servidor, se dispone de un computador personal de las siguientes características:

- Procesador Intel Core 2 Duo
- Memoria RAM de 4GB
- Disco duro de 320GB
- Tarjeta de Red de 100Mbps

Las máquinas virtuales que se configuraran para formar la red contaran con los siguientes requerimientos de hardware:

SISTEMA OPERATIVO	DISCO DURO	MEMORIA RAM
Ubuntu 11.04	10GB	512MB
CentOS 5.3	10GB	512MB
Fedora 14	10GB	512MB
Windows 7	10GB	1024MB

Tabla 3.1.1 Configuración de Sistemas Operativos.

Las máquinas virtuales a crearse van a ser a través de un software que es el VMWARE el cual debe ser instalado y configurado correctamente, una guía completa de instalación y configuración del VMWARE se encuentra en el Anexo 1. Después de haber instalado y configurado el VMWARE, vamos a utilizarlo para crear las máquinas virtuales. Voy a crear una máquina virtual que va a ser un servidor, con el cual me voy a poder conectar desde cualquier parte de la red a través de máquinas clientes con el fin de poder controlar el servidor. El Sistema Operativo a ser instalado en el servidor va a ser Ubuntu 11.04; una guía completa de instalación y configuración de Ubuntu 11.04 se encuentra en el Anexo 2.

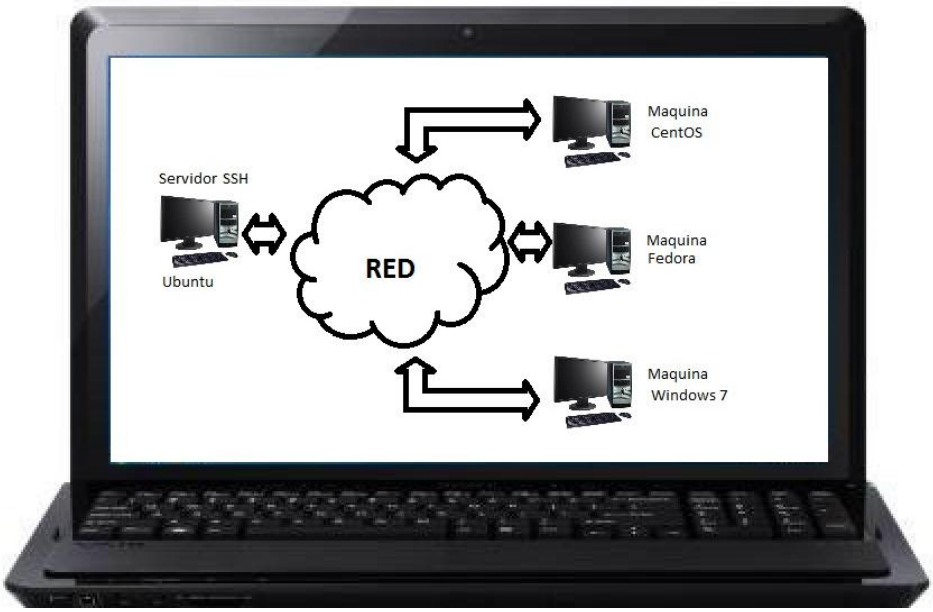

Figura 3.1.1 Red con un servidor SSH.

Una vez instalado el Sistema Operativo de Ubuntu, el cual lo vamos a utilizar como servidor SSH procedemos a hacer las respectivas configuraciones de red. Para instalar el servidor de SSH en nuestro sistema lo que se necesita es instalar el paquete openssh-server; para ello en una terminal y con conexión a internet para la descarga e instalación del software, introducimos:

$ sudo apt-get install openssh-server

Con lo cual se me va a instalar el servidor SSH, se muestra el ejemplo en la siguiente gráfica:

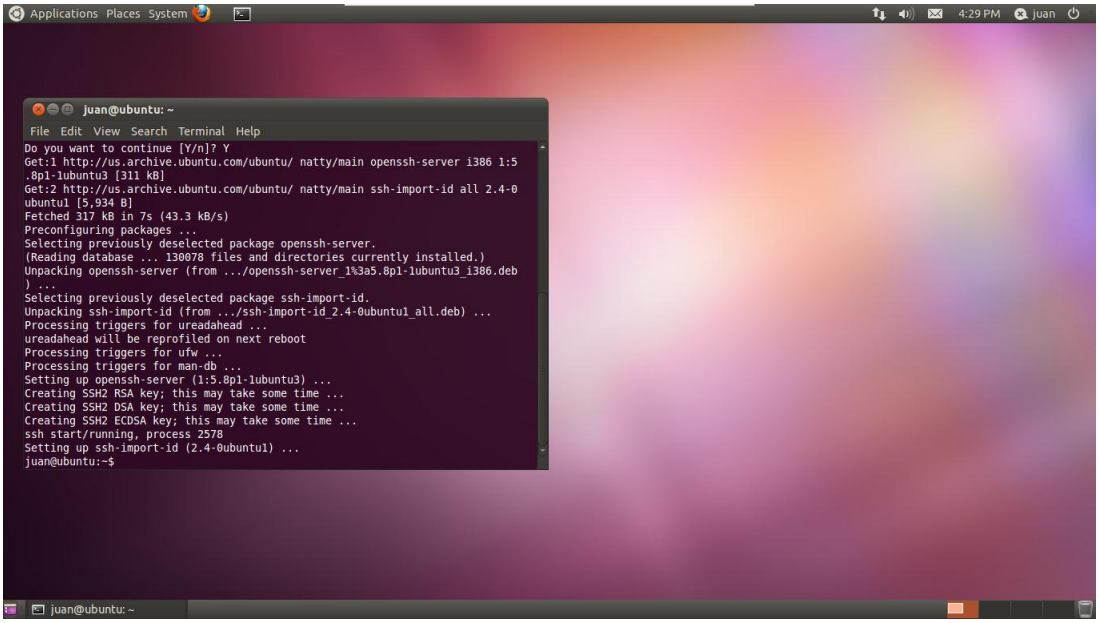

Figura 3.1.2 Instalación de openssh-server.

Para reiniciar el servicio de SSH en el servidor:

$ sudo /etc/init.d/ssh restart

Después de este proceso de instalación vamos a configurar un servidor DHCP, esto es con el fin de lograr una conexión de red virtual entre las máquinas virtuales existentes con el servidor virtual. Lo que haremos será asignar direcciones IP dentro de un rango determinado, por una cierta cantidad de tiempo, además se asignarán parámetros como la puerta de enlace.

Para configurar la tarjeta de red editamos el fichero /etc/network/interfaces:

$ sudo nano /etc/network/interfaces

Como necesitamos una IP "estática" para el servidor DHCP modificamos el archivo que quede de esta forma:

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
address 192.168.1.3
netmask 255.255.255.0
gateway 192.168.1.1
network 192.168.1.0
broadcast  192.168.1.255
```

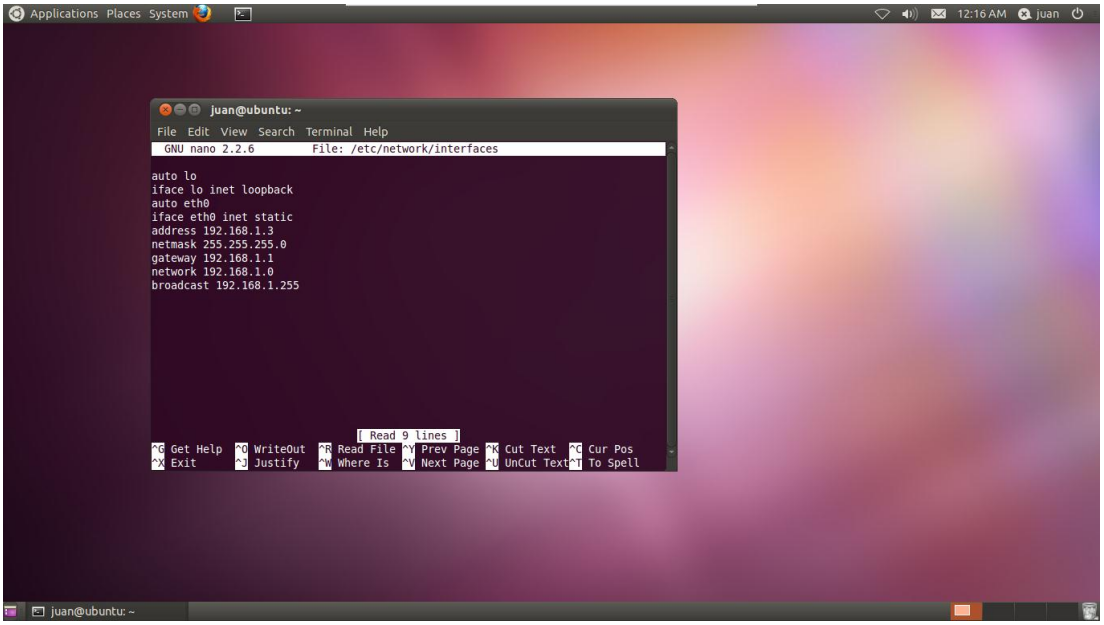

Figura 3.1.3 IP estática del servidor.

Para guardar correctamente los cambios reinicio el servicio de red:

$ sudo /etc/init.d/networking restart

Después de esto, en el archivo dhcpd.conf:

$ sudo nano /etc/dhcp/dhcpd.conf

Configuramos nuestro servidor DHCP escribiendo las siguientes líneas:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
range 192.168.1.20 192.168.1.100;
option domain-name-servers 192.168.1.2;
option domain-name "ubuntu.net";
option routers 192.168.1.3;
option broadcast-address 192.168.1.255;
default-lease-time 600;
max-lease-time 7200;
interfaces=eth0;
}
```

Una vez terminada la configuración de los comandos, iniciamos el servidor DHCP:

$ sudo /etc/init.d/isc-dhcp-server start

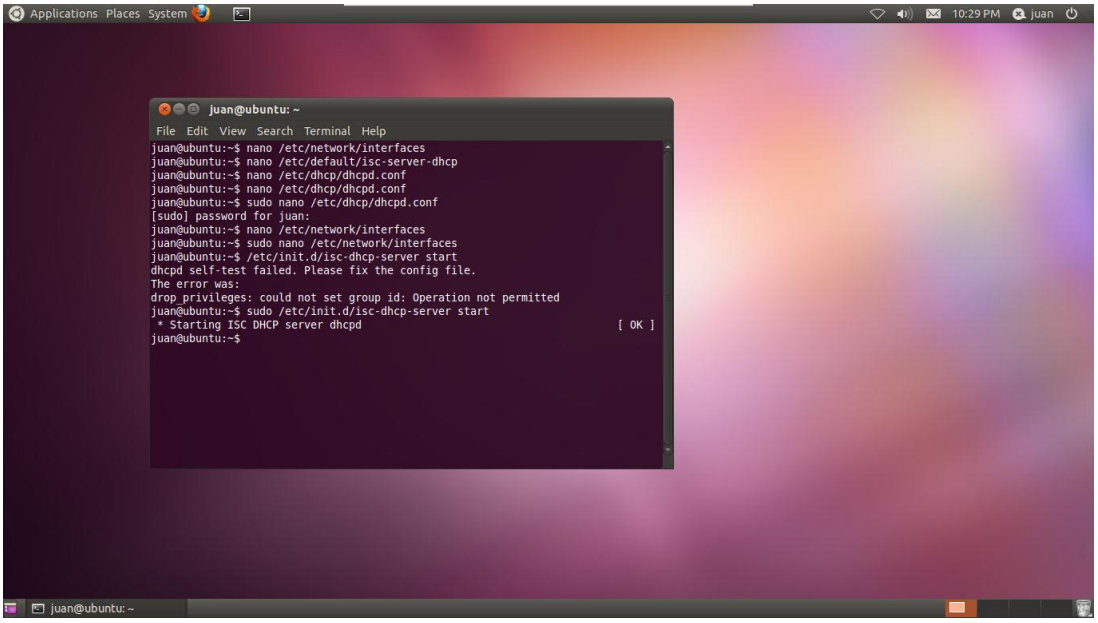

Figura 3.1.4 Configuración del servidor DHCP

Con esto tenemos instalado y configurado nuestro servidor SSH para que exista conectividad entre las maquinas virtuales y puedan conectarse las maquinas atacantes a través de SSH para controlar el servidor.

3.2 CONFIGURACIÓN DE LAS DIFERENTES MÁQUINAS QUE VAN A ATACAR AL SERVIDOR.

3.2.1 CONFIGURACIÓN DE CENTOS 5.3

El sistema operativo de CentOS 5.3 está instalado y configurado sobre una máquina virtual, de modo que me sirva como una de las máquinas a conectarse a la red y al servidor; una guía completa de instalación y configuración de CentOS 5.3 se encuentra en el Anexo 3. Una vez instalado CentOS vamos a verificar la conectividad con el servidor Ubuntu, para esto hacemos ping con el servidor para comprobar la conexión de red.

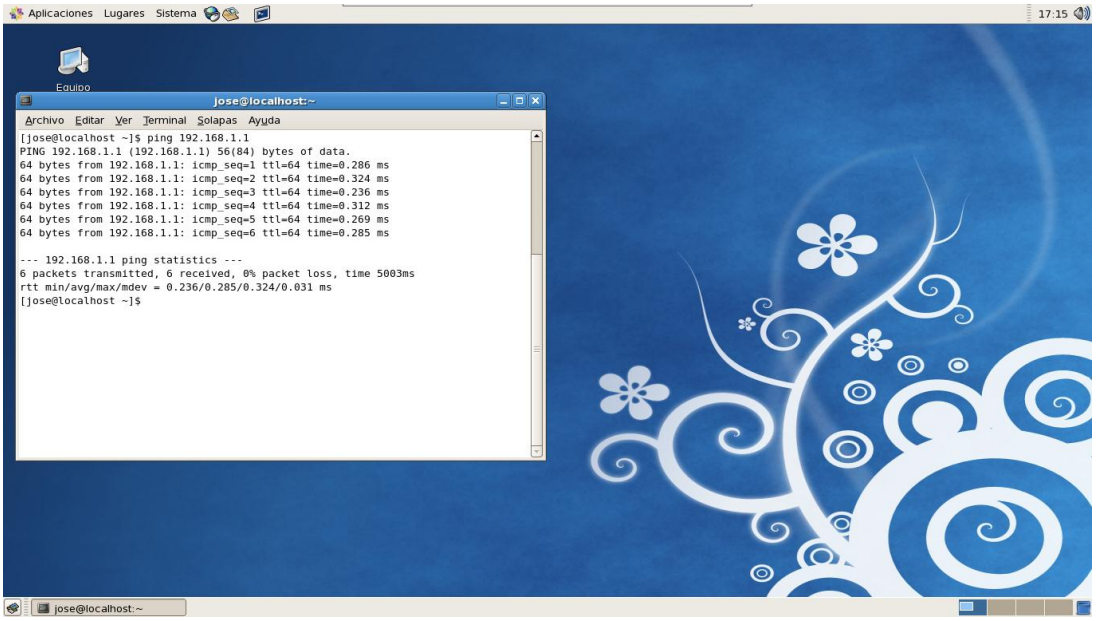

Figura 3.2.1.1 Conexión CentOS con el Servidor.

Para saber cuál es nuestra IP, introducimos en una ventana de terminal:

$ /sbin/ifconfig

Después de esto no necesitamos configurar el cliente SSH que ya viene instalado por defecto. Solo necesitamos conectarnos por SSH ingresando el nombre de usuario y la dirección del Servidor, como en el ejemplo:

$ ssh juan@192.168.1.1 (nombre_de_usuario@dirección_del_servidor)

Ingresamos la contraseña en el momento en que se nos pide y con esto ya estamos conectados al servidor mediante SSH, como se ilustra la siguiente gráfica:

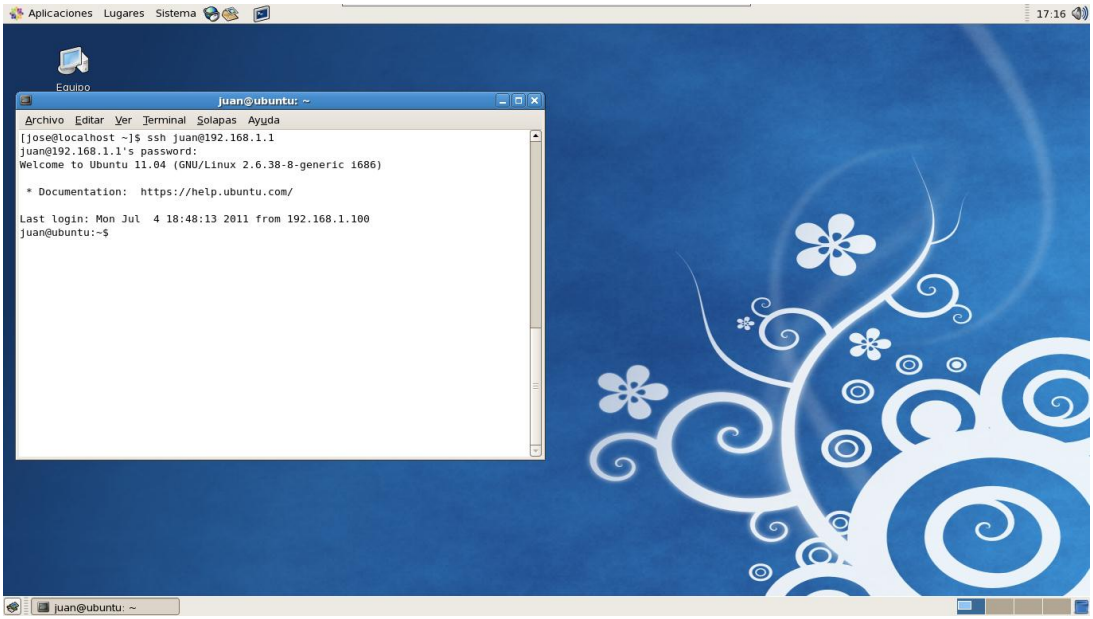

Figura 3.2.1.2 Conexión SSH al servidor desde CentOS.

3.2.2 CONFIGURACIÓN DE FEDORA 14

El sistema operativo de Fedora 14 está instalado y configurado sobre una máquina virtual, de modo que me sirva como una de las máquinas a conectarse a la red y al servidor; una guía completa de instalación y configuración de Fedora 14 se encuentra en el Anexo 4. Una vez instalado Fedora vamos a verificar la conectividad con el servidor Ubuntu, para esto hacemos ping con el servidor para comprobar la conexión de red.

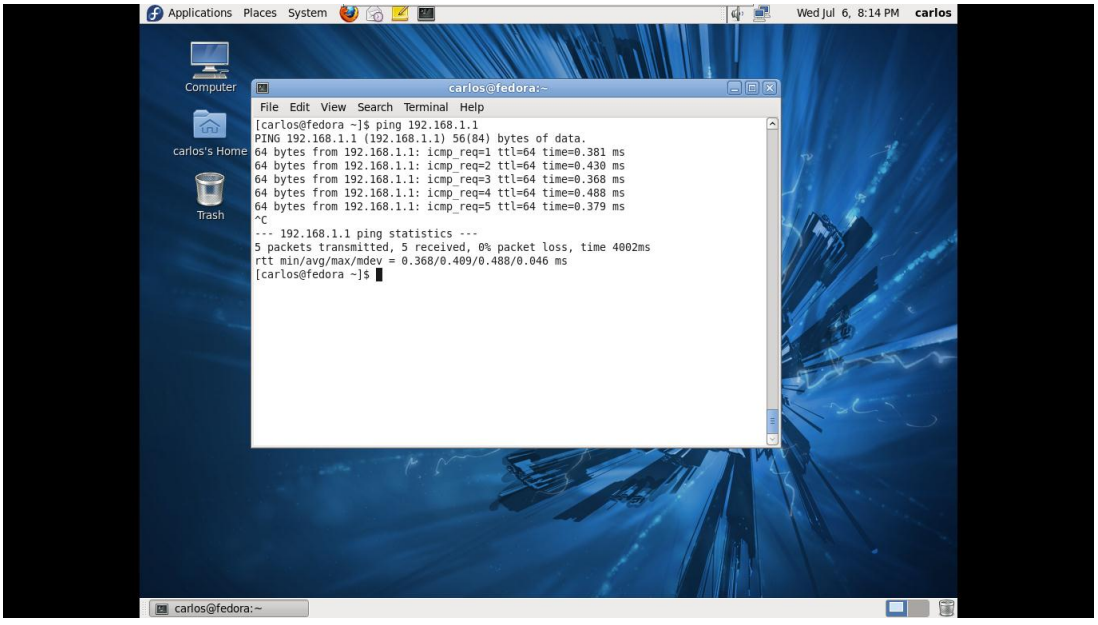

Figura 3.2.2.1 Conexión Fedora con el Servidor.

Para saber cuál es nuestra IP, introducimos en una ventana de terminal:

$ ifconfig

Después de esto no necesitamos configurar el cliente SSH que ya viene instalado por defecto. Solo necesitamos conectarnos por SSH ingresando el nombre de usuario y la dirección del Servidor.

$ ssh juan@192.168.1.1 (nombre_de_usuario@dirección_del_servidor)

Ingresamos la contraseña en el momento en que se nos pide y con esto ya estamos conectados al servidor mediante SSH, como se ilustra la siguiente gráfica:

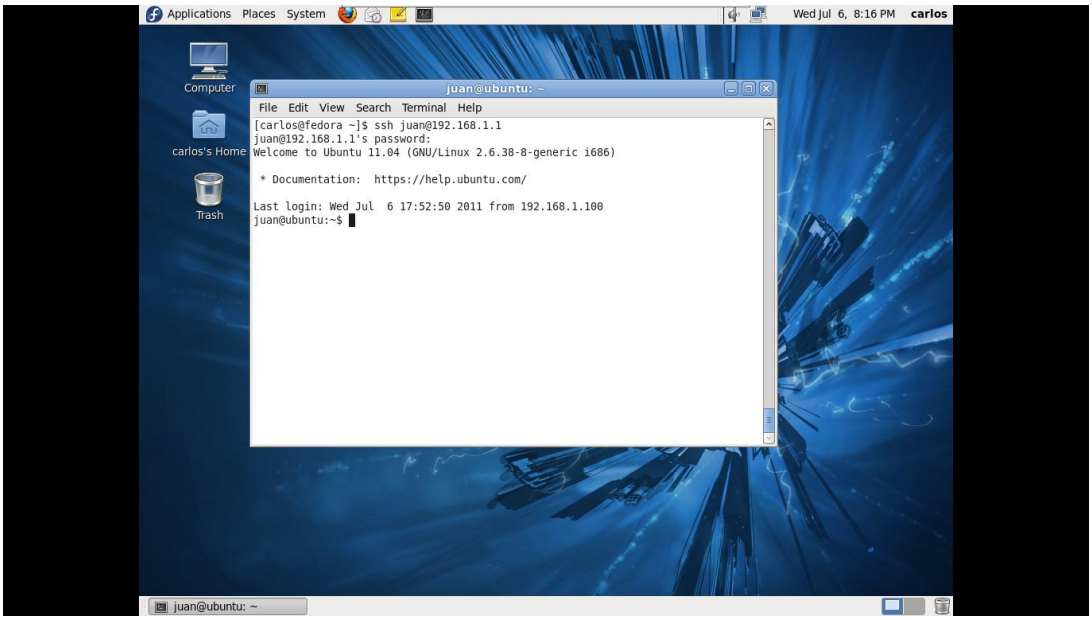

Figura 3.2.2.2 Conexión SSH al servidor desde Fedora.

3.2.3 CONFIGURACIÓN DE WINDOWS 7

El sistema operativo de Windows 7 está instalado y configurado sobre una máquina virtual, de modo que me sirva como una de las máquinas a conectarse a la red y al servidor; una guía completa de instalación y configuración de Windows 7 se encuentra en el Anexo 5. Una vez instalado Windows 7 vamos a verificar la conectividad con el servidor Ubuntu, en este caso tenemos que conectarnos manualmente con el servidor.

Vamos a Inicio/Panel de control/Redes e Internet/Centro de redes y recursos compartidos. Ingresamos en conexión de área local, luego en Propiedades.

Después, ingresamos a propiedades del Protocolo de Internet versión 4 (TCP/IPv4), y ajustamos los valores manualmente.

Lo principal es elegir una IP que no se encuentre en uso como por ejemplo:

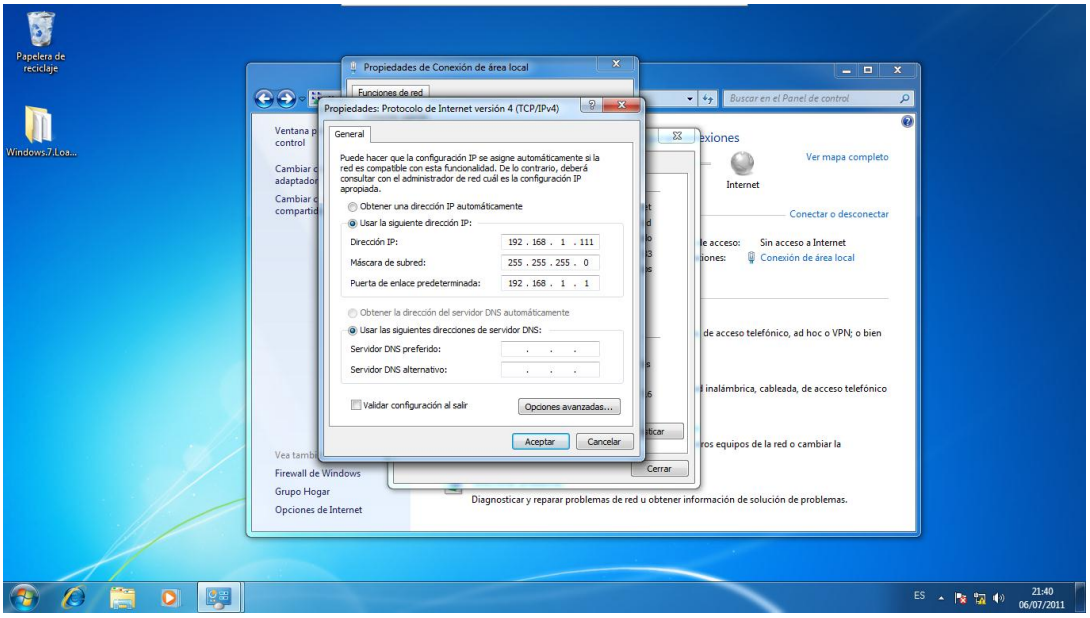

Figura 3.2.3.1 Configuración de red de Windows 7.

Configurada la Dirección IP, la máscara de subred y la puerta de enlace, aceptamos y cerramos todas las ventanas.

A continuación abrimos un editor de comandos y para comprobar la conexión hacemos ping con el servidor.

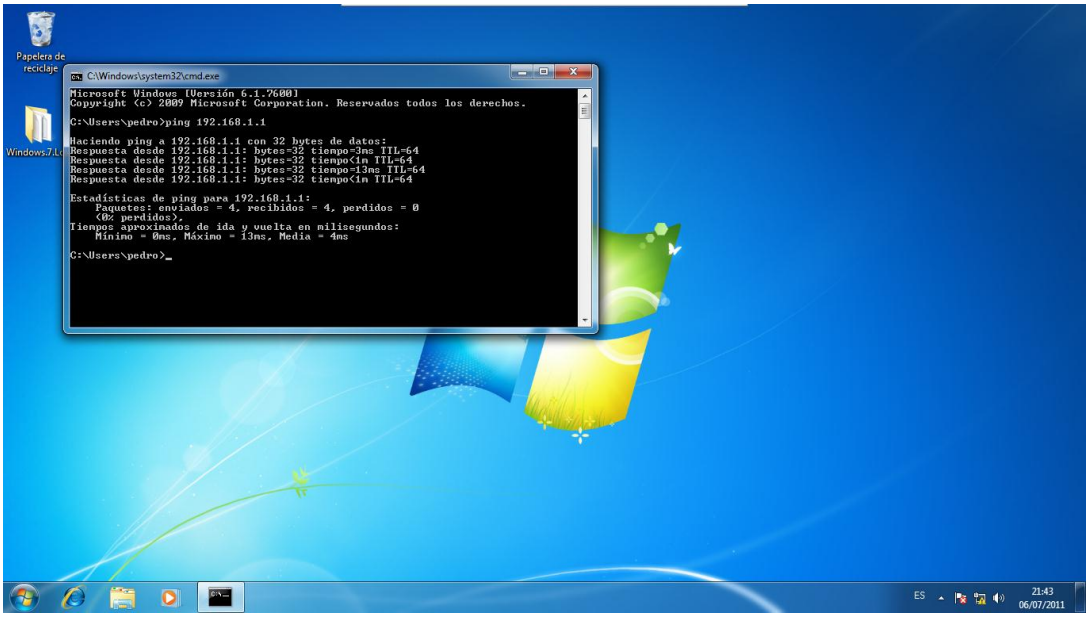

Figura 3.2.3.2 Conexión Windows 7 con el Servidor.

Después de esto vamos a necesitar configurar el cliente SSH ya que en Windows no viene instalado por defecto. Para esto vamos a utilizar un programa denominado PuTTY; se trata de un cliente SSH para Windows que permite acceder en modo texto al sistema Linux desde sistemas Windows.

Configuramos los valores de la IP a la que nos vamos a conectar y su puerto (el puerto SSH es el 22 a través de Windows).

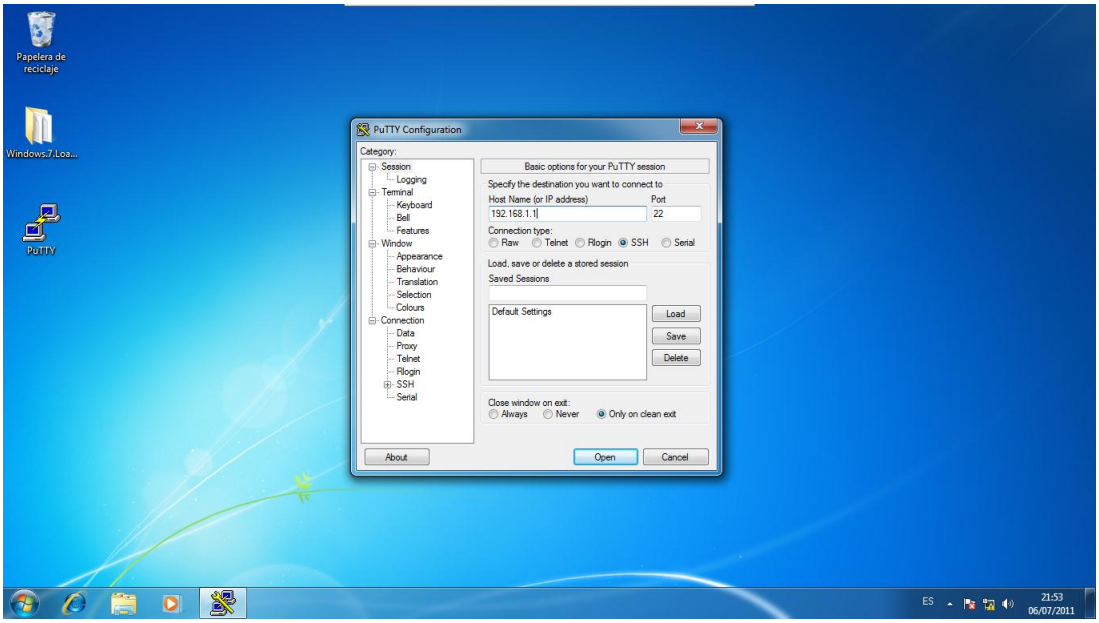

Figura 3.2.3.3 Configuración para la conexión al servidor SSH.

Damos clic en Open y se nos va a abrir un editor de comandos en el cual vamos a ingresar el nombre de usuario y la contraseña del servidor.

Y con esto ya estamos conectados al servidor, como se ilustra en la gráfica:

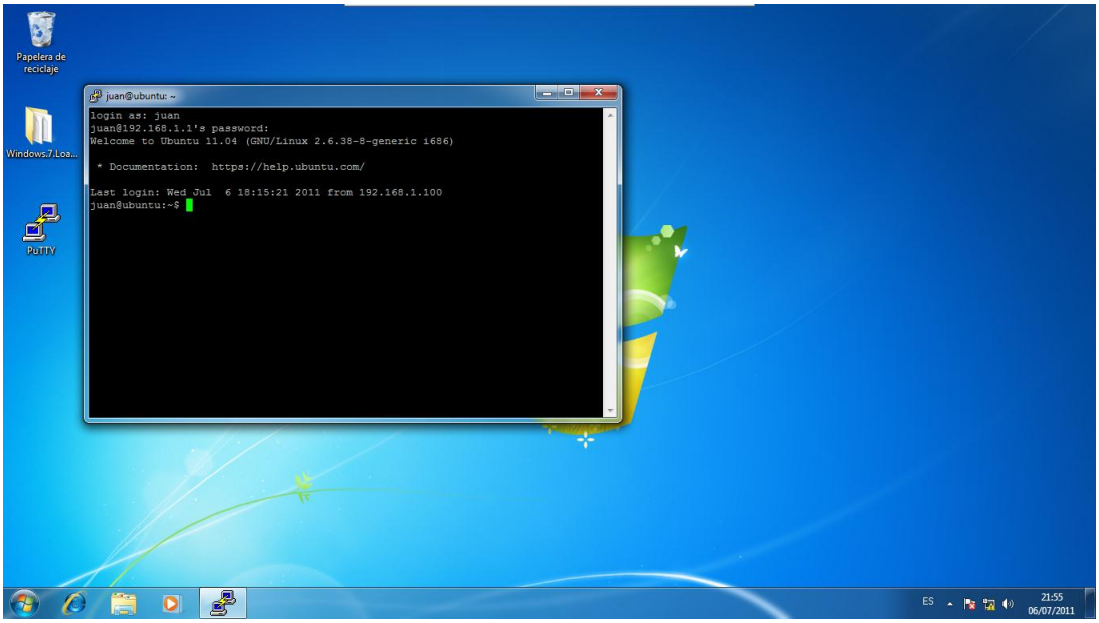

Figura 3.2.3.4 Conexión SSH desde Windows 7 al servidor.

3.3 IMPLEMENTACIÓN DE KIPPO EN EL SERVIDOR.

Con estas configuraciones anteriores ya tengo instalado y configurado un Servidor y también tengo instalado y configuradas máquinas que se conectan a través de SSH a este Servidor. Puedo instalar y configurar un sin fin numero de máquinas que todas van a tener acceso a este Servidor.

El problema está en que no puedo limitar a bloquear IPs debido a que necesitamos que la conectividad sea a través de cualquier IP; es decir sea desde cualquier parte del mundo, independientemente de donde nos podamos encontrar.

Cada día la tecnología se desarrolla y con esto también los intrusos adquieren mejores técnicas para espiar, sabotear o robar información, por lo tanto la implementación de KIPPO se basa en detectar estas acciones de los atacantes cuando han ganado acceso mediante SSH para mejorar la seguridad de la red de un Servidor.

Dependiendo cual sea el fin se puede utilizar KIPPO, pero el objetivo de utilizar KIPPO es estudiar los nuevos método con los cuales operan los intrusos para mejorar la seguridad de una red para evitar que se puedan seguir dando estos ataques.

Vamos a configurar el Honeypot KIPPO para emular un servidor SSH. Como primer paso, en el servidor vamos a instalar python-twisted:

$ sudo apt-get install python-twisted

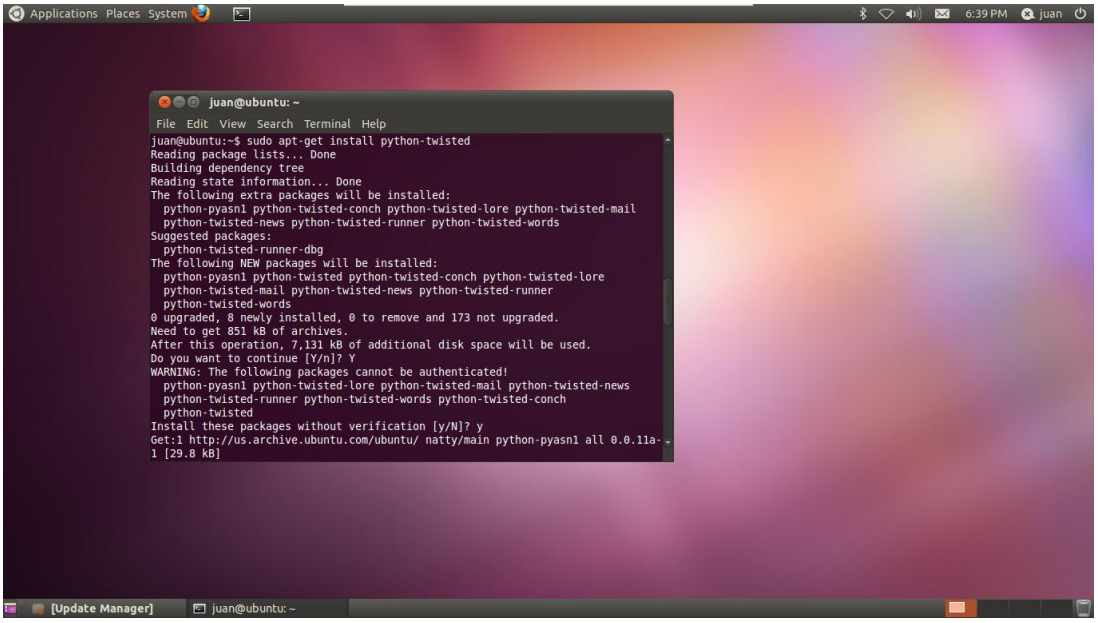

Figura 3.3.1 Instalación de python-twisted.

Después de que termine la instalación procedemos a descargarnos KIPPO:

$ wget http://kippo.googlecode.com/files/kippo-0.5.tar.gz

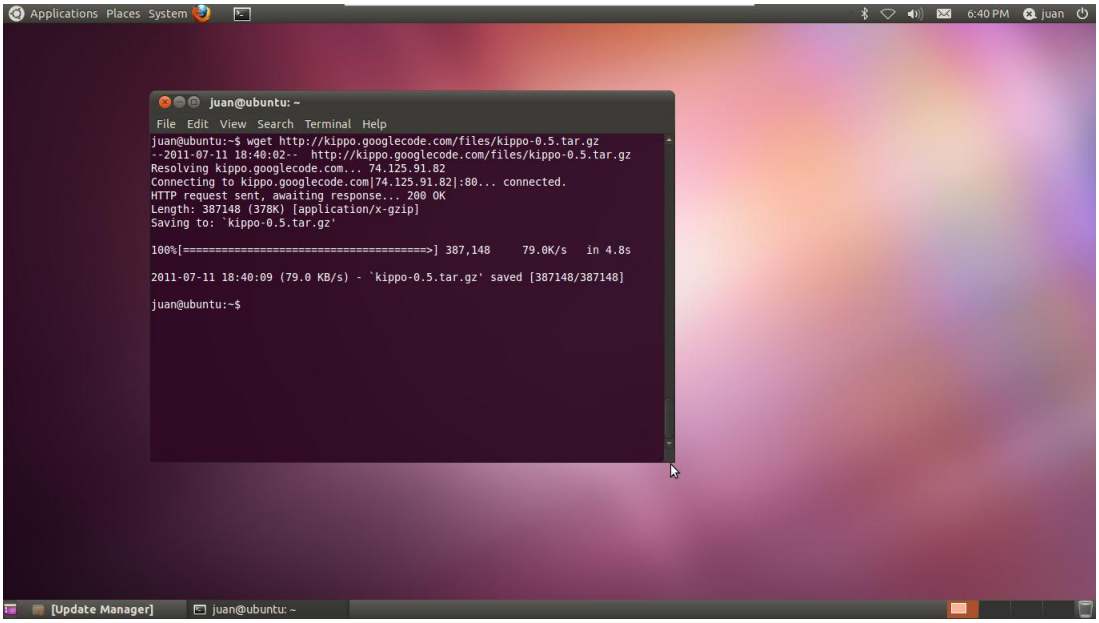

Figura 3.3.2 Descarga de KIPPO.

Luego de descargarnos descomprimimos el archivo para poder utilizarlo:

$ tar -xvzf kippo-0.5.tar.gz

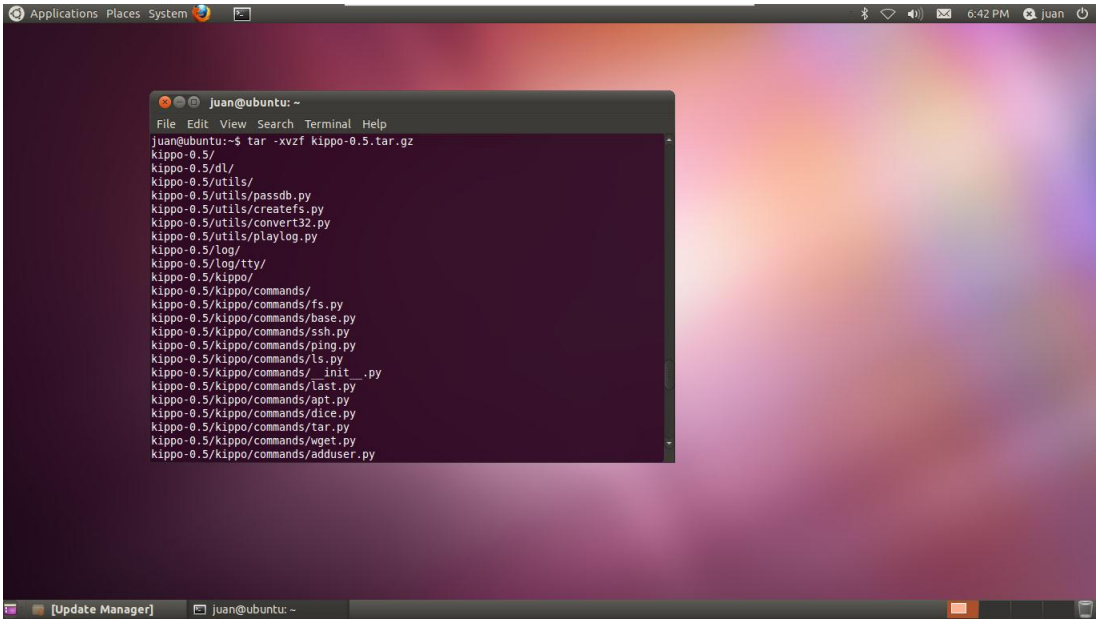

Figura 3.3.3 Descomprimir el programa KIPPO.

Una vez que se termina de descomprimir, obtenemos todo el software listo para ser usado, dentro de este tenemos los archivos de configuración para el acceso, los registros, los

comandos a ser utilizados en la emulación de una máquina, los directorios y carpetas con las cuales vamos a engañar al intruso de modo que le sea interesante atacar, y en fin podemos crear más directorios depende de lo que queramos que sea atacado.

Para arrancar KIPPO ingresamos al directorio:

$ cd kippo-0.5/

Y ejecutamos:

./start.sh

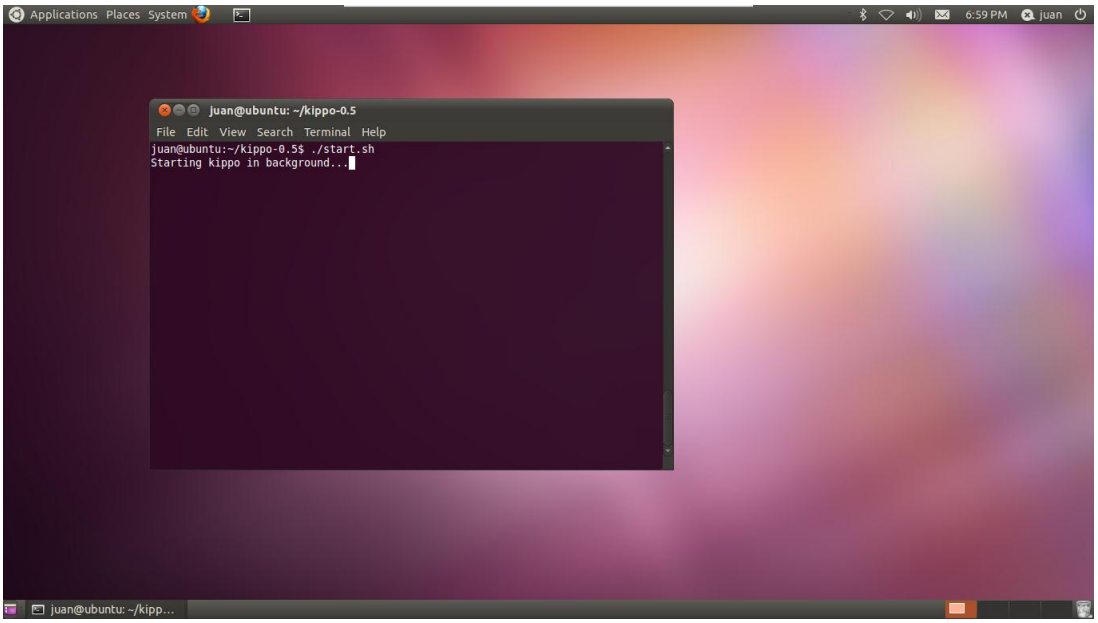

Figura 3.3.4 Como ejecutar KIPPO.

Con lo que ya tenemos configurado KIPPO en el servidor y listo para ser atacado.

El registro de salida por defecto de KIPPO será redirigido al archivo log/kippo.log. Para ver el archivo de registro de datos de KIPPO podemos utilizar el siguiente comando sobre una ventana terminal:

$ tail –f log/kippo.log

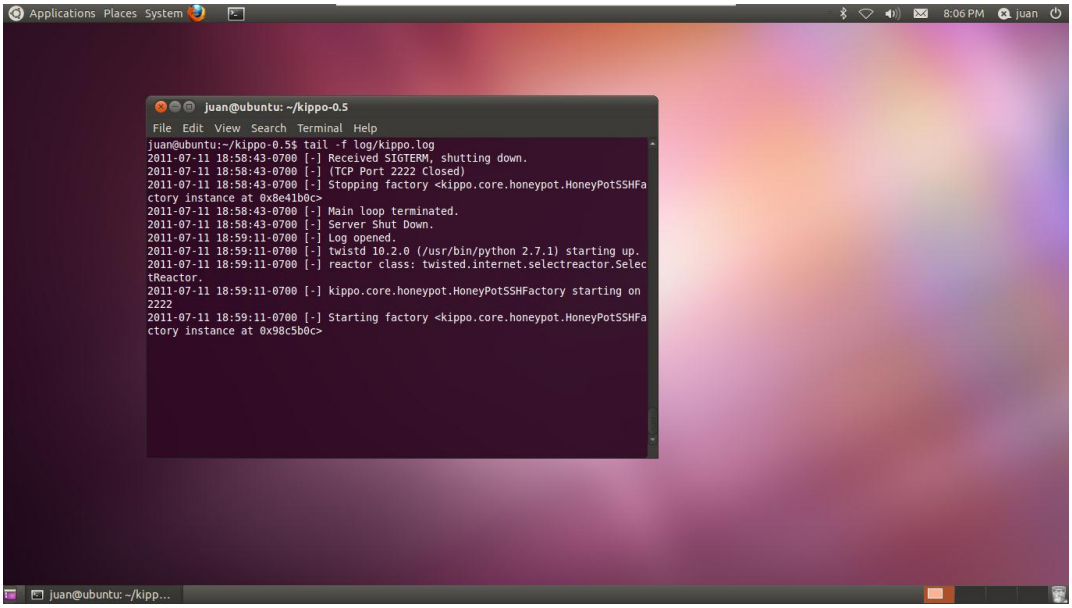

Figura 3.3.5 Registro de KIPPO.

Por defecto KIPPO se ejecuta en el Puerto 2222. Si este se ejecuta en Windows, generalmente utiliza el puerto 22. En Linux, el puerto 22 está restringido para utilizar como root, excepto si aplicamos los siguientes cambios:

$ sudo iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 22 -j REDIRECT --to-port 2222

Para realizar las pruebas de funcionamiento nos conectamos en una ventana de terminal del servidor con el servidor KIPPO en el puerto 2222, usando como nombre de usuario root, y como contraseña 123456.

$ ssh 127.0.0.1 -p 2222 -l root

Y podemos observar cómo me conecta a un supuesto servidor:

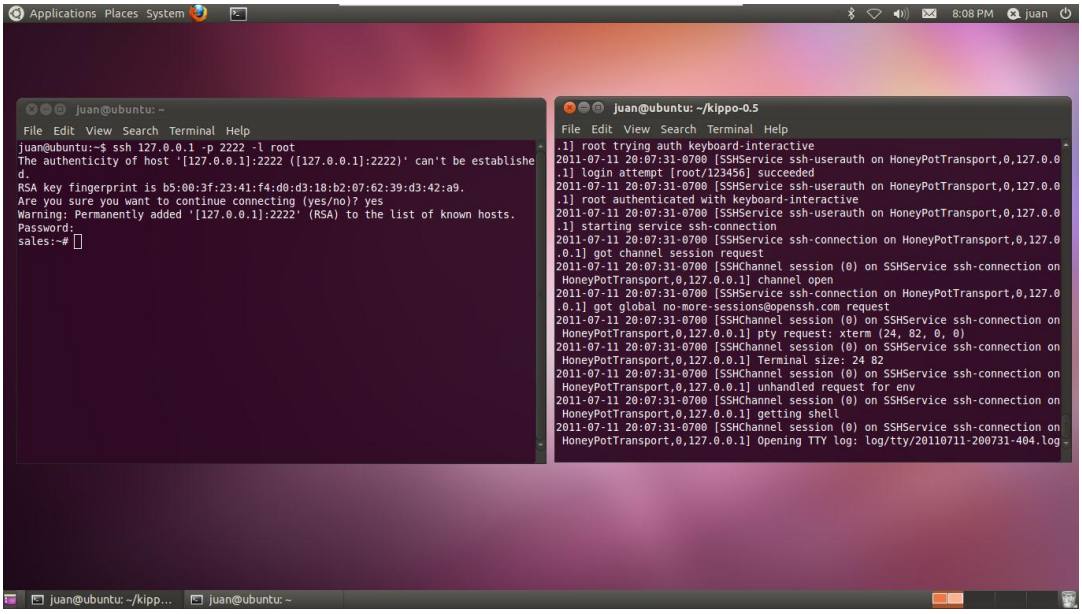

Figura 3.3.6 Conexión al servidor emulado por KIPPO.

Con lo cual una vez que el intruso haya ganado acceso por SSH va a comenzar a utilizar todas las técnicas que conoce para hacer suyo el servidor, y es en donde el intruso a caído en nuestra trampa, en el Honeypot que hemos creado a través de KIPPO; ya que vamos a poder observar todos sus movimientos y técnicas a utilizar con el fin de estudiar estos ataques, para comenzar a proteger y mejorar la seguridad de nuestras redes para que nuestros servidores reales que exponen valiosa información al mundo del internet no sigan cayendo en manos de intrusos.

CAPÍTULO IV

ANÁLISIS Y RESULTADOS DE LOS ATAQUES

4.1 PRUEBAS DE ATAQUES.

Para realizar las pruebas de ataque se va a hacer un ataque de fuerza bruta para ingresar al servidor mediante un ataque de diccionario. Otra forma es hacer un ataque manual, usando nombre de usuario y contraseña que con frecuencia se utilizan, y otra es utilizando la ingeniería social para obtener estos datos e ingresar al servidor.

De una u otra manera se va a identificar el tipo de ataque que está utilizando el intruso, las técnicas que utiliza y los datos que ingresa o lo que busca.

4.1.1 ATAQUE UTILIZANDO SISTEMA OPERATIVO CENTOS.

El Sistema Operativo de CentOS es el sistema que se ha escogido para instalar y configurar la máquina virtual atacante por medio de fuerza bruta a través de ataque por diccionario. Este ataque es uno de los más comunes que existen en el mundo, ya que es un ataque por programación que realiza la máquina para obtener el nombre de usuario y la contraseña para ingresar a través de un determinado puerto en un servidor.

Para comenzar con un ataque a un servidor de la red abrimos un terminal y ejecutamos el programa de ataque, poniendo la dirección del servidor, el lugar en donde se encuentran el diccionario con los nombres de usuario, el lugar en donde se encuentran el diccionario con las contraseñas, y también indicamos que vamos a atacar por el puerto SSH.

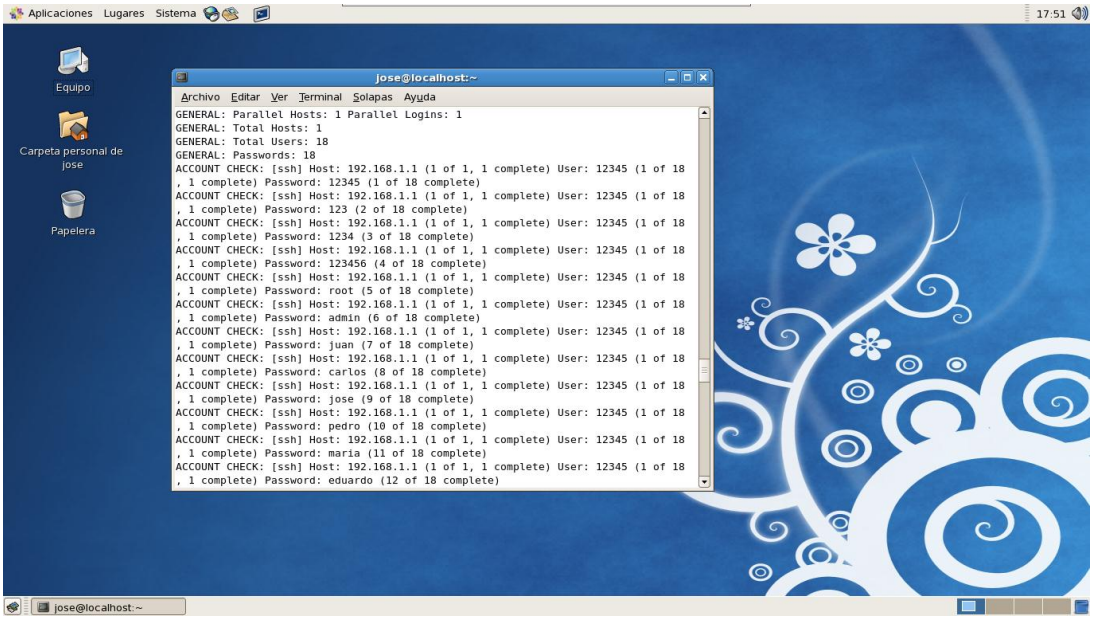

Figura 4.1.1.1 Ataque de fuerza bruta por diccionario

Ahora solo se espera a que me consiga el nombre de usuario y la contraseña, una vez encontrado el programa deja de ejecutarse y me muestra los datos.

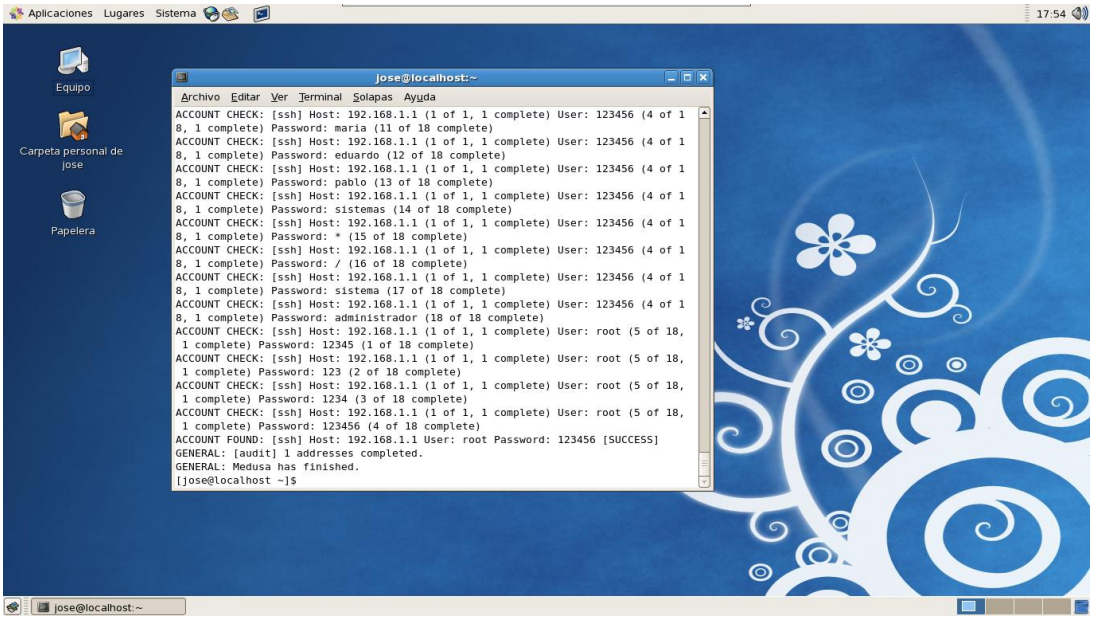

Figura 4.1.1.2 Ataque de fuerza bruta exitoso

Con el nombre de usuario y contraseña encontrados puedo ingresar al servidor como en el ejemplo:

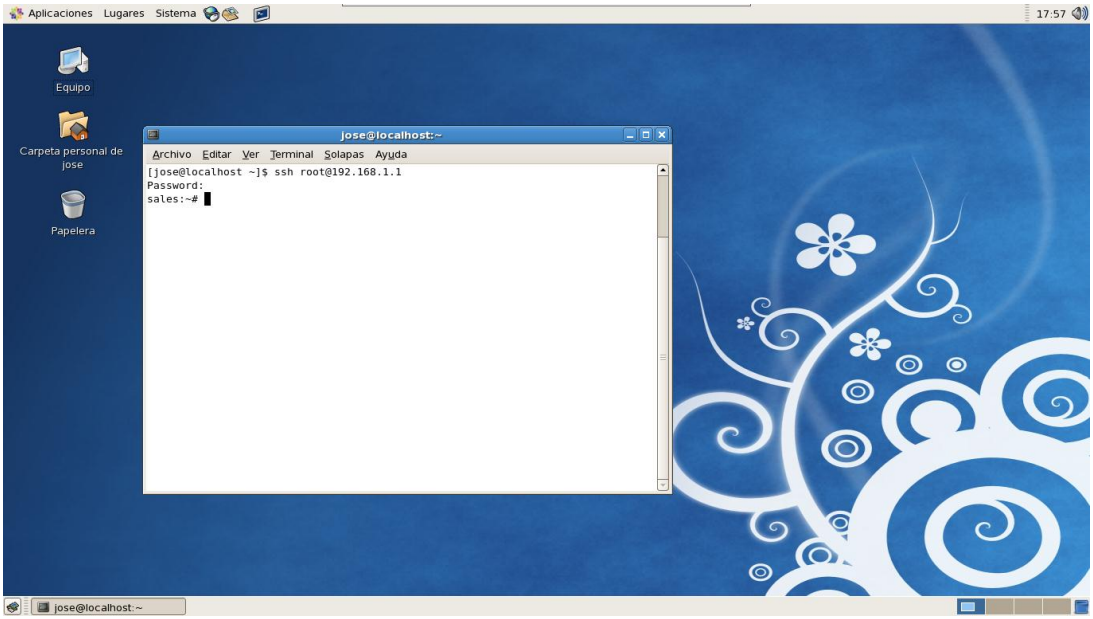

Figura 4.1.1.3 Ingreso por usuario y contraseña encontrados en CentOS

Con lo que como atacante supuestamente ya se puede ingresar en el servidor y hacer todo lo que se quiera.

4.1.2 ATAQUE UTILIZANDO SISTEMA OPERATIVO FEDORA.

El Sistema Operativo de Fedora es el sistema que se ha elegido para instalar y configurar la máquina virtual atacante por medio de ataque por nombres de usuario y contraseña que generalmente se utilizan. Este ataque es muy común y generalmente siempre se utiliza, ya que el atacante intenta vulnerar el sistema con nombres de usuario y contraseñas aleatorios, es la forma más rápida para ingresar a través de un puerto en un servidor antes de generar cualquier otro tipo de ataque.

Para comenzar con un ataque a un servidor de la red abrimos un terminal nos conectamos a través de SSH con cierto nombre de usuario y la dirección del servidor. Por ejemplo:

$ ssh jose@192.168.1.1

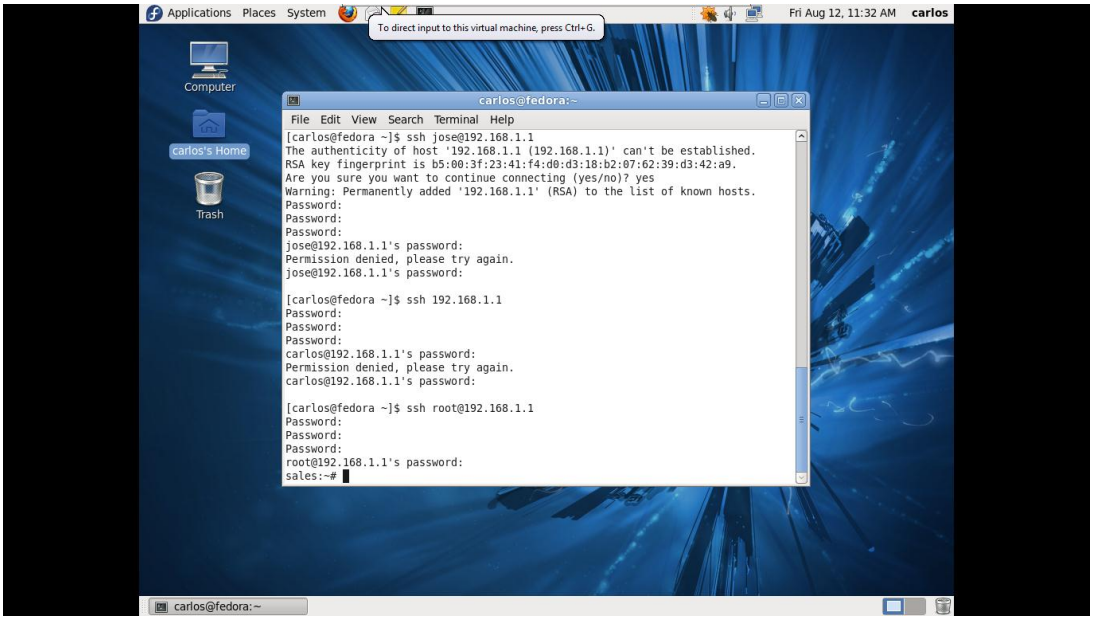

Figura 4.1.2.1 Ataque utilizando palabras comunes

Como podemos visualizar en la figura, comenzamos atacando con un nombre de usuario llamado “jose”, luego ataco con otro a través de mi mismo nombre de usuario, luego selecciono un usuario llamado “root” y en las contraseñas intento una que sea válida y general, y si se logra vulnerar el sistema de esta forma rápida, ya estoy dentro del supuesto servidor.

4.1.3 ATAQUE UTILIZANDO SISTEMA OPERATIVO WINDOWS

El Sistema Operativo de Windows 7 es el sistema que se ha elegido para instalar y configurar la máquina virtual atacante por medio de ataque por ingeniería social. En este tipo el ataque el atacante hace lo posible por conseguir el nombre de usuario y contraseña para ingresar de manera directa al servidor; este tipo de ataques es muy poco probable que ocurran debido a que generalmente toda empresa tiene la política de no revelar los datos y mucho menos relacionadas con algún servidor. Pero han existido casos que se hacen pasar por otras personas, o a través de correos de publicidad, malwares, etc.

Para comenzar con un ataque a un servidor de la red después de haber conseguido el nombre de usuario y contraseña nos podemos conectar al servidor SSH a través de un programa libre que se llama PuTTY.

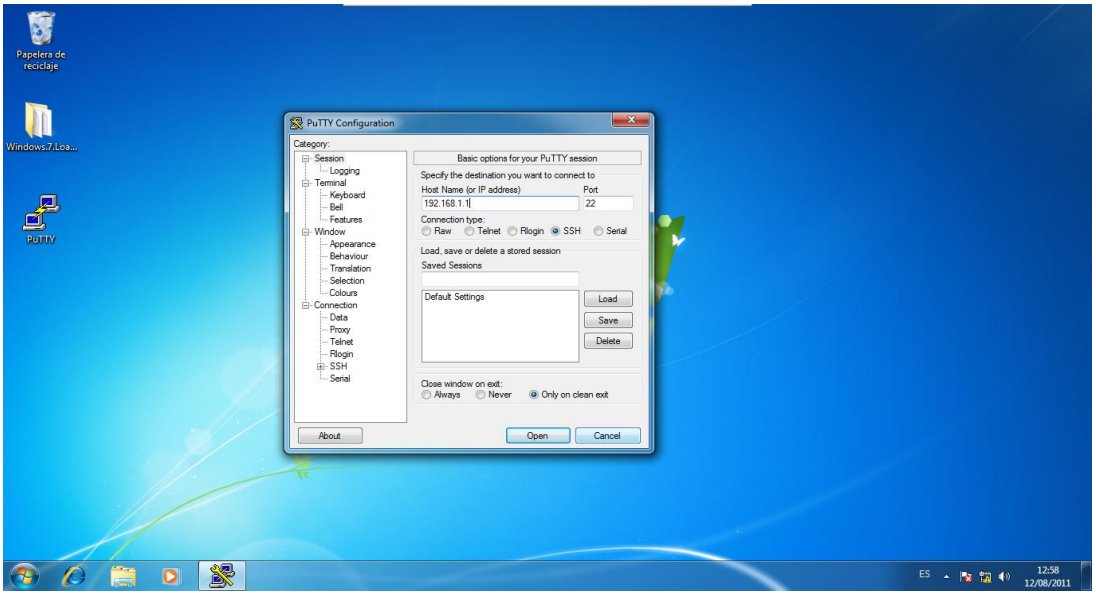

Figura 4.1.3.1 Ataque utilizando Ingeniería Social

Ingresamos la dirección del servidor y hacemos un clic en Open para abrir el puerto a través de SSH; después de esto me pedirá el nombre de usuario y la contraseña, después de ingresar estos datos ya estoy conectado al servidor.

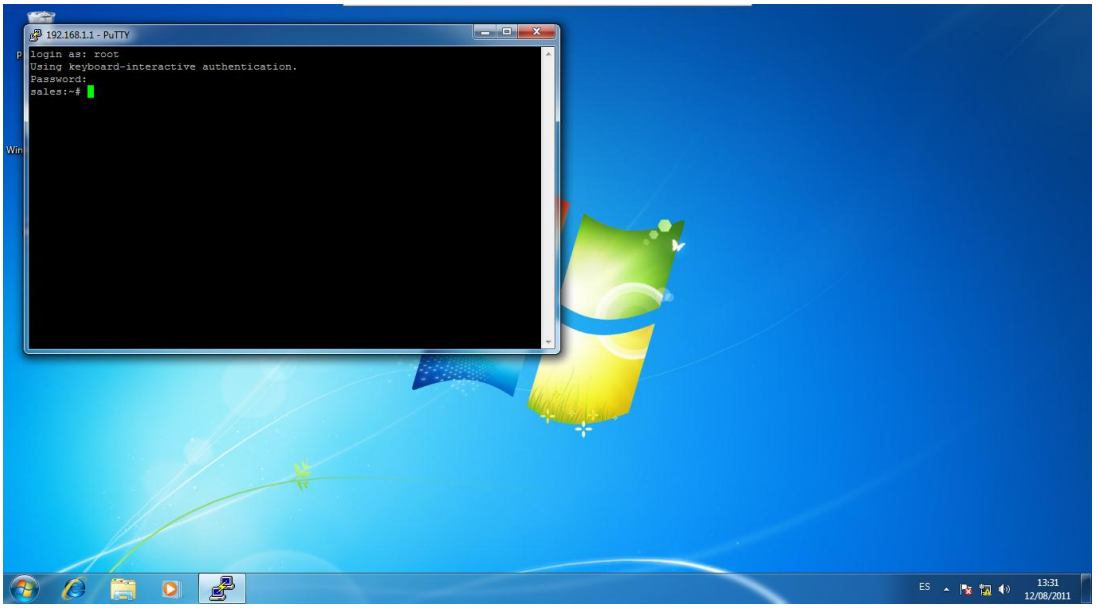

Figura 4.1.3.2 Ingreso por usuario y contraseña mediante Windows 7

4.1.4 PRUEBAS DE ATAQUE AL SERVIDOR SSH PARA LA VERIFICACIÓN Y ANÁLISIS DEL REGISTRO.

Al conectarme a través de SSH como atacante se va a espiar información de la máquina.

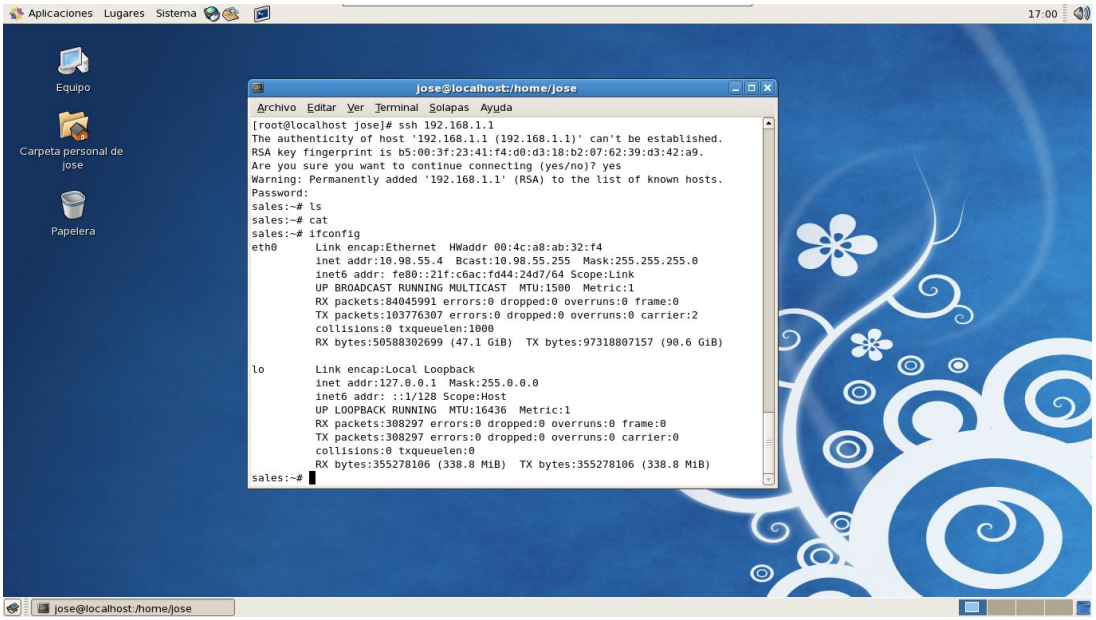

Figura 4.1.4.1 Ataque para espiar información del servidor

Se puede mirar archivos y carpetas.

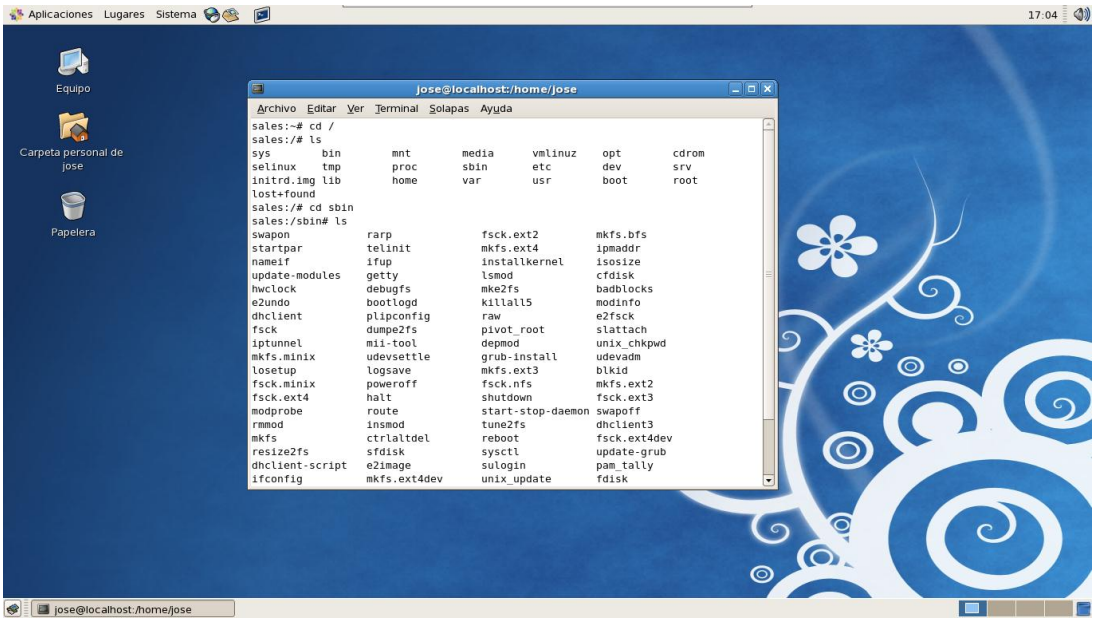

Figura 4.1.4.2 Ataque para explorar carpetas en el servidor

Generar llaves públicas, crear carpetas, copiar y borrar archivos, agregar usuarios.

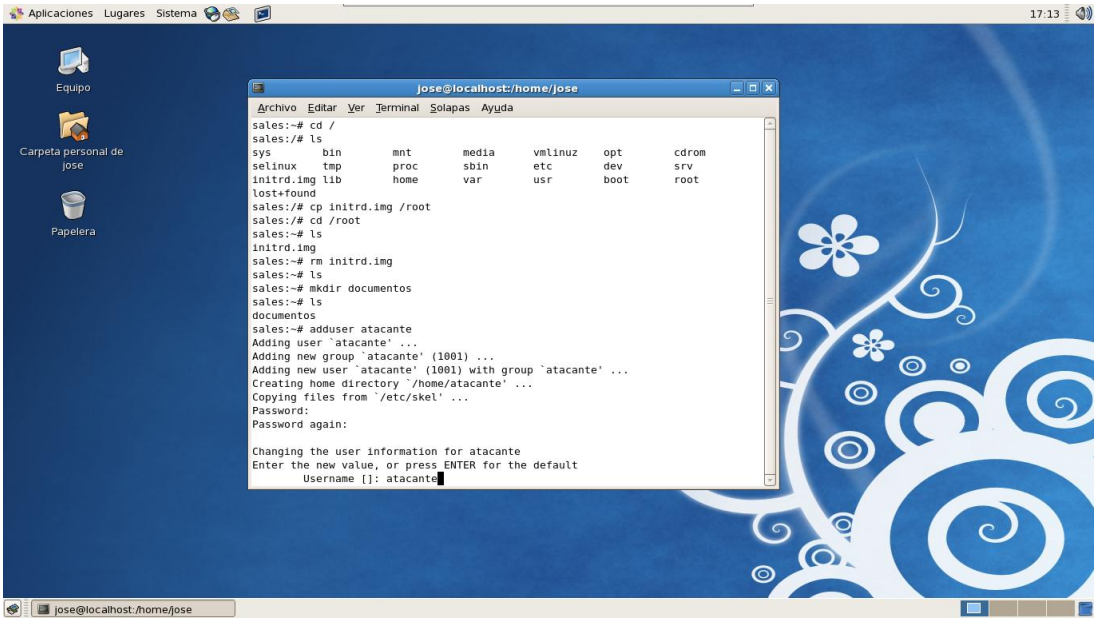

Figura 4.1.4.3 Ataque para copiar y borrar archivos o agregar usuarios en el servidor

En fin puedo hacer lo que me permitiría hacer un sistema operativo real.

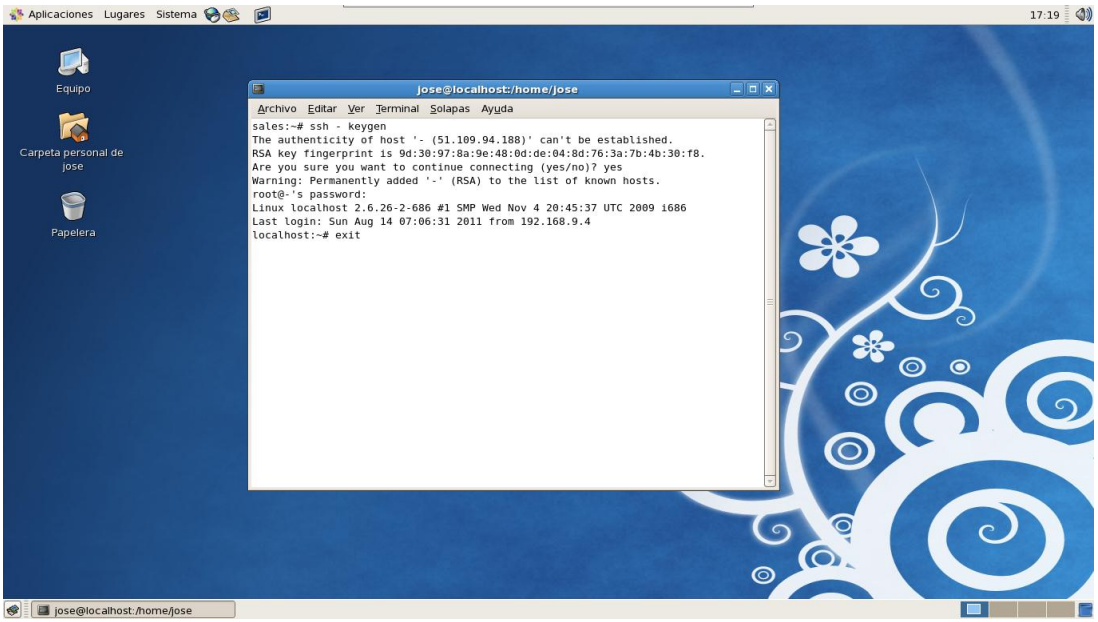

Figura 4.1.4.4 Ataque para crear llaves públicas en el servidor

Uno de las trampas que incluyen KIPPO es también que al desconectarse de la máquina atacada, KIPPO simula una desconexión para seguir trabajando en la máquina virtual propia del atacante, con lo que sigue obteniendo datos ingresados.

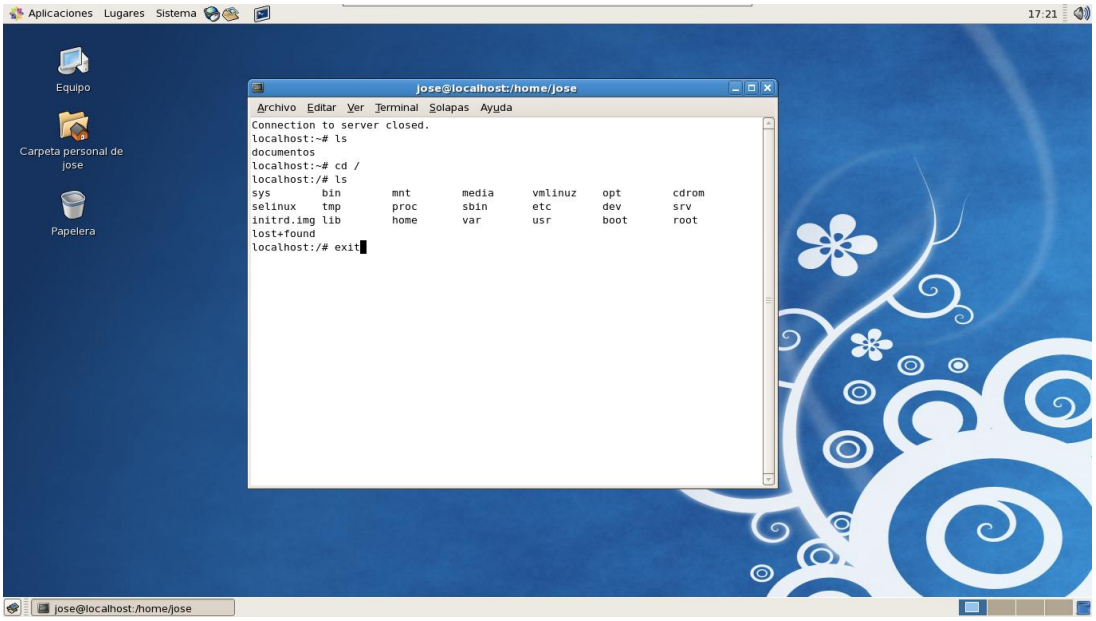

Figura 4.1.4.5 Desconexión por comandos del servidor

4.2 VERIFICACIÓN Y ANÁLISIS DE LOS REGISTROS.

Todos los registros se me almacenan en un archivo en la carpeta “Logs”, denominado “kippo.log”; esta almacena el día, la hora y todo lo relacionado con los datos del atacante, así como también las conexiones que utiliza y a través de que puertos.

En el registro siguiente me muestra un ataque de fuerza bruta realizado por la máquina atacante CentOS y almacenado en los registros de KIPPO:

2011-08-02 17:48:30-0700 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 192.168.1.108:50121 (192.168.1.1:2222) [session: 31]
2011-08-02 17:48:30-0700 [HoneyPotTransport,31,192.168.1.108] Remote SSH version: SSH-2.0-MEDUSA_1.0
2011-08-02 17:48:30-0700 [HoneyPotTransport,31,192.168.1.108] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2011-08-02 17:48:30-0700 [HoneyPotTransport,31,192.168.1.108] outgoing: aes128-ctr hmac-sha1 none
2011-08-02 17:48:30-0700 [HoneyPotTransport,31,192.168.1.108] incoming: aes128-ctr hmac-sha1 none
2011-08-02 17:48:30-0700 [HoneyPotTransport,31,192.168.1.108] NEW KEYS
2011-08-02 17:48:30-0700 [HoneyPotTransport,31,192.168.1.108] starting service ssh-userauth
2011-08-02 17:48:30-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth none
2011-08-02 17:48:30-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth password
2011-08-02 17:48:30-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] login attempt [12345/12345] failed
2011-08-02 17:48:31-0700 [-] 12345 failed auth password
2011-08-02 17:48:31-0700 [-] unauthorized login:
2011-08-02 17:48:31-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth none
2011-08-02 17:48:31-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth password
2011-08-02 17:48:31-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] login attempt [12345/123] failed
2011-08-02 17:48:32-0700 [-] 12345 failed auth password
2011-08-02 17:48:32-0700 [-] unauthorized login:
2011-08-02 17:48:32-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth none
2011-08-02 17:48:32-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth password
2011-08-02 17:48:32-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] login attempt [12345/1234] failed
2011-08-02 17:48:33-0700 [-] 12345 failed auth password
2011-08-02 17:48:33-0700 [-] unauthorized login:
2011-08-02 17:48:33-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth none
2011-08-02 17:48:33-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth password
2011-08-02 17:48:33-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] login attempt [12345/123456] failed

2011-08-02 17:48:34-0700 [-] 12345 failed auth password
2011-08-02 17:48:34-0700 [-] unauthorized login:
2011-08-02 17:48:34-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth none
2011-08-02 17:48:34-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] 12345 trying auth password
2011-08-02 17:48:34-0700 [SSHService ssh-userauth on HoneyPotTransport,31,192.168.1.108] login attempt [12345/root] failed
2011-08-02 17:48:35-0700 [-] 12345 failed auth password
2011-08-02 17:48:35-0700 [-] unauthorized login:

	Dirección y Puerto del atacante
	Dirección y Puerto del servidor
	Numero de Conexión intentadas por el atacante
	Software que está siendo utilizado por el atacante
	Nombre de usuario en el servidor
	Contraseña del usuario en el servidor
	Estado de la conexión

Tabla 4.2.1 Análisis de los ataques de fuerza bruta por diccionario

Los nombres de usuario y contraseñas son generadas a través de un ataque de fuerza bruta por diccionario, los ataques se me generan hasta encontrar en nombre de usuario y contraseña válido, como me muestra en el siguiente ejemplo:

2011-08-02 17:49:44-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth none
2011-08-02 17:49:44-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth password
2011-08-02 17:49:44-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] login attempt [root/12345] failed
2011-08-02 17:49:45-0700 [-] root failed auth password
2011-08-02 17:49:45-0700 [-] unauthorized login:
2011-08-02 17:49:45-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth none
2011-08-02 17:49:45-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth password
2011-08-02 17:49:45-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] login attempt [root/123] failed
2011-08-02 17:49:46-0700 [-] root failed auth password
2011-08-02 17:49:46-0700 [-] unauthorized login:
2011-08-02 17:49:46-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth none
2011-08-02 17:49:46-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth password
2011-08-02 17:49:46-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] login attempt [root/1234] failed
2011-08-02 17:49:47-0700 [-] root failed auth password
2011-08-02 17:49:47-0700 [-] unauthorized login:
2011-08-02 17:49:47-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth none
2011-08-02 17:49:47-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root trying auth password
2011-08-02 17:49:47-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] login attempt [root/123456] succeeded
2011-08-02 17:49:47-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] root authenticated with password
2011-08-02 17:49:47-0700 [SSHService ssh-userauth on HoneyPotTransport,35,192.168.1.108] starting service ssh-connection

	Nombre de usuario
	Contraseña del usuario
	Estado de la conexión

Tabla 4.2.2 Análisis del ataque de fuerza bruta exitoso

Cuando me encuentra en nombre de usuario y contraseña, me detiene de generar contraseñas. Después del ataque de fuerza bruta obtengo el registro de ingreso por parte del atacante como en el ejemplo:

2011-08-02 17:57:28-0700 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 192.168.1.108:33082 (192.168.1.1:2222) [session: 38]
2011-08-02 17:57:28-0700 [HoneyPotTransport,38,192.168.1.108] Remote SSH version: SSH-2.0-OpenSSH_4.3
2011-08-02 17:57:28-0700 [HoneyPotTransport,38,192.168.1.108] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2011-08-02 17:57:28-0700 [HoneyPotTransport,38,192.168.1.108] outgoing: aes128-cbc hmac-md5 none
2011-08-02 17:57:28-0700 [HoneyPotTransport,38,192.168.1.108] incoming: aes128-cbc hmac-md5 none
2011-08-02 17:57:28-0700 [HoneyPotTransport,38,192.168.1.108] NEW KEYS
2011-08-02 17:57:28-0700 [HoneyPotTransport,38,192.168.1.108] starting service ssh-userauth
2011-08-02 17:57:28-0700 [SSHService ssh-userauth on HoneyPotTransport,38,192.168.1.108] root trying auth none
2011-08-02 17:57:28-0700 [SSHService ssh-userauth on HoneyPotTransport,38,192.168.1.108] root trying auth keyboard-interactive
2011-08-02 17:57:33-0700 [SSHService ssh-userauth on HoneyPotTransport,38,192.168.1.108] login attempt [root/123456] succeeded
2011-08-02 17:57:33-0700 [SSHService ssh-userauth on HoneyPotTransport,38,192.168.1.108] root authenticated with keyboard-interactive
2011-08-02 17:57:33-0700 [SSHService ssh-userauth on HoneyPotTransport,38,192.168.1.108] starting service ssh-connection
2011-08-02 17:57:33-0700 [SSHService ssh-connection on HoneyPotTransport,38,192.168.1.108] got channel session request
2011-08-02 17:57:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,38,192.168.1.108] channel open
2011-08-02 17:57:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,38,192.168.1.108] pty request: xterm (24, 80, 0, 0)
2011-08-02 17:57:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,38,192.168.1.108] Terminal size: 24 80
2011-08-02 17:57:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,38,192.168.1.108] unhandled request for env
2011-08-02 17:57:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,38,192.168.1.108] getting shell
2011-08-02 17:57:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,38,192.168.1.108] Opening TTY log: log/tty/20110802-175733-3384.log

	Dirección y Puerto del atacante
	Dirección y Puerto del servidor
	Numero de Conexión intentadas por el atacante
	Software que está siendo utilizado por el atacante
	Nombre de usuario en el servidor
	Contraseña del usuario en el servidor
	Estado de la conexión
	Dirección del registro de los movimientos

Tabla 4.2.3 Análisis de conexión mediante CentOS al servidor

El siguiente registro es del ataque por nombres de usuarios y contraseñas comunes que fue utilizando un terminal en sistema operativo Fedora:

2011-08-12 09:22:09-0700 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 192.168.1.109:47972 (192.168.1.1:2222) [session: 2]
2011-08-12 09:22:09-0700 [HoneyPotTransport,2,192.168.1.109] Remote SSH version: SSH-2.0-OpenSSH_5.5
2011-08-12 09:22:09-0700 [HoneyPotTransport,2,192.168.1.109] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2011-08-12 09:22:09-0700 [HoneyPotTransport,2,192.168.1.109] outgoing: aes128-ctr hmac-md5 none
2011-08-12 09:22:09-0700 [HoneyPotTransport,2,192.168.1.109] incoming: aes128-ctr hmac-md5 none
2011-08-12 09:22:13-0700 [HoneyPotTransport,2,192.168.1.109] NEW KEYS
2011-08-12 09:22:13-0700 [HoneyPotTransport,2,192.168.1.109] starting service ssh-userauth
2011-08-12 09:22:13-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose trying auth none
2011-08-12 09:22:13-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose trying auth keyboard-interactive
2011-08-12 09:22:23-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] login attempt [jose/jose] failed
2011-08-12 09:22:23-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose failed auth keyboard-interactive
2011-08-12 09:22:23-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] unauthorized login:
2011-08-12 09:22:23-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose trying auth keyboard-interactive
2011-08-12 09:22:26-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] login attempt [jose/juan] failed
2011-08-12 09:22:26-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose failed auth keyboard-interactive
2011-08-12 09:22:26-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] unauthorized login:
2011-08-12 09:22:26-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose trying auth keyboard-interactive
2011-08-12 09:22:28-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] login attempt [jose/pedro] failed

2011-08-12 09:22:28-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose failed auth keyboard-interactive
2011-08-12 09:22:28-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] unauthorized login:
2011-08-12 09:22:39-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] jose trying auth password
2011-08-12 09:22:39-0700 [SSHService ssh-userauth on HoneyPotTransport,2,192.168.1.109] login attempt [jose/123456] failed
2011-08-12 09:22:40-0700 [-] jose failed auth password
2011-08-12 09:22:40-0700 [-] unauthorized login:
2011-08-12 09:22:44-0700 [HoneyPotTransport,2,192.168.1.109] connection lost
2011-08-12 09:22:54-0700 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 192.168.1.109:47973 (192.168.1.1:2222) [session: 3]
2011-08-12 09:22:54-0700 [HoneyPotTransport,3,192.168.1.109] Remote SSH version: SSH-2.0-OpenSSH_5.5
2011-08-12 09:22:54-0700 [HoneyPotTransport,3,192.168.1.109] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2011-08-12 09:22:54-0700 [HoneyPotTransport,3,192.168.1.109] outgoing: aes128-ctr hmac-md5 none
2011-08-12 09:22:54-0700 [HoneyPotTransport,3,192.168.1.109] incoming: aes128-ctr hmac-md5 none
2011-08-12 09:22:55-0700 [HoneyPotTransport,3,192.168.1.109] NEW KEYS
2011-08-12 09:22:55-0700 [HoneyPotTransport,3,192.168.1.109] starting service ssh-userauth
2011-08-12 09:22:55-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos trying auth none
2011-08-12 09:22:55-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos trying auth keyboard-interactive
2011-08-12 09:22:59-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] login attempt [carlos/123] failed
2011-08-12 09:22:59-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos failed auth keyboard-interactive
2011-08-12 09:22:59-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] unauthorized login:
2011-08-12 09:22:59-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos trying auth keyboard-interactive
2011-08-12 09:23:05-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] login attempt [carlos/pasword] failed
2011-08-12 09:23:05-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos failed auth keyboard-interactive
2011-08-12 09:23:05-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] unauthorized login:
2011-08-12 09:23:05-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos trying auth keyboard-interactive
2011-08-12 09:23:15-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] login attempt [carlos/admin] failed
2011-08-12 09:23:15-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos failed auth keyboard-interactive
2011-08-12 09:23:15-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] unauthorized login:
2011-08-12 09:23:19-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] carlos trying auth password
2011-08-12 09:23:20-0700 [SSHService ssh-userauth on HoneyPotTransport,3,192.168.1.109] login attempt [carlos/root] failed
2011-08-12 09:23:21-0700 [-] carlos failed auth password
2011-08-12 09:23:21-0700 [-] unauthorized login:
2011-08-12 09:23:28-0700 [HoneyPotTransport,3,192.168.1.109] connection lost
2011-08-12 09:23:40-0700 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 192.168.1.109:47974 (192.168.1.1:2222) [session: 4]
2011-08-12 09:23:40-0700 [HoneyPotTransport,4,192.168.1.109] Remote SSH version: SSH-2.0-OpenSSH_5.5
2011-08-12 09:23:40-0700 [HoneyPotTransport,4,192.168.1.109] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2011-08-12 09:23:40-0700 [HoneyPotTransport,4,192.168.1.109] outgoing: aes128-ctr hmac-md5 none
2011-08-12 09:23:40-0700 [HoneyPotTransport,4,192.168.1.109] incoming: aes128-ctr hmac-md5 none
2011-08-12 09:23:40-0700 [HoneyPotTransport,4,192.168.1.109] NEW KEYS
2011-08-12 09:23:40-0700 [HoneyPotTransport,4,192.168.1.109] starting service ssh-userauth
2011-08-12 09:23:40-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root trying auth none
2011-08-12 09:23:40-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root trying auth keyboard-interactive
2011-08-12 09:23:43-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] login attempt [root/root] failed
2011-08-12 09:23:43-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root failed auth keyboard-interactive
2011-08-12 09:23:43-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] unauthorized login:
2011-08-12 09:23:43-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root trying auth keyboard-interactive
2011-08-12 09:23:49-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] login attempt [root/password] failed
2011-08-12 09:23:49-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root failed auth keyboard-interactive
2011-08-12 09:23:49-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] unauthorized login:
2011-08-12 09:23:49-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root trying auth keyboard-interactive
2011-08-12 09:23:51-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] login attempt [root/12345] failed
2011-08-12 09:23:51-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root failed auth keyboard-interactive
2011-08-12 09:23:51-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] unauthorized login:
2011-08-12 09:23:54-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root trying auth password
2011-08-12 09:23:54-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] login attempt [root/123456] succeeded
2011-08-12 09:23:54-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] root authenticated with password
2011-08-12 09:23:54-0700 [SSHService ssh-userauth on HoneyPotTransport,4,192.168.1.109] starting service ssh-connection
2011-08-12 09:23:54-0700 [SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] got channel session request
2011-08-12 09:23:54-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] channel open
2011-08-12 09:23:54-0700 [SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] got global no-more-sessions@openssh.com request
2011-08-12 09:23:54-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] pty request: xterm (24, 80, 0, 0)
2011-08-12 09:23:54-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] Terminal size: 24 80

2011-08-12 09:23:54-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] unhandled request for env
2011-08-12 09:23:54-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] unhandled request for env
2011-08-12 09:23:54-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] getting shell
2011-08-12 09:23:54-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,4,192.168.1.109] Opening TTY log: log/tty/20110812-092354-3844.log

	Dirección y Puerto del atacante
	Dirección y Puerto del servidor
	Numero de Conexión intentadas por el atacante
	Software que está siendo utilizado por el atacante
	Nombre de usuario en el servidor
	Contraseña del usuario en el servidor
	Estado de la conexión
	Dirección del registro de los movimientos

Tabla 4.2.4 Análisis de los ataques utilizando palabras comunes

El siguiente registro es el de ingreso de un atacante por medio de PuTTY en Windows 7, este registro simula ser originado a partir de ingeniería social por lo que se obtuvo el nombre de usuario y la contraseña:

2011-08-12 10:31:06-0700 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 192.168.1.110:49162 (192.168.1.1:2222) [session: 5]
2011-08-12 10:31:07-0700 [HoneyPotTransport,5,192.168.1.110] Remote SSH version: SSH-2.0-PuTTY_Release_0.60
2011-08-12 10:31:07-0700 [HoneyPotTransport,5,192.168.1.110] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2011-08-12 10:31:07-0700 [HoneyPotTransport,5,192.168.1.110] outgoing: aes256-ctr hmac-sha1 none
2011-08-12 10:31:07-0700 [HoneyPotTransport,5,192.168.1.110] incoming: aes256-ctr hmac-sha1 none
2011-08-12 10:31:08-0700 [HoneyPotTransport,5,192.168.1.110] NEW KEYS
2011-08-12 10:31:08-0700 [HoneyPotTransport,5,192.168.1.110] starting service ssh-userauth
2011-08-12 10:31:16-0700 [SSHService ssh-userauth on HoneyPotTransport,5,192.168.1.110] root trying auth none
2011-08-12 10:31:16-0700 [SSHService ssh-userauth on HoneyPotTransport,5,192.168.1.110] root trying auth keyboard-interactive
2011-08-12 10:31:23-0700 [SSHService ssh-userauth on HoneyPotTransport,5,192.168.1.110] login attempt [root/123456] succeeded
2011-08-12 10:31:23-0700 [SSHService ssh-userauth on HoneyPotTransport,5,192.168.1.110] root authenticated with keyboard-interactive
2011-08-12 10:31:23-0700 [SSHService ssh-userauth on HoneyPotTransport,5,192.168.1.110] starting service ssh-connection
2011-08-12 10:31:23-0700 [SSHService ssh-connection on HoneyPotTransport,5,192.168.1.110] got channel session request
2011-08-12 10:31:23-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,5,192.168.1.110] channel open
2011-08-12 10:31:23-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,5,192.168.1.110] pty request: xterm (24, 80, 0, 0)
2011-08-12 10:31:23-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,5,192.168.1.110] Terminal size: 24 80
2011-08-12 10:31:23-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,5,192.168.1.110] getting shell
2011-08-12 10:31:24-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,5,192.168.1.110] Opening TTY log: log/tty/20110812-103124-2914.log

	Dirección y Puerto del atacante
	Dirección y Puerto del servidor
	Numero de Conexión intentadas por el atacante
	Software que está siendo utilizado por el atacante
	Nombre de usuario en el servidor
	Contraseña del usuario en el servidor
	Estado de la conexión
	Dirección del registro de los movimientos

Tabla 4.2.5 Análisis de los ataques utilizando Ingeniería Social.

El siguiente registro fue obtenido en el subcapítulo 4.1.4 (PRUEBAS DE ATAQUE AL SERVIDOR SSH PARA LA VERIFICACIÓN Y ANÁLISIS DEL REGISTRO), con el fin de verificar los datos obtenidos y analizar estos registros.

Registro del la Figura 4.1.4.1 Ataque para espiar información del servidor:

2011-08-15 16:59:15-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 16:59:15-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls
2011-08-15 16:59:18-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cat
2011-08-15 16:59:18-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cat
2011-08-15 16:59:29-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ifconfig
2011-08-15 16:59:29-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ifconfig
2011-08-15 16:59:29-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Reading txtcmd from "/home/juan/kippo-0.5/txtcmds/sbin/ifconfig"

	Comando ingresado
	Dirección del registro en nuestro ordenador

Tabla 4.2.6 Análisis de los comandos espiando información del servidor

Registro de la Figura 4.1.4.2 Ataque para explorar carpetas en el servidor:

2011-08-15 17:02:41-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: clear
2011-08-15 17:02:41-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: clear
2011-08-15 17:02:53-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cd /
2011-08-15 17:02:53-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cd /
2011-08-15 17:02:55-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:02:55-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls
2011-08-15 17:03:42-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cd sbin
2011-08-15 17:03:42-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cd sbin
2011-08-15 17:03:44-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:03:44-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls

	Comando ingresado

Tabla 4.2.7 Análisis de los comandos explorando carpetas del servidor

Registro de la Figura 4.1.4.3 Ataque para copiar y borras archivos o agregar usuarios en el servidor:

2011-08-15 17:10:16-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cp initrd.img /root
2011-08-15 17:10:16-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cp initrd.img /root

2011-08-15 17:10:24-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: clear
2011-08-15 17:10:24-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: clear
2011-08-15 17:10:31-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cd /
2011-08-15 17:10:31-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cd /
2011-08-15 17:10:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:10:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls
2011-08-15 17:10:38-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cp initrd.img /root
2011-08-15 17:10:38-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cp initrd.img /root
2011-08-15 17:10:47-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cd /root
2011-08-15 17:10:47-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cd /root
2011-08-15 17:10:49-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:10:49-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls
2011-08-15 17:11:02-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: rm initrd.img
2011-08-15 17:11:02-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: rm initrd.img
2011-08-15 17:11:04-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:11:04-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls
2011-08-15 17:11:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: mkdir documentos
2011-08-15 17:11:33-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: mkdir documentos
2011-08-15 17:11:34-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:11:34-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls
2011-08-15 17:12:25-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: adduser atacante
2011-08-15 17:12:25-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: adduser atacante

	Comando ingresado

Tabla 4.2.8 Análisis de los comandos copiando y borrando archivos y creando usuarios

Registro de la Figura 4.1.4.4 Ataque para crear llaves públicas en el servidor:

2011-08-15 17:18:25-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ssh - keygen
2011-08-15 17:18:25-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ssh - keygen
2011-08-15 17:18:27-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] INPUT (ssh): yes
2011-08-15 17:18:32-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] INPUT (ssh): 123456
2011-08-15 17:20:05-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: exit
2011-08-15 17:20:05-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: exit

	Comando ingresado
	Datos de entrada ingresados

Tabla 4.2.9 Análisis de los comandos creando llaves públicas

Registro de la Figura 4.1.4.5 Desconexión por comandos del servidor:

2011-08-15 17:21:37-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: exit
2011-08-15 17:21:37-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: exit
2011-08-15 17:21:39-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:21:39-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls
2011-08-15 17:21:41-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: cd /
2011-08-15 17:21:41-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: cd /
2011-08-15 17:21:45-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] CMD: ls
2011-08-15 17:21:45-0700 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,0,192.168.1.108] Command found: ls

	Comando ingresado

Tabla 4.2.10 Análisis de los comandos intentando salir de la conexión SSH

Otra gran herramienta y una de las mas importante que tenemos en KIPPO es el registro de la grabación y ejecución del terminal que fue usada por el atacante, este registro es muy importante ya que así también podemos descubrir las verdaderas intenciones del atacante, algún comando que lo borro o quiso esconder para que no se den en cuenta; en fin, el comando es siguiente:

$ python playlog.py -m (tiempo de demora en seg) (dirección) 0

Como por ejemplo:

$ python playlog.py -m 1 /home/juan/kippo-0.5/log/tty/20110815-165822-1773.log 0

Las siguientes 3 figuras muestran la réplica de la ejecución del terminal atacante captado y reproducido en el ordenador del servidor.

Esta figura reproduce el ataque de la Figura 4.1.4.1 Ataque para espiar información del servidor:

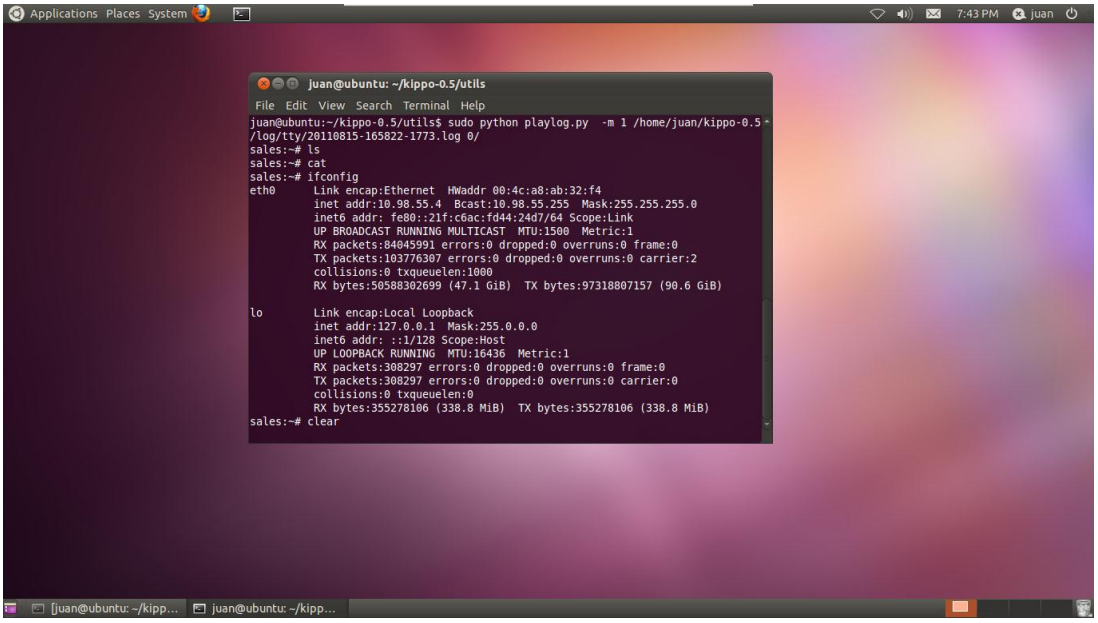

Figura 4.2.1 Reproducción del ataque de la información del servidor

Esta figura reproduce el ataque de la Figura 4.1.4.3 Ataque para copiar y borrar archivos, agregar carpetas y usuarios en el servidor

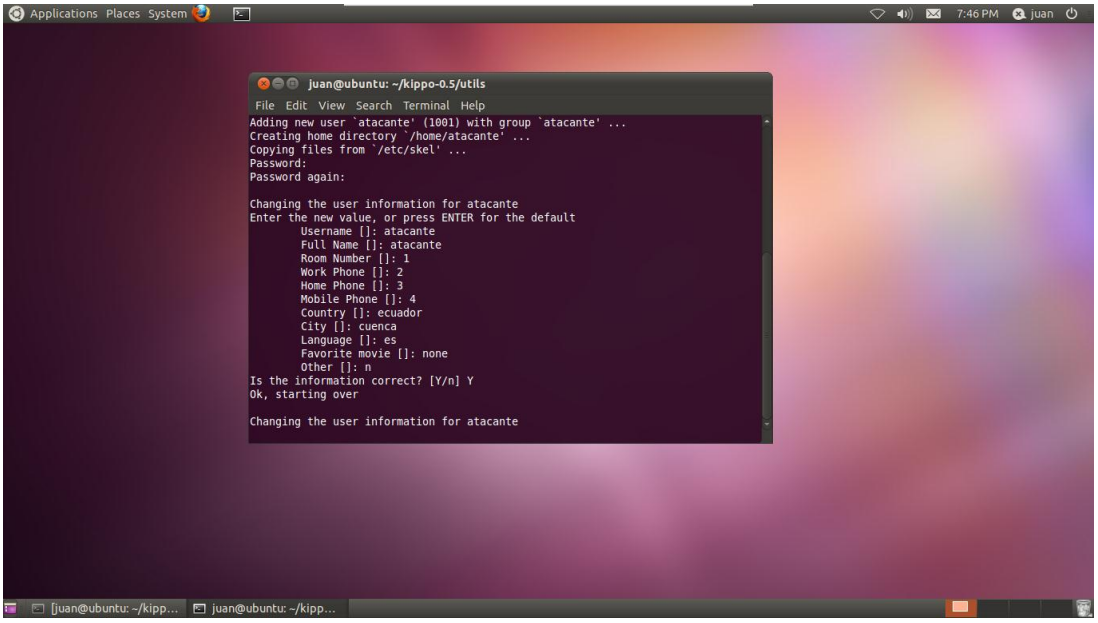

Figura 4.2.2 Reproducción del ataque creando un usuario

Esta figura reproduce el ataque de la Figura 4.1.4.4 Ataque para crear llaves públicas en el servidor:

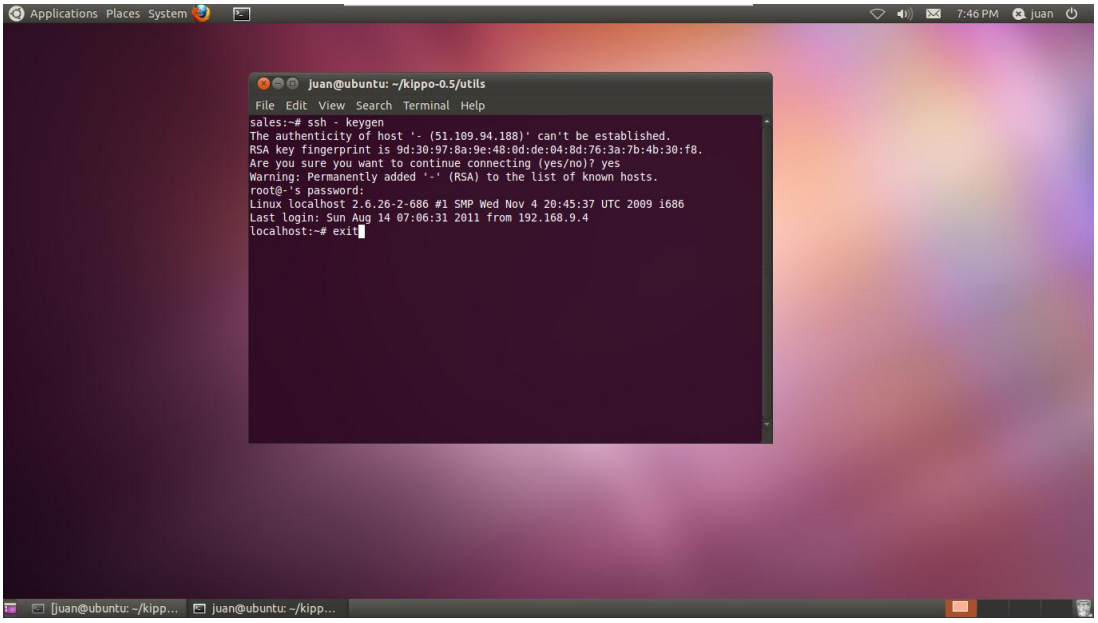

Figura 4.2.3 Reproducción del ataque creando una llave pública

4.3 ANEXOS.

4.3.1 ANEXO 1: Instalación y configuración del VMware

El programa VMware lo podemos descargar desde el sitio web oficial www.vmware.com, este programa es un simulador de máquinas virtuales para trabajar en diferentes plataformas y al mismo tiempo. Ejecutamos el programa de instalación de la máquina virtual VMware. Se me aparece la pantalla de inicio de instalación y vamos a continuar, después de eso seleccionamos en “typical” que es el modo de instalación por defecto.

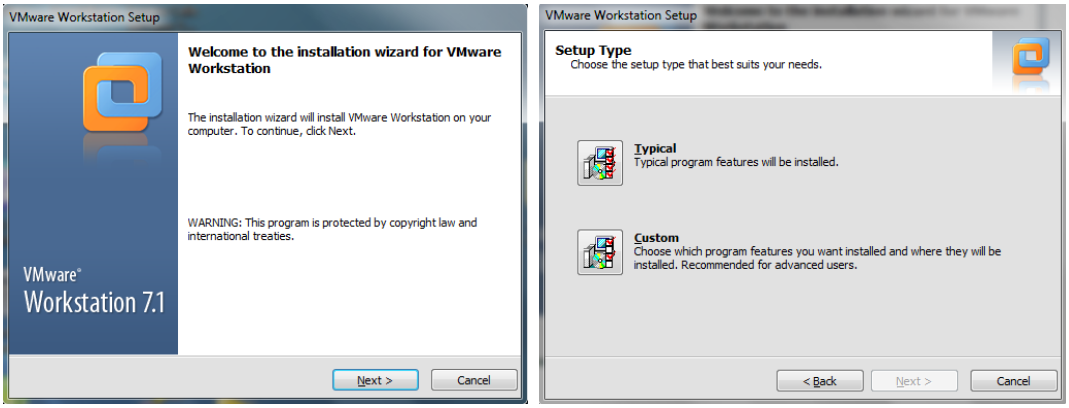

Figura 4.3.1.1 Inicio y Selección de modo de instalación de VMWare

Seleccionamos la carpeta de destino de instalación y continuamos con el proceso; la siguiente pantalla a esa nos pregunta si deseamos actualizaciones de software del programa cada vez que inicia; en nuestro caso deshabilitamos esa opción y continuamos.

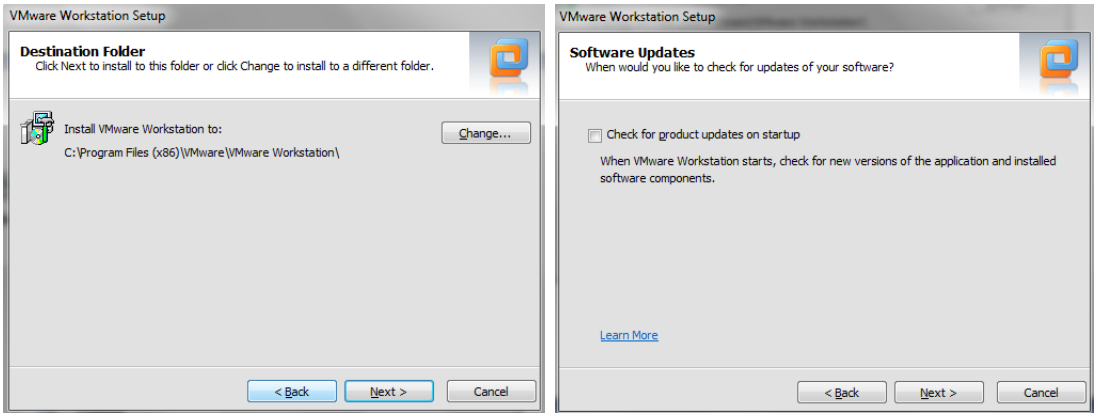

Figura 4.3.1.2 Directorio y Actualización de instalación del VMWare

Después me pregunta si deseo participar en las mejoras de VMWare; en nuestro caso deshabilitamos la opción y continuamos con la instalación, elegimos los accesos rápidos.

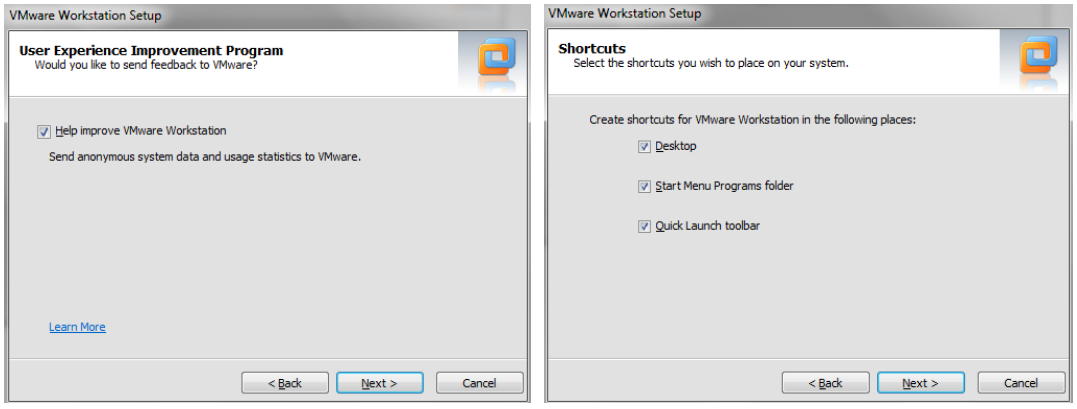

Figura 4.3.1.3 Participación de mejoras y accesos rápidos en la instalación de VMWare

La siguiente pantalla me indica que todo está listo para continuar con el proceso, si queremos cambiar algo podemos regresar o de lo contrario continuamos para iniciar la instalación.

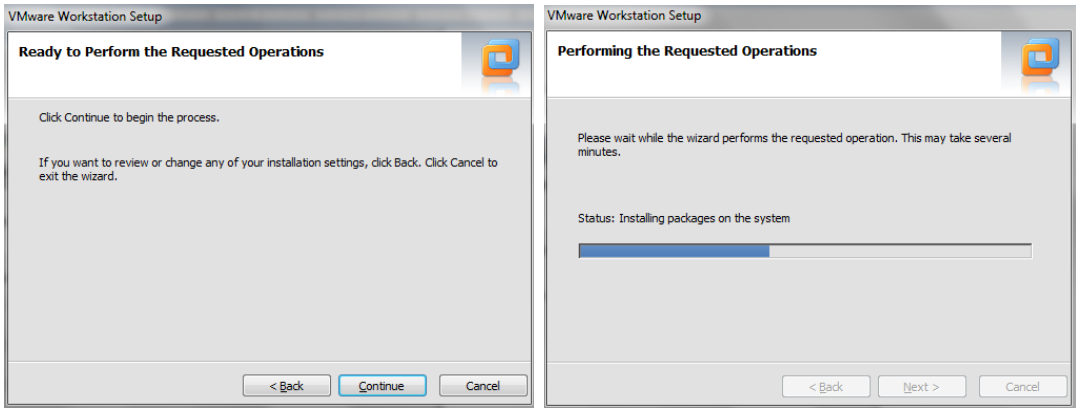

Figura 4.3.1.4 Inicio de instalación del VMWare

Después de terminada la instalación ingreso el número de la licencia reinicio el sistema para guardar los cambios.

Al abrir el VMware me pregunta si acepto los términos de la licencia con lo cual selecciono “Yes” para aceptarlos y a continuación en OK para validar la información. Siguiente a esto se abre la pantalla principal el programa VMware, con lo cual ya está listo para configurar y ejecutar las máquinas virtuales deseadas.

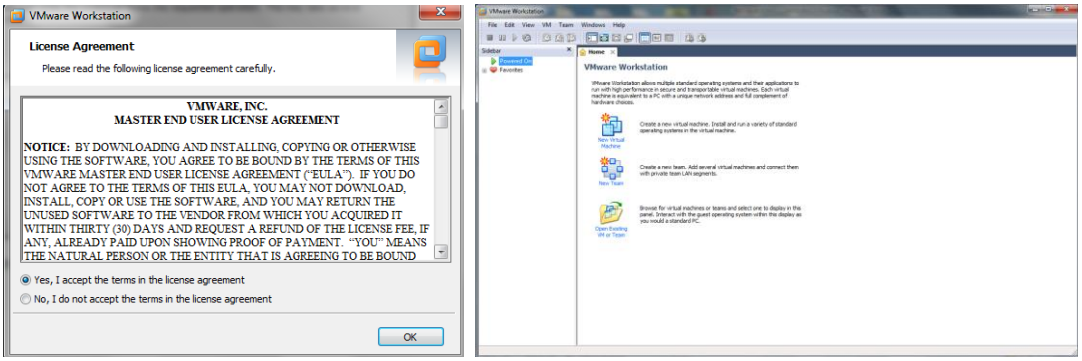

Figura 4.3.1.5 Términos de la licencia y ejecución del VMWare

4.3.2 ANEXO 2: Configuración de la máquina virtual UBUNTU 11.04

Los siguientes pasos son para configurar la máquina virtual Ubuntu 11.04. Creamos una nueva máquina virtual “New Virtual Machine”; seleccionamos la configuración recomendada haciendo clic en “Typical” y continuamos; seleccionamos desde donde vamos a instalar la nueva máquina virtual, si es de un CD/DVD o si es a través de una imagen de disco y continuamos con la instalación.

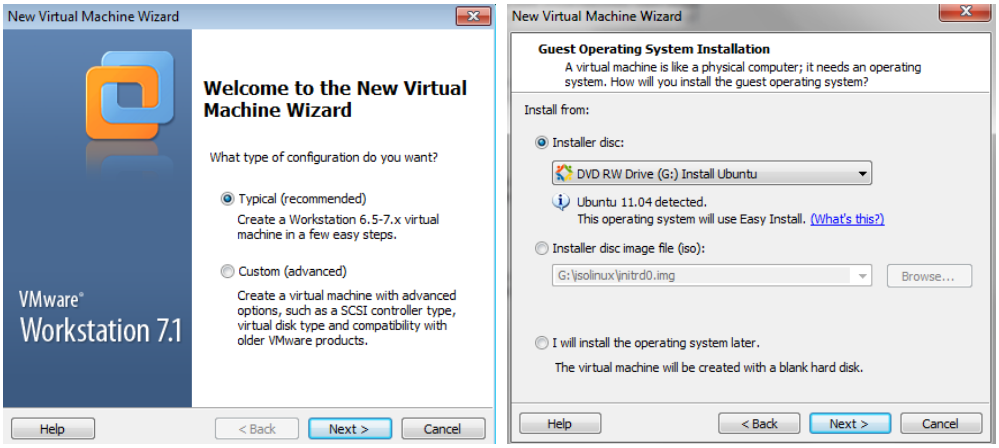

Figura 4.3.2.1 Selección de modo y sistema operativo Ubuntu

Seleccionamos el nombre de usuario y la contraseña deseada, en nuestro caso elegimos para esta máquina:
User name: juan
Password juan

Seleccionamos el nombre como queremos que se llame esa máquina, por defectos aparece con el mismo nombre de Ubuntu 11.04 y la ubicación.

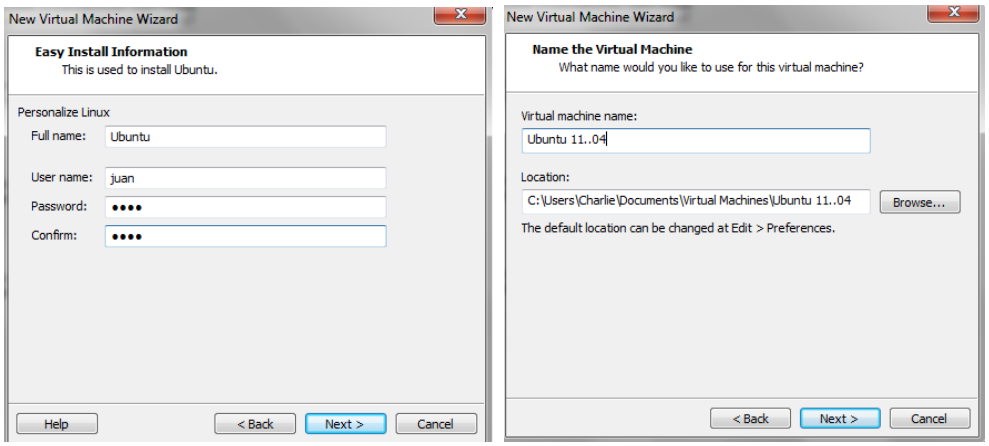

Figura 4.3.2.2 Usuario, contraseña y nombre de la máquina Ubuntu

Seleccionamos el tamaño del disco que va a ser 10GB de memoria; a continuación me muestra a detalle la máquina virtual a configurarse, con lo cual si estamos de acuerdo damos clic en “Finish” con lo cual terminamos de configurar la máquina virtual y pasamos a la instalación de Ubuntu sobre este disco.

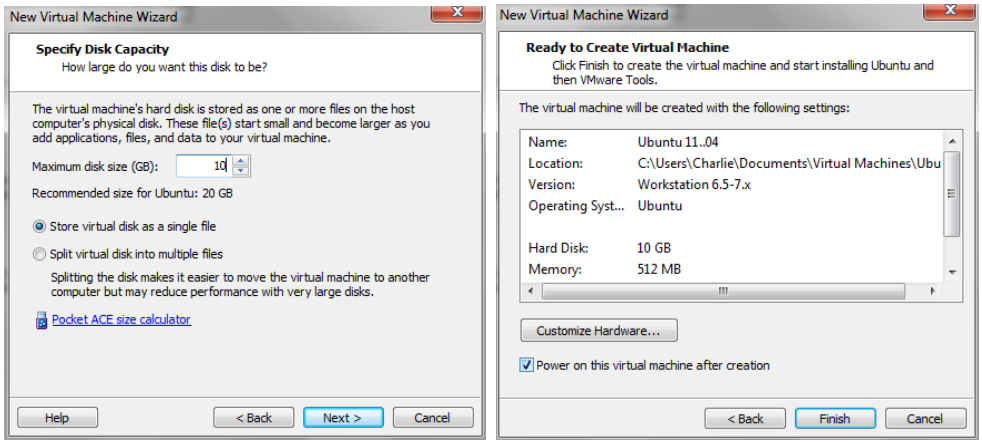

Figura 4.3.2.3 Capacidad del disco y resumen de configuración de Ubuntu

Esperamos un momento a que se cargue el software de instalación de Ubuntu y automáticamente se instala el sistema operativo.

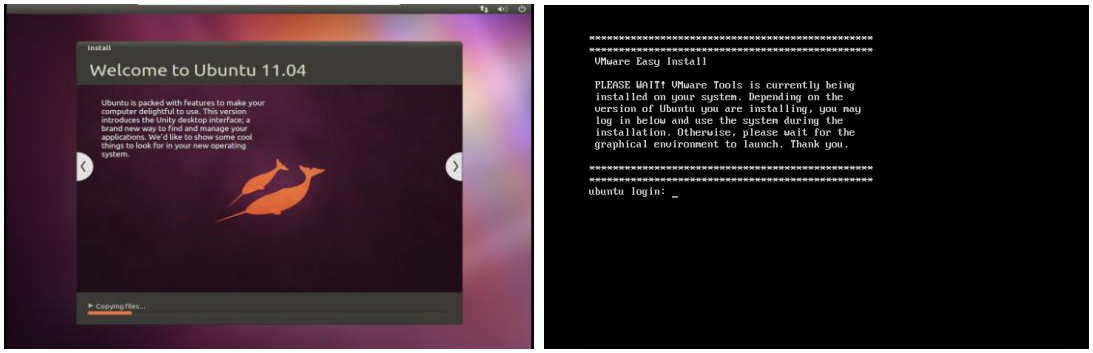

Figura 4.3.2.4 Instalación automática de Ubuntu

Una vez terminada la instalación ingresamos nombre de usuario y contraseña creada, y con esto ya esta lista la máquina virtual Ubuntu para ser usada.

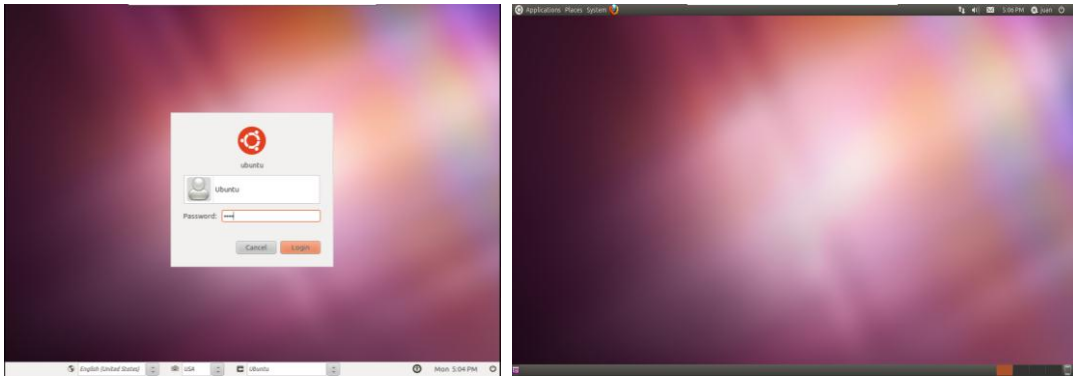

Figura 4.3.2.5 Inicio de sesión y ejecución de Ubuntu

4.3.3 ANEXO 3: Configuración de la máquina virtual CENTOS 5.3

Los siguientes pasos son para configurar la máquina virtual CentOS 5.3. Creamos una nueva máquina virtual “New Virtual Machine”; seleccionamos la configuración recomendada haciendo clic en “Typical” y continuamos; seleccionamos el directorio desde donde vamos a instalar la nueva máquina virtual, si es de un CD/DVD o si es a través de una imagen de disco y continuamos con la instalación.

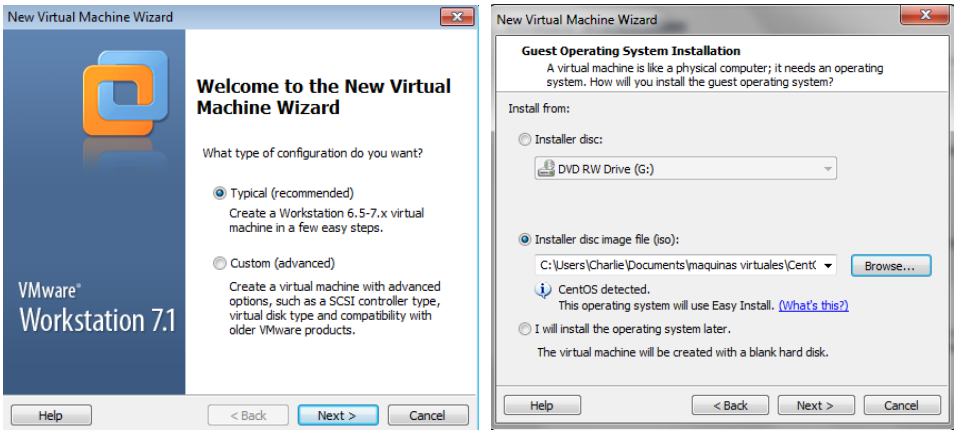

Figura 4.3.3.1 Selección de modo y sistema operativo CentOS

Seleccionamos el nombre de usuario y la contraseña deseada, en nuestro caso para esta máquina:
User name: jose
Password: jose

Seleccionamos el nombre como queremos que se llame esa máquina, por defectos aparece con el mismo nombre de CentOS 5.3 y la ubicación.

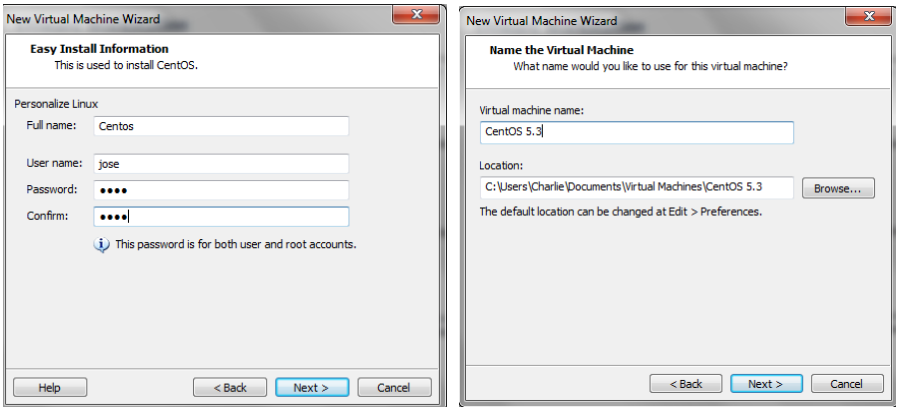

Figura 4.3.3.2 Usuario, contraseña y nombre de la máquina CentOS

Seleccionamos el tamaño del disco que va a ser 10GB de memoria; a continuación me muestra a detalle la máquina virtual a configurarse, con lo cual si estamos de acuerdo damos clic en “Finish” con lo cual terminamos de configurar la máquina y pasamos a la instalación de CentOS sobre este disco.

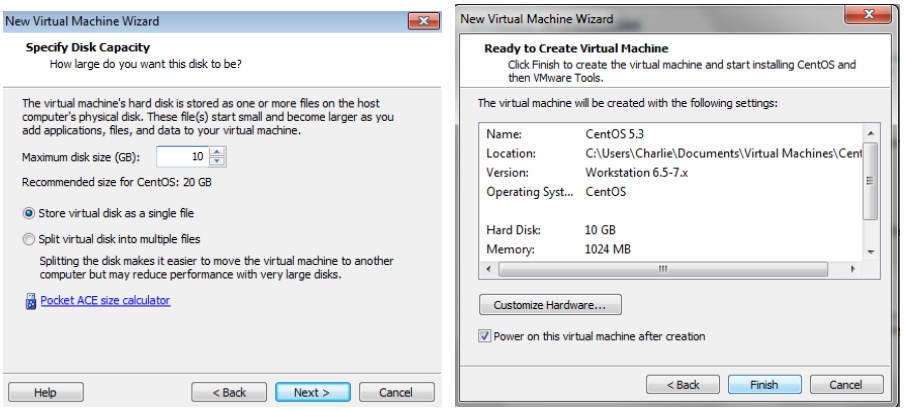

Figura 4.3.3.3 Capacidad del disco y resumen de configuración de CentOS

Después de configurada la máquina se me abre la ventana de instalación y esta comienza automáticamente, después de que finalice introducimos nombre de usuario y contraseña, también podemos seleccionar el idioma en esta misma ventana, y con esto ya esta lista la máquina virtual CentOS para ser usada.

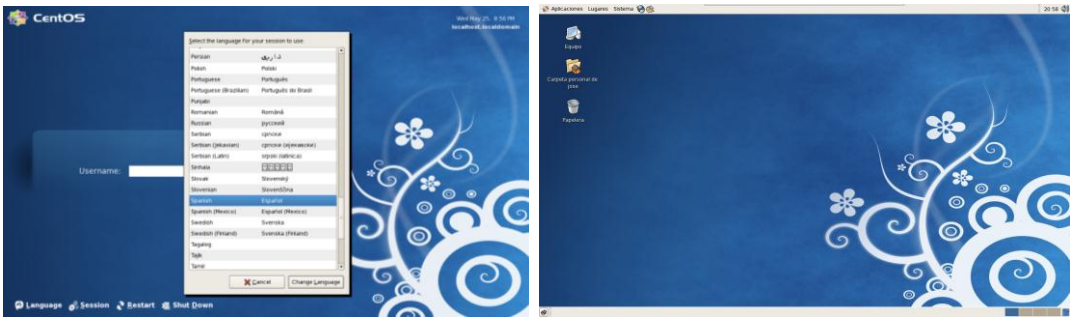

Figura 4.3.3.4 Inicio de sesión, idioma y ejecución de Ubuntu

4.3.4 ANEXO 4: Configuración de la máquina virtual FEDORA 14

Los siguientes pasos son para configurar la máquina virtual Fedora 14. Creamos una nueva máquina virtual “New Virtual Machine”; seleccionamos la configuración recomendada haciendo clic en “Typical” y continuamos; seleccionamos desde donde vamos a instalar la nueva máquina virtual, si es de un CD/DVD o si es a través de una imagen de disco y continuamos con la instalación.

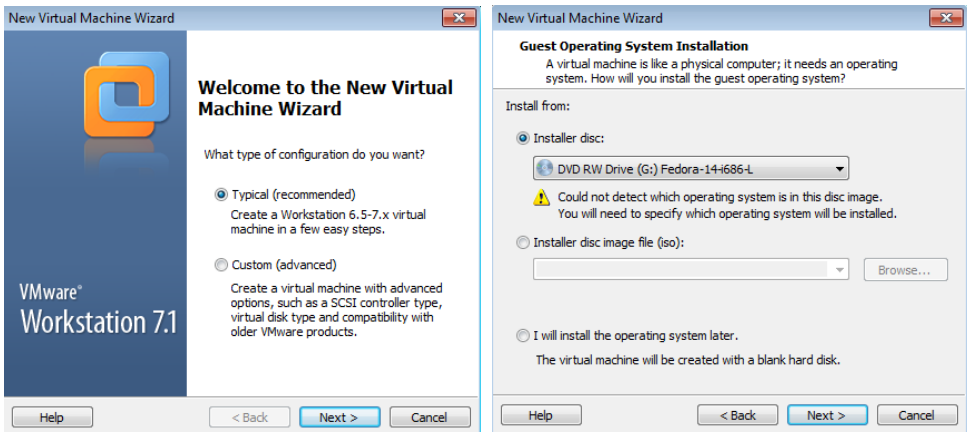

Figura 4.3.4.1 Selección de modo y sistema operativo de Fedora.

A continuación Seleccionamos el tipo de sistema operativo Linux y de la versión Fedora, después seleccionamos el nombre como queremos que se llame esa máquina y la ubicación.

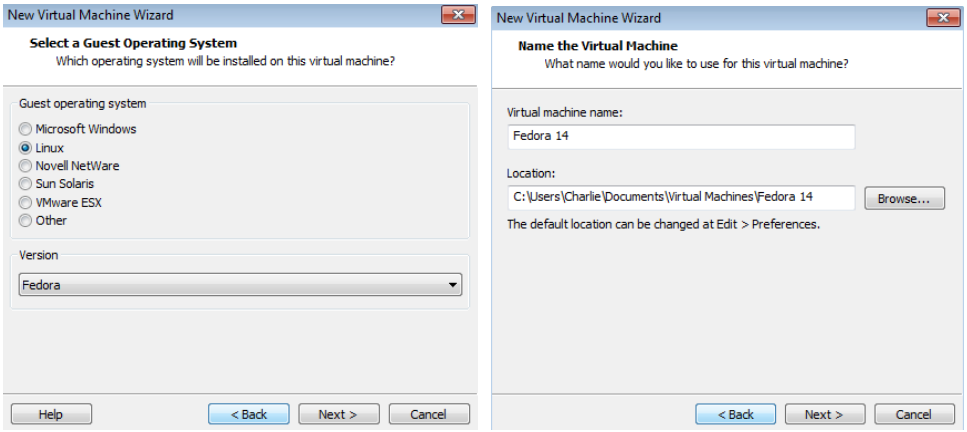

Figura 4.3.4.2 Selección del sistema operativo y nombre de la máquina de Fedora

Seleccionamos el tamaño del disco que va a ser 10GB de memoria; a continuación me muestra a detalle la máquina virtual a configurarse, con lo cual si estamos de acuerdo damos clic en “Finish” con lo cual terminamos de configurar la máquina y pasamos a la instalación de Fedora sobre este disco.

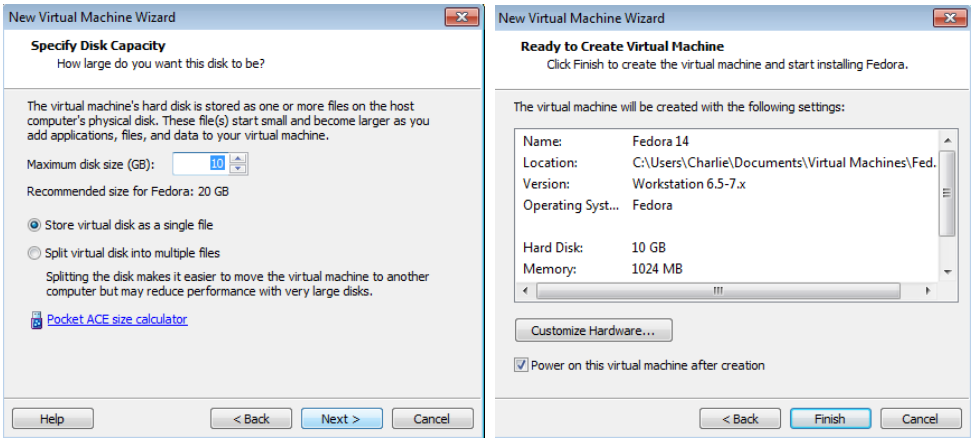

Figura 4.3.4.3 Capacidad del disco y resumen de configuración de Fedora

Después de configurada la máquina se me abre la ventana de arranque del sistema operativo desde el DVD; una vez iniciada la sesión automática, en el escritorio tengo la opción de instalar el sistema operativo al disco duro, con lo cual voy a seleccionar esa opción que es nuestro objetivo.

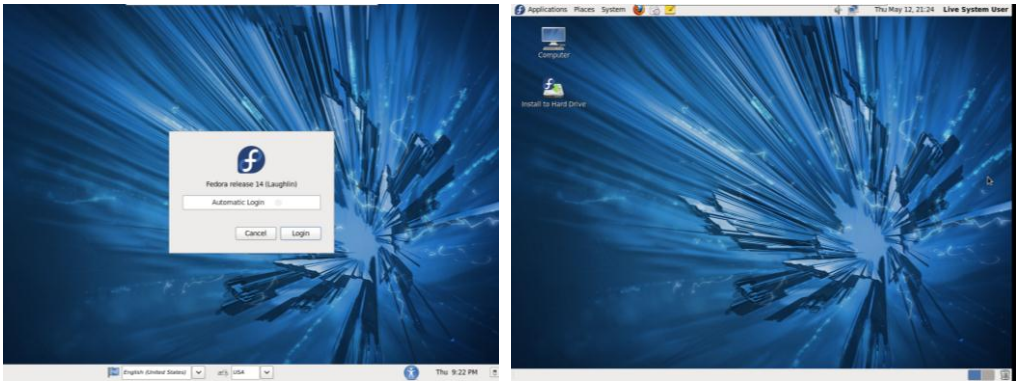

Figura 4.3.4.4 Sesión automática y arranque desde el DVD de Fedora

Ejecutando la instalación del sistema operativo al disco duro, voy a configurar todos los parámetros necesarios; configuro el teclado, así como también el tipo de dispositivos de almacenamiento con los que voy a trabajar, en nuestro caso seleccionamos “Basic Storage Device”.

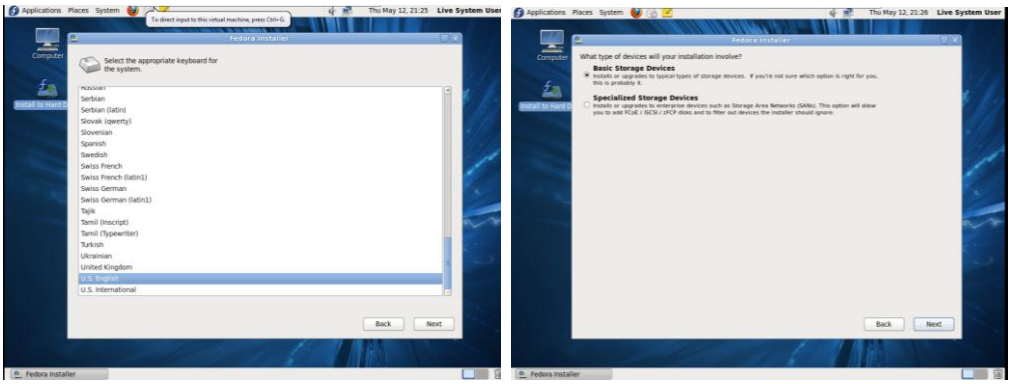

Figura 4.3.4.5 Configuración de teclado y dispositivos de Fedora

En la siguiente pantalla selecciono el nombre de servidor con el cual se me va a identificar mi computadora en una red; siguiente a esto configuro la zona horaria, tengo que seleccionar la cuidad en la que nos encontramos o la más cercana, en nuestro caso seleccionamos “Guayaquil”.

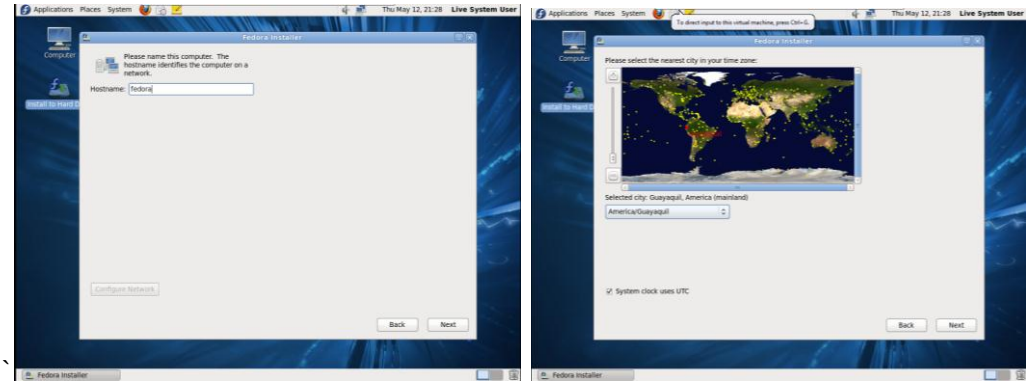

Figura 4.3.4.6 Configuración del nombre del servidor y zona horaria de Fedora

Después configuramos la contraseña y la confirmamos para validar que la contraseña esté correcta; y continuamos con el proceso de configuración, la siguiente pantalla me pregunta por el tipo de instalación que deseamos realizar, en nuestro caso seleccionamos “Replace Existing Linux System”, con lo que voy a hacer la instalación completa del sistema operativo Fedora en todo el disco.

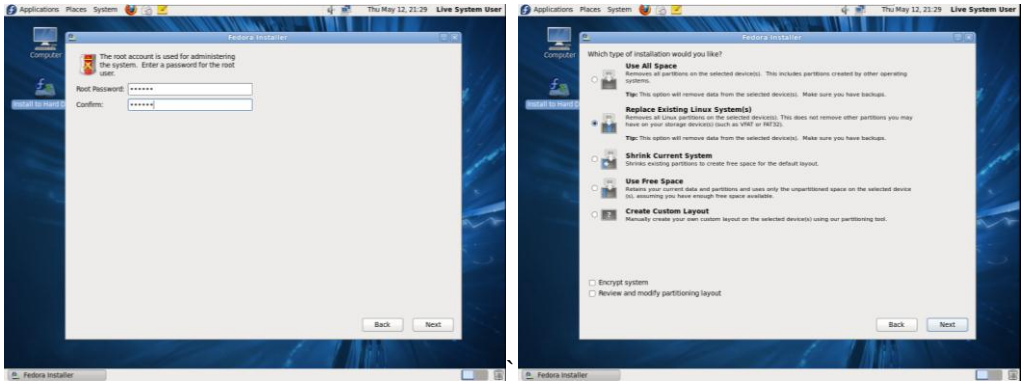

Figura 4.3.4.7 Configuración de la contraseña y tipo de instalación de Fedora

La siguiente ventana me va a indicar que el disco será reformateado y que se perderán todos los cambios, aceptamos en “Write Changes to Disk” y la instalación del sistema operativo al disco va a comenzar.

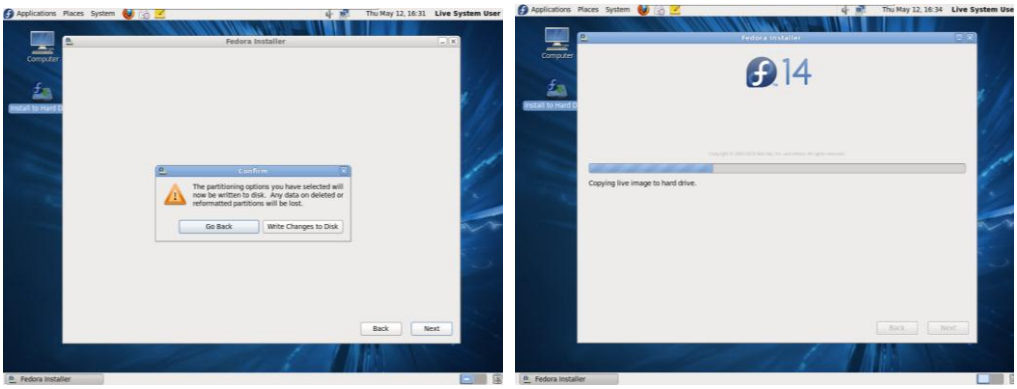

Figura 4.3.4.8 Instalación del sistema operativo de Fedora

Una vez terminada la instalación me va a aparecer una ventana indicándome que la instalación está completa, cerramos esta ventana, vamos a System, Shut down, y hacemos clic en restart con el fin de que esta vez ya me arranque desde el disco virtual y ya no del DVD; y sacamos el disco de instalación del computador.

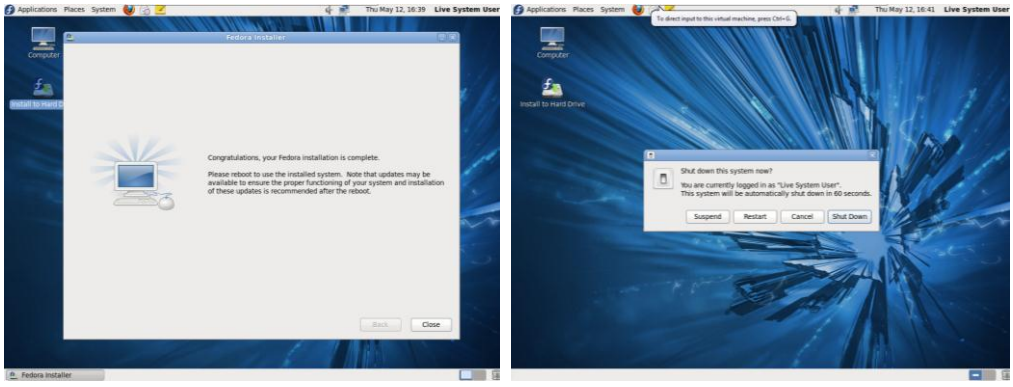

Figura 4.3.4.9 Instalación completa y reinicio de la máquina de Fedora

A lo que se me inicia la máquina me va a aparecer la pantalla de bienvenida; a continuación me la a aparecer la información de la Licencia del Sistema operativo Fedora

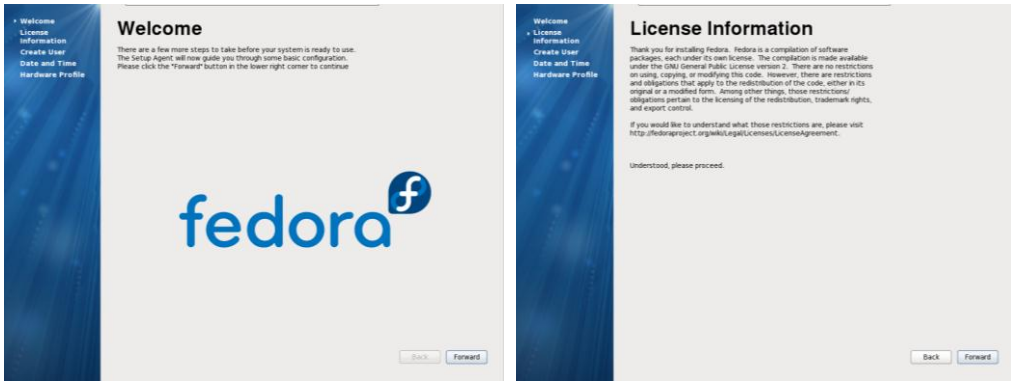

Figura 4.3.4.10 Bienvenida e Información de Licencia de Fedora

A continuación vamos a crear un usuario y su contraseña, así como también vamos a configurar la fecha y la hora. En nuestro caso seleccionamos como usuario y contraseña:

Username: carlos
Password: carlos

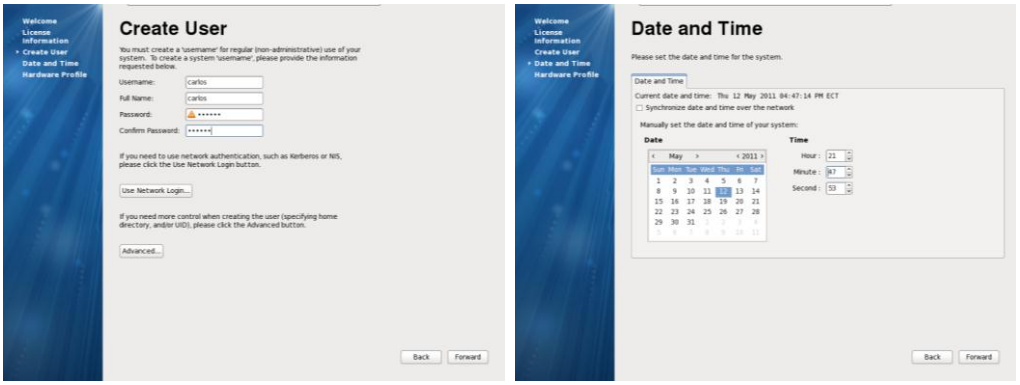

Figura 4.3.4.11 Nombre de usuario, contraseña, configuración de hora y fecha de Fedora

La siguiente ventana me muestra la información del Hardware de la máquina virtual, y tengo la elección de enviar al Centro de Desarrollo de Fedora para su registro, en nuestro caso elegimos que no y confirmamos que no queremos enviar el registro.

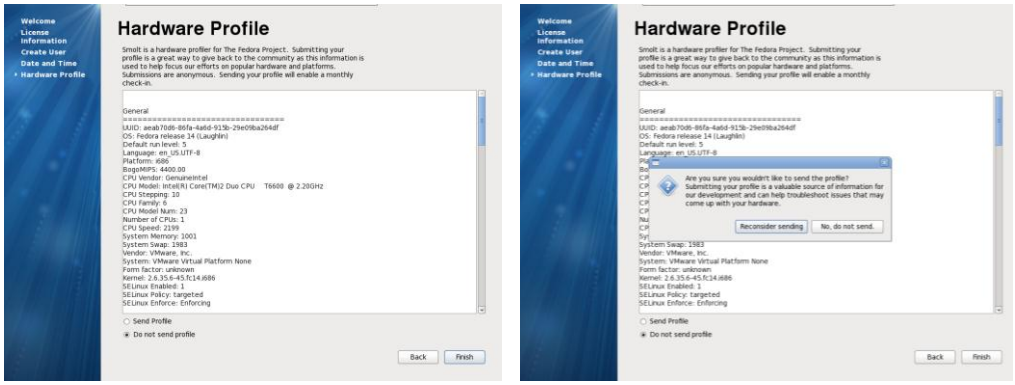

Figura 4.3.4.12 Registro de Hardware de Fedora

La siguiente ventana ya me muestra la pantalla definitiva para el inicio de sesión en el sistema operativo Fedora, y con esto ya estoy listo para comenzar a trabajar.

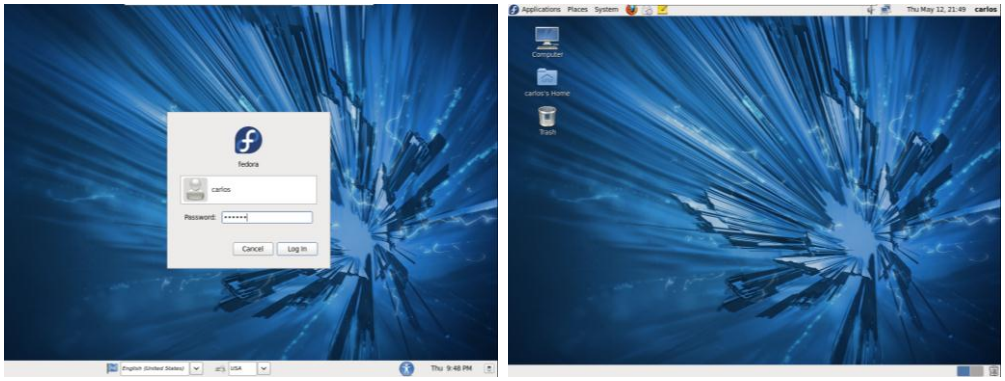

Figura 4.3.4.13 Inicio de sesión y ejecución de Fedora

4.3.5 ANEXO 5: Configuración de la máquina virtual Windows 7

Los siguientes pasos son para configurar la máquina virtual Windows 7. Creamos una nueva máquina virtual “New Virtual Machine”; seleccionamos la configuración recomendada haciendo clic en “Typical” y continuamos; seleccionamos desde donde vamos a instalar la nueva máquina virtual, si es de un CD/DVD o si es a través de una imagen de disco y continuamos con la instalación.

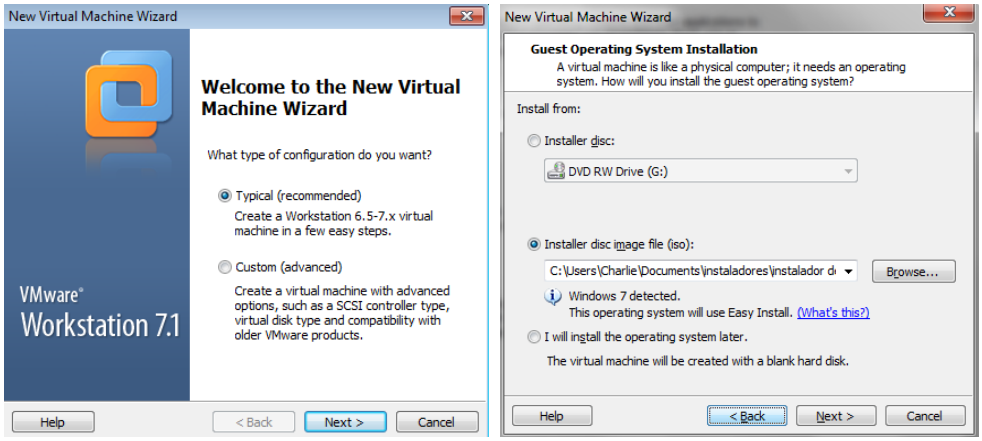

Figura 4.3.5.1 Selección de modo y sistema operativo de Windows 7.

A continuación introducimos la llave de producto, la versión de Windows, seleccionamos el nombre de usuario y la contraseña deseada, en nuestro caso elegimos para esta máquina:
Username: pedro
Password: pedro

Seleccionamos el nombre como queremos que se llame esa máquina, por defectos aparece con el mismo nombre de Windows 7 y la ubicación.

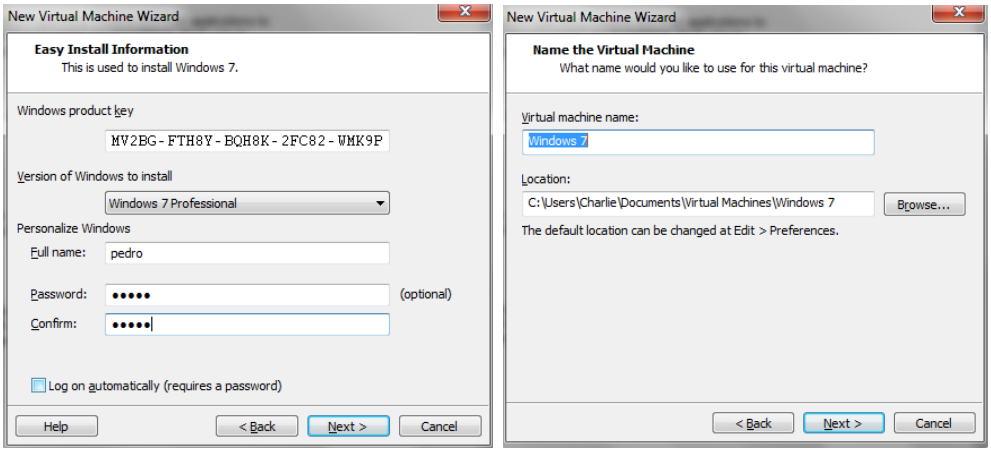

Figura 4.3.5.2 Usuario, contraseña y nombre de la máquina Windows 7

Seleccionamos el tamaño del disco que va a ser 10GB de memoria; a continuación me muestra a detalle la máquina virtual a configurarse, con lo cual si estamos de acuerdo damos clic en “Finish” con lo cual terminamos de configurar la máquina virtual y pasamos a la instalación de Windows sobre este disco.

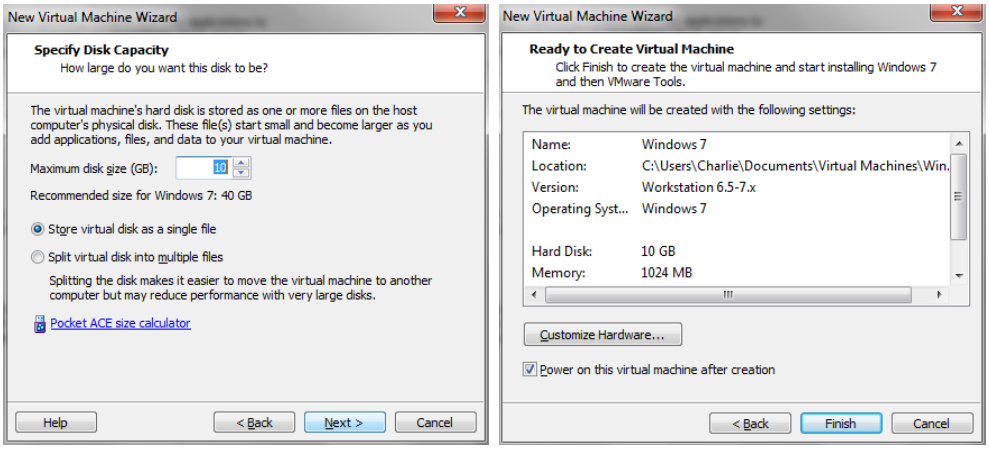

Figura 4.3.5.3 Capacidad del disco y resumen de configuración de Windows 7

Esperamos un momento a que se copien los archivos al disco para su instalación y automáticamente se instala el sistema operativo. Una vez terminada la instalación ingresamos nombre de usuario y contraseña configurada.

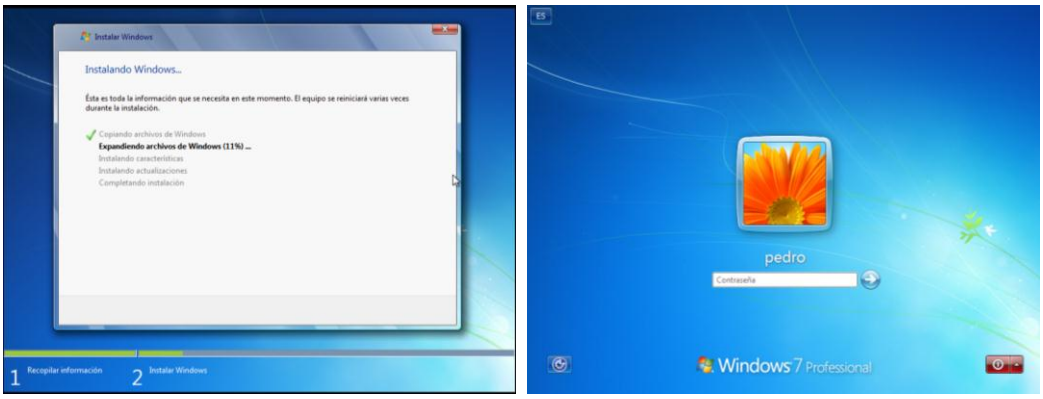

Figura 4.3.5.4 Instalación automática e Inicio de sesión de Windows 7

Y con esto ya esta lista la máquina virtual de Windows 7 para ser usada.

Figura 4.3.5.5 Ejecución de Windows 7

CONCLUSIONES Y RECOMENDACIONES

En el presente trabajo monográfico se ha implementado un Honeypot sobre un servidor con el fin de capturar las acciones de un atacante que ha ganado acceso por SSH. Un Honeypot es una herramienta de seguridad informática utilizada para recoger información sobre los atacantes y sus técnicas, simulando ser un sistema o servidor vulnerable a ataques. Se ha escogido KIPPO, debido a que presenta una mejor alternativa al uso de los Honeypots, debido a sus herramientas de captura de datos y la reproducción de la pantalla de trabajo del atacante en tiempo real, datos ingresados, errores de escritura y tiempos de espera que presentan una gran ventaja.

Se ha redactado cada uno de los pasos a seguir desde las configuraciones de hardware e instalaciones de cada uno de los sistemas operativos ya que es importante tener el conocimiento de la máquina sobre la que se trabaja.

La configuración del servidor se realizo desde la configuración del puerto de acceso SSH para el control de la máquina, hasta la asignación dinámica de cada una de las máquinas que se conectan a la red de trabajo del servidor.

En la configuración de KIPPO sobre el servidor se trabajo mucho para poder realizar y redactar de una forma clara, precisa y de fácil comprensión para el lector, debido a que en las investigaciones realizadas no se encontró una documentación precisa que explique de forma interpretativa la correcta configuración, teniendo en cuenta que casi toda la documentación consultada esta en el idioma inglés.

Otra de las investigaciones en las que tomo bastante tiempo fue en la configuración y uso del software de ataque por fuerza bruta, que consiste en el ataque por diccionario a través del puerto SSH debido a que existen muchas restricciones para el uso de estos programas ya que son programas de ataque a máquinas por lo cual no es fácil de encontrar en la web.

El análisis de los registros de datos no son difíciles de interpretar, por lo cual se obtiene los datos del atacante y la posibilidad de reproducirlos en tiempo real.

Se recomienda la realización de cada uno de los pasos seguidos en el presente trabajo monográfico ya que se ha resumido, pero sin omitir ningún paso, y se ha mostrado de manera gráfica los requerimientos para la implementación y la funcionalidad de KIPPO; el no seguir uno de los pasos podría ocasionar que el sistema no funcione correctamente.

En la implementación de KIPPO en un servidor real se necesita que se revise periódica o diariamente los registros; dependiendo el grado de exposición del servidor, por lo cual toda empresa debería contar con un ingeniero encargado del área de la seguridad en sus redes, y así poder mejorar la seguridad en dicha red dependiendo de la vulnerabilidad.

BIBLIOGRAFÍA

REFERENCIAS BIBLIOGRÁFICAS:

KIPPO SSH Honeypot, página oficial proyecto KIPPO, http://code.google.com/p/kippo/ [consultada 28 de febrero del 2011]

Infosanity's Blog, Comenzando con KIPPO, http://blog.infosanity.co.uk/2010/07/06/starting-with-kippo/ [consultada 01 de marzo del 2011]

Infosanity's Blog, Registros de KIPPO, construcción y estadísticas iniciales de KIPPO Honeypot, http://blog.infosanity.co.uk/category/honeypot/kippo/ [consultada 01 de marzo del 2011]

Xombra Team, Honeypots (Servidores Trampa), http://www.xombra.com/go_articulo.php?articulo=56 [consultada 03 de marzo del 2011]

Universidad de Wisconsin, Dave Woutersen & Dave De Coste, Que es KIPPO? Más allá del ataque de fuerza bruta SSH, http://www.govcert.nl/binaries/live/govcert/hst:content/community/symposium-2010/speakers/dave-woutersen/dave-woutersen/govcert:documentResource/govcert:resource [consultada 07 de marzo del 2011]

Articuloz, Directorio de artículos, Seguridad Para Pequeñas Y Medianas Empresas En Ecuador, http://www.articuloz.com/seguridad-articulos/seguridad-para-pequenas-y-medianas-empresas-en-ecuador-2048082.html [consultada 08 de marzo del 2011]