

Colaboradores:

Jose Noe Poveda
Diego Osorio
Steven Sierra
Andrés Acosta
Laydi Bautista
Fernando Pineda
Marlon Cárdenas
Sebastian Sanchez
Jesus David Romero
Alejandro Cortazar
Jorge Ulises Useche
Juan Carlos Velandía
Leidy Marcela Aldana

Publicidad

Occam's Razor Direct
occams-razor@uvigo.es

Impresión

Por ahora tu mismo... Si te
apetece

©2017 GLUD Magazine
Esta obra está bajo una licencia
Reconocimiento 2.5 España de
Creative Commons. Para ver
una copia de esta licencia, visite
creativecommons.org



EDITORIAL

Grupo GNU Linux
Universidad Distrital
Francisco José de Caldas

El Grupo GNU Linux Universidad Distrital agradece a cada uno de los lectores de esta revista, y a los desarrolladores del código fuente de la misma, es decir, a los encargados de la revista Occam's Razor.

El Grupo GNU Linux es un Grupo Lider en la apropiación, desarrollo, uso y difusión de tecnología, ciencia y cultura libre; el cual, hace parte de la Universidad Distrital Francisco José de Caldas.

GLUD Magazine fue fundada en el año 2011, seis años después, para este segundo volumen publica su segundo número para compartir artículos acerca de ciencia, tecnología y cultura libre.

Como grupo, esperamos que sea agradable para usted leer esta revista y que sirva como aporte a la comunidad.

GRUPO GLUD.

Las opiniones expresadas en los artículos, así como los contenidos de los mismos, son responsabilidad de los autores de éstos.

Puede obtener la versión electrónica de esta publicación, por medio de la página:

glud.udistrital.edu.co

Así como el *código fuente* de la misma y los distintos ficheros de datos asociados a cada artículo en el repositorio de la revista en Github:

github.com

Tendencias de la Ingeniería de Software en 2017.

Trends in Software Engineering in 2017.

Steven Sierra Forero, Ingeniero de Sistemas

Resumen

Este paper expone de una manera generalizada las tendencias de la ingeniería de software en el 2017, aquí se expondrán los campos más relevantes en el desarrollo de software que están haciendo cambios significativos en los flujos de trabajo de las grandes compañías de desarrollo de software y por ende cambiando la forma en la que accedemos a la tecnología.

Metodologías como las contempladas por Agile, apoyando el desarrollo rápido de aplicaciones, despliegue de software automatizado en la nube con Kubernetes, y el uso intensivo de servicios en la nube como AWS en el ámbito de nubes privadas y empaquetamiento de software a través del uso de tecnologías de contención de software como Docker han sentado un precedente en la actualidad reevaluando constantemente la forma de hacer el delivery de productos de software y cambiando el pensamiento de quienes se involucran en el día a día de la ingeniería de software.

Palabras clave: Cloud computing, código abierto, ingeniería, software.

I. Introducción

En la actualidad la ingeniería de software ha tenido que enfrentar diversos cambios adaptándose a la necesidades de la industria, metodologías de planeación, construcción y evaluación del software han sido reevaluadas siguiendo el ritmo de la globalización.

A través de un recorrido general sobre el estado del arte de las tendencias en la ingeniería de software, se tratarán temas como el desarrollo de software en el campo del código abierto u open source, el desarrollo y uso de aplicaciones móviles, metodologías como las contempladas por Agile y finalmente computación en la nube, las cuales dan cuenta del avance de la ciencia y la tecnología en el siglo XXI los cuales promueven cambios positivos para el desarrollo de la sociedad.

II. OPEN SOURCE

El software de código abierto ha tenido grandes avances en los últimos años, un referente muy conocido es RedHat, quien tomó el sistema operativo Linux y lo adaptó a las necesidades de un mercado corporativo, haciendo uso de su código de fuente abierta. El código abierto ha evolucionado desde los días en que Microsoft lo etiquetó como *un destructor de la propiedad intelectual*, hoy en día una nueva generación de líderes de Microsoft espera que su plataforma Azure basada en Linux (con sus SDKs de código abierto) proporcione un crecimiento significativo a la compañía ya sus clientes.

Históricamente la tecnología open source, no ha sido respetada en el sentido de ser malinterpretada como software no estable o software sobre el cual no es viable hacer desarrollos corporativos de gran escala. Sin embargo en la actualidad existen organizaciones como Linux Foundation que tienen proyectos que no solo controlan Linux, sino también Cloud Foundry, el Proyecto Hyperledger (enfocado en la cadena de bloqueo empresarial), la Open Connectivity Foundation (que recientemente se fusionó con la antigua Alianza AllSeen y que se centra en la Internet de las Cosas), La Fundación JS (centrada en JavaScript).

También la fundación Apache (Apache software foundation) mantiene la tradición de código abierto de software libre a través de su estricta licencia con una comunidad bastante activa. Actualmente, Apache tiene más de 180 proyectos, incluyendo nombres conocidos como CloudStack, Hadoop, Cassandra, Groovy, Kafka, Mesos, Maven y Tomcat.

Según la revista Forbes [3], en el evento anual que se lleva a cabo en los Estados Unidos, donde se exponen las últimas tendencias en el mercado del software de código abierto (open source) algunos de los proyectos más relevantes que se expusieron fueron:

- Realidad aumentada y Realidad Virtual, esto se evidenció con Pokémon Go un juego para móviles que estremeció el mercado de las aplicaciones móviles con la integración de realidad aumentada y el uso de librerías de código fuente.
- Inteligencia artificial la cual parecía el material de las películas no hace mucho tiempo. El aprendizaje no supervisado hecho por la IA es un gran salto en el desarrollo de software. En este año se esperan sistemas que realmente puedan aprender y cambiar su comportamiento, abriendo el camino para dispositivos más inteligentes. Ejemplo de ello son todos los productos que ahora están en el mercado en liderado por los gigantes de la tecnología como Google, Amazon, Apple, Tesla e IBM quienes quieren potenciar el uso de inteligencia artificial en todos los productos de software que hacen parte del diario vivir de los consumidores de servicios digitales [1].

III. APLICACIONES MÓVILES

Los teléfonos inteligentes se han convertido en una parte inseparable de la vida cotidiana de muchas personas, empezando por la alarma en la mañana, el uso de WhatsApp, la música en la nube en aplicaciones como Spotify, videos en YouTube, correos electrónicos, con Gmail, Yahoo, Outlook, todas estas aplicaciones que se ejecutan sin problemas desde nuestros dispositivos móviles. Actualmente 2.1 billones de personas en todo el mundo posee un Smartphone, una persona en promedio usa su teléfono unas 264 veces al día, incluyendo textos y llamadas, es un número alto, pero corresponde a una realidad cotidiana en 2017, por lo tanto, el uso de Smartphone y aplicaciones móviles no se va a reducir en 2017 al contrario las tendencias de las aplicaciones para móviles pueden evidenciar nuevos horizontes en el desarrollo de software.

Tecnologías como las implementadas por Google en el proyecto AMP (Accelerated Mobile Pages) en donde Google ha anunciado que habrá un índice de búsqueda independiente para la web móvil abre un campo revolucionario que cambia completamente las tendencias de desarrollo de aplicaciones móviles, especialmente desde SEO y la perspectiva de aplicaciones web. Un ejemplo de las aplicaciones móviles desarrolladas en el ámbito corporativo puede ser Evernote a través del cual diferentes equipos en las compañías pueden colaborar en proyectos desde sus dispositivos móviles. Por otro lado, Facebook messenger y aplicaciones de listas de tareas son ejemplos de micro-aplicaciones que están soportando los procesos corporativos de las compañías en 2017.

IV. METODOLOGÍAS ÁGILES

En tanto que las metodologías Ágiles y DevOps no desaparecerán de ninguna manera, los líderes de software empresarial pasarán de tener visión de escalar en

los procesos ágiles y hacer un esfuerzo en la integración de profesionales DevOps ampliando la cadena de valor en el delivery de software de punta a punta. En 2017 hay una tendencia generalizada por la adopción de equipos de profesionales DevOps en procesos ágiles lo cual permite una maduración de las organizaciones en la entrega de software a un punto en el que están dispuestos a ver la agilidad de manera holística.

A medida que los líderes de software empresarial comienzan a enfocarse en su agilidad empresarial holística y en sus flujos de valor, buscan una solución definitiva en el reto de unificar la planificación estratégica, el desarrollo ágil y la implementación de DevOps.

Durante muchos años, las grandes organizaciones empresariales se han centrado en acelerar la entrega escalando ágil. Muchas de estas organizaciones también se han centrado en mejorar la planificación estratégica inicial con prácticas ágiles. En los últimos años estas organizaciones han comenzado a centrarse en los beneficios que DevOps puede tener para la entrega de productos de software en un menor tiempo, lamentablemente, cada una de estas iniciativas se realiza típicamente de forma aislada dando como resultado un esfuerzo fragmentado que no llega a toda la organización.

La mayoría de los flujos de valor de las organizaciones empresariales se dividen en tres fases principales [3]: estrategia, desarrollo y entrega. En un flujo de valor fragmentado, cada fase evoluciona en sus propios equipos, procesos, herramientas e información que no están totalmente integrados con el resto del flujo de valor. Esta evolución ha creado un entorno en el que, aunque la adopción ágil y de DevOps es alta, los resultados no satisfacen las expectativas de liderazgo.

Una buena cantidad de organizaciones empresariales reconocidas se han esforzado en los últimos años en el mejoramiento del proceso ágil optimizando los flujos de valor de toda la empresa [2].

Esta es una transformación importante que no es fácil de lograr. Afortunadamente, la lección aprendida por grandes organizaciones es un punto de partida para otras organizaciones que están empezando en el mercado de la ingeniería de software.

V. CLOUD COMPUTING

La computación en nube ha reducido drásticamente el costo del despliegue de software. En lugar de necesitar equipos de cómputo y hardware especializado, espacio en el centro de datos, nómina de administradores de sistemas, el software se puede contratar de los proveedores de nube con unos pocos clics y un costo relativamente bajo por hora. El éxito ya no es un problema, ya sea: construya bien su software y los recursos adicionales de la nube le ayudarán a escalar hacia arriba y hacia abajo según sea necesario. Las pequeñas empresas e incluso los ingenieros de software individuales pueden proporcionar el mismo nivel de servicio en la nube que las grandes empresas. El porcentaje de empresas en la actualidad que tienen una estrategia para utilizar despliegue [6] de software en la nube creció hasta el 85 % (vs 82 % en 2016), con un 58 % de planificación híbrida (vs. 55 por ciento en 2016). También hubo un aumento en el número de empresas que planean múltiples nubes públicas (de un 16 % a un 20 %) y una disminución concurrente en las que planean múltiples nubes privadas (de 11 % a 7 %). En 2017, los desafíos para el despliegue de software en la nube se redujeron en todos los ámbitos, con la excepción de la gobernanza y el control, que se mantuvo estable [3].

Como parte de la adopción de los procesos de DevOps para el despliegue de software en la nube, las empresas han optado por implementar nuevas herramientas que les permitan estandarizar y automatizar la implementación y configuración de servidores y aplicaciones. Estas herramientas incluyen herramientas de gestión de configuración (como Chef, Puppet y Ansible), más recientemente, tecnologías de contenedores, como Docker y orquestación de contenedores y herramientas de programación como Kubernetes, Swarm y Mesosphere.

La encuesta sobre el estado de la nube de 2017 realizada por la Cloud Native Computing foundation revela que aunque AWS continúa liderando la adopción de la nube pública (57 % de los encuestados actualmente ejecutan aplicaciones en AWS), este número se ha mantenido plano desde 2016 y 2015. Es importante señalar que mientras que el porcentaje de las empresas que ejecutan al menos una aplicación en AWS es plana, el número de aplicaciones y máquinas virtuales que están ejecutando está aumentando, con lo que el aumento de los ingresos de AWS. El aumento sustancial en el uso de contenedores ahora hace Docker la herramienta de DevOps superior entre los incluidos en nuestra encuesta. La adopción general de Docker subió al 35 %, tomando la ventaja sobre Chef y Puppet en 28 %.

La computación en nube ha ayudado a muchas empresas a transformarse en los últimos cinco años, pero los expertos están de acuerdo en que el mercado está

entrado en una nueva etapa, tanto para cloud público como para cloud computing construido o alojado en centros de datos corporativos. El mercado de la nube se ve creciendo a un buen ritmo en 2017 y lo seguirá haciendo medida que las empresas busquen obtener eficiencias a medida que escalan sus recursos computacionales para servir mejor a los clientes.

Según Dave Bartoletti, "Las empresas con grandes presupuestos, centros de datos y aplicaciones complejas ahora están viendo a cloud como un lugar viable para ejecutar aplicaciones empresariales básicas". Esto se evidencia en las estadísticas entregadas por Amazon donde se evidencia que con la primera ola de cloud computing creada como Amazon Web Services, en 2006 ha dado utilidades por más de 11.000 millones de dólares.

Referencias

- [1] insights.dice.com
- [2] www.oreilly.com
- [3] forbes.com
- [4] Barry W. Boehm. Get ready for the agile methods, with care. IEEE Computer, 35(1):64–69, Jan 2002.
- [5] Alan C. Wills. Agile components. Technical report, Trireme International Ltd, Feb 2002. www.ltt.de
- [6] Mihaly Csikszentmihalyi. Flow: The Psychology of Optimal Experience. Harper Perennial Modern Classics. 2008.
- [7] Jeannette M. Wing. Computational thinking. Communications of the ACM, March 2006. Page 33- 35.

Steven Sierra Forero Estudiante de Ingeniería de Sistemas, Universidad Distrital Francisco José de Caldas.
ssierraf@correo.udistrital.edu.co



Emisora en línea con Tecnologías Libres

Online Broadcasting Station with Free and Open Source Technologies

Jorge Useche, Ingeniero Electrónico
Alejandro Cortázar, Bachiller Académico

Resumen

En nuestra sociedad la radio hace aún parte importante de nuestras vidas; tal vez escuchada más por personas que han venido envejeciendo con estos medios y desplazada por el Internet una nueva forma más eficiente de obtener información de todo tipo sin necesidad de publicidad ni cosas innecesarias porque siempre hay muchas opciones de donde escoger.

Es necesario recuperar la calidez humana que nos brinda escuchar la información de otra persona, imaginar de cierta forma y creer en un mundo que es generado solo por una voz; aprovechar el Internet para que la gente de distintos rincones accedan a la información y a los pensamientos que queremos compartir. El espíritu de compartir se ve truncado en veces por no acceder a las tecnologías por ser muy caras y ahí es donde entra el software libre a permitirnos compartir lo que queramos, y para la intención anterior existe un conjunto de herramientas que permitirán que tengamos nuestra propia emisora en Internet o netradio, muy fácil de manejar y gratuita.

Palabras clave: Radio, Online, Comunitaria.

I. Introducción

Siempre es mejor confiar en la estabilidad de un sistema GNU/Linux que en la de otros sistemas operativos; para este fin se recomienda ampliamente utilizar una distribución como Debian GNU/Linux o algo aún más libre como Trisquel GNU/Linux ya que nos ofrece paquetes de excelente calidad y muy estables. Por esta razón aparte de la cuestión moral de que sea libre, únicamente se utilizará tecnologías y contenidos libre que cumpla las 4 libertades del software según Richard Stallman adaptadas al tipo de contenido. La creación del sistema para la emisora en Internet se abordará con el siguiente método:

1. ¿Por qué una radio en línea?
2. Conceptos generales sobre servicios o demonios en GNU/Linux.

3. Terminología usada en la transmisión de medios.

4. Piezas de software usadas.

5. Alternativas para emisión de audio y ventajas y desventajas de éstas.

6. ¿Qué música puedo poner?, licencias asociadas.

7. Ilustración general del método usado en Radio GLUD.

8. Instrucciones para la primera emisión en una red local.

Con esto claro se entienden ya los alcances de este escrito y se quiere dejar en claro, que para entender lo que se hace para crear la emisora en línea es necesario que se expliquen los conceptos relacionados y la razón de todo lo que hallamos tenido que decidir, para que entendiendo esto se pueda automatizar procesos o mejorar la forma en que se hicieron, de lo contrario el artículo que convertiría en un tutorial más.

Radio GLUD
Libertad para tus oídos.

Antes de desarrollar los contenidos, me parece necesario decir porque se cree tener el criterio necesario para escribir este artículo. Desde hace un tiempo existe la emisora en línea del GLUD, esta emisora se llama "Radio GLUD", y desde sus comienzos ha dado valioso conocimiento a todo el grupo con el cual se han solucionado algunos problemas con soluciones definitivas o parciales en que a pesar de todo el trabajo invertido hasta ahora, todavía falta un camino largo por recorrer para acercarla lo más posible a lo que puedan imaginar nuestras mentes. La invitación cordial va a que visiten el sitio web de este proyecto <http://glud.udistrital.edu.co/radioglud>

II. ¿Por qué una radio en línea?

Es una pregunta que se hizo a su debido momento, por lo general se presentan muchas cosas a favor y otras en contra, pero las razones más valiosas que se encontraron fueron positivas y más enfocadas en *impacto social* que en cuestiones técnicas.

La madre de todas es el poder social que se presenta cuando las personas comparten algo, pero ¿cuál era el medio correcto para compartir?; la radio es nuestro ancestro más cercano (para comunicarnos), todavía vivo, más interesante y nos ofrece las siguientes ventajas derivadas del poder social.

El primero de ellos es *el sentido de comunidad*, que se crea cuando muchas personas trabajan por una misma causa y es conocido que es mucho más fácil hacer un programa radial que uno televisivo; el segundo es *compartir ideas y sentimientos*, que se ha visto mucho en otras radio, pero ¿realmente son las que nos interesan en la comunidad académica?; la tercera es *las experiencias que se adquieren*, no puede ser solo libros, pueden haber otros espacios de esparcimiento muy provechosos; el cuarto es *compartir gustos musicales*, que hace que estemos más unidos, o ¿quién no ha cantado una canción que a todo mundo le gusta con un grupo de amigos?; y la última, *la creatividad como reflejo de sociedad*, tenemos derecho a expresarnos y nos volvemos más creativos (A mi parecer) solo mostrando nuestra voz que nuestros rostros.

III. Conceptos generales sobre servicios o demonios en GNU/Linux

Primero abarcaré la forma en que el sistema operativo gestiona programas como icecast (ver sección Piezas de software usadas). Dicho programa se gestiona como un servicio (daemon en sistemas unix) donde la principal característica es que es un programa que es persistente con una ejecución continua, se ejecuta en segundo plano y no posee una interfaz de usuario directa o shell.

Disk And Execution MONitor es lo que significa DAE-MON; “según una investigación realizada por Richard Steinberg, la palabra fue utilizada en 1963 por primera vez, en el área de la informática, para denominar a un proceso que realizaba backups en unas cintas, este proceso se utilizó en el proyecto MAC del MIT y en una computadora IBM 7094,1 dicho proyecto estaba liderado por Fernando J. Corbató, quien afirma que se basó en el demonio de James Maxwell”[1]

Hay dos formas de iniciar un servicio si éste no se encuentra activo, la primera es ejecutando el programa relacionado con los parámetros para iniciarlo como daemon que indique el manual y la segunda es por medio de un mediador como service (programa en shell) que facilita la tarea ya que no hay que recordar la ruta ni el comando de ejecución del servicio.

IV. Terminología usada en la transmisión de medios

La transmisión de medios específicamente de audio, representa la acción esencial del proceso de crear una emisora, hay que tener claro que el proceso no es más que el streaming de audio que posee una interfaz web para poder escucharlo cómodamente.

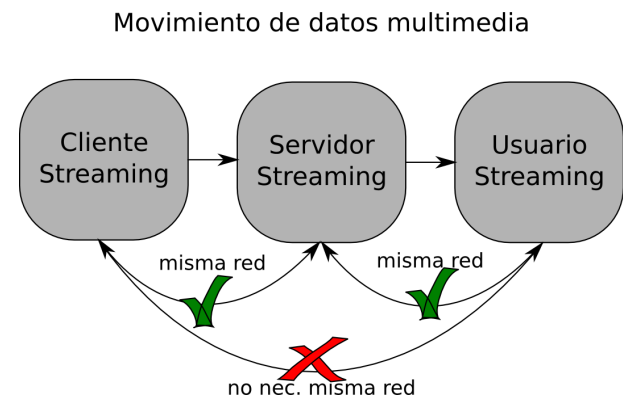


FIGURA 1: Gráfico del movimiento de los datos multimedia desde la generación del fuente hasta que lo escucha el usuario o oyente de la radio o canal de vídeo

El streaming (que se refiere a corriente continua) es la distribución de contenido multimedia a través de una red de computadores con la ventaja de que por medio de un búfer de datos se agrega la posibilidad de disfrutar el contenido mientras este descarga; el componente búfer de datos en este caso es un espacio de memoria que almacena datos para evitar que la reproducción del medio se quede sin información durante la transferencia.

Latencia (relay - retardo del reloj) es el tiempo o retardos temporales en que se realiza un proceso o procesos (o en su defecto ninguno) debido por lo general al tiempo de propagación de los circuitos o incluso al tiempo en que se estabiliza que es debido a la densidad del material y la temperatura principalmente.

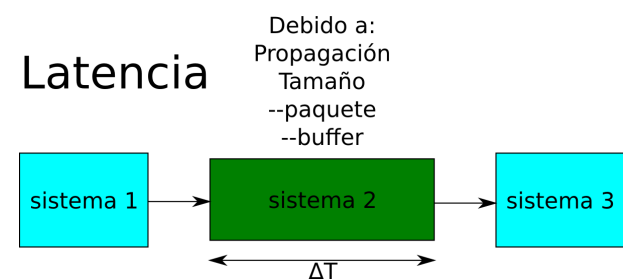


FIGURA 2: Gráfico de un sistema intermedio (sistema 2) donde se muestra que existe un tiempo desde que los datos entran hasta que salen del sistema y los causales posibles de ese Delta

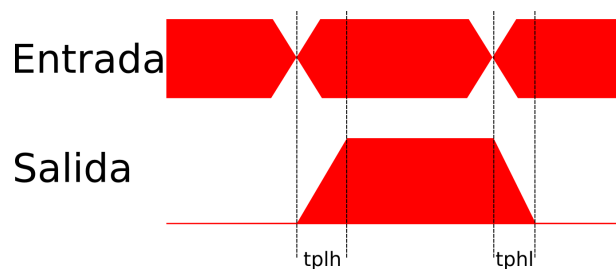


FIGURA 3: Gráfico de los tiempos de propagación de un sistema, comenzando con el tiempo de propagación de bajo a alto (tplh) y terminando su respuesta con el tiempo de propagación de alto a bajo (tphl)

El tiempo real (Real Time - RT), se presenta esta referencia para todos los sistemas en donde hay que considerar que ese tiempo real no tiene que ser necesariamente sinónimo de rapidez o inmediatez; como concepto, dicho tiempo es la latencia suficiente para que el sistema pueda resolver el problema para el cual está dedicado. Para el caso del audio en informática existe un tiempo de respuesta crítico con un porcentaje que no exceda el 3 % (aunque por lo general es así con todos los sistemas RT) que para llevarse a cabo necesita algunos componentes esenciales como un sistema operativo (OS) que soporte el tiempo real (para el caso de un sistema GNU/Linux solo sería el kernel) y un software diseñado para esto (el mismo OS RT ofrece un marco para el desarrollo de dichas aplicaciones).

Hay que distinguir entre 2 tipos de sistemas RT, los pasivos y los activos donde la diferencia es el daño o fallo que puede causar un retraso en la señal; por ejemplo tenemos que un sistema de audio o vídeo en directo sería un sistema RT pasivo debido a que un retraso en la señal simplemente será un retraso en la visibilidad del vídeo o escucha del audio tal vez con desmejora del multimedia pero luego seguirá funcionando normal como si nada hubiera pasado y un ejemplo de un sistema RT activo son la mayoría de sistemas médicos (como el marcador de pasos artificiales o marcapasos) donde un retraso de la señal puede causar el fallo del entorno donde el sistema interfiere causando una desmejora apreciable del paciente o incluso la muerte.

El servidor de sonido para streaming es un software con la función de gestionar los flujos de datos reproduciendo las muestras. Cualquier programa reproductor, grabador, gestor de audio, necesita soporte para trabajar con un determinado servidor de audio por ejemplo Jack Audio Connection Kit (en adelante simplemente JACK) es uno de los más conocidos para flujos en tiempo real.

El servidor de streaming de medios es un sistema de distribución de multimedia por streaming que utiliza un tipo de formato y un protocolo determinados para realizar el proceso de distribución de contenidos multimedia, para el caso de icecast el formato puede ser ogg, webm, aac o mp3 y el protocolo http o https.

El flujo multimedia se refiere al audio o al vídeo que

es enviado desde un punto de la red hasta otro. Este flujo debe ser continuo aunque no necesariamente estable (fluctuante), pero debe tener un mínimo de velocidad de transferencia de un punto a otro para que el streaming sea efectivo, hay que introducir entonces algunos conceptos relacionados a las propiedades intrínsecas del archivo multimedia que se quiere enviar por streaming para estimar esa velocidad mínima que debe tener la red. Hay que tener en cuenta que cualquier archivo multimedia (audio y/o vídeo) posee un bitrate, que es la cantidad de bits que posee el flujo en unidad de tiempo, y que obedece a la composición del archivo, se mide por lo general en kbps (kilo bit por segundo).

También es importante conocer el término multibitrate que surgió de la necesidad de enviar una fuente multimedia a diferentes calidades, por ejemplo una calidad HD, normal y móvil; para cada una de estas es necesario un bitrate distinto para que el usuario pueda consumirlo de acuerdo a sus necesidades o limitación de ancho de banda de su conexión.

Ya que introducimos los bitrates y las velocidades de conexión de la red, es necesario entrar un poco a hacer el análisis de requerimientos de ancho de banda. Cuando queremos enviar un flujo multimedia desde nuestro computador hasta un servidor de streaming es pertinente revisar la velocidad de conexión (de subida) a este servidor para enviar nuestro flujo o flujos; para llevar a cabo este análisis es necesario realizar un test de velocidad o Speed Test para saber nuestra velocidad real de conexión; hay que tener en cuenta que pueden presentarse fluctuaciones inesperadas en esa velocidad máxima por lo tanto es recomendable enviar flujos con bitrates que sumados estén dentro del rango de un 50 % a un 70 % de la capacidad total, una vez tengamos el dato podemos saber cuantos flujos y de que calidades podemos enviar a el servidor; por ejemplo si nuestra conexión es de 512 kbps se deben hacer los cálculos sobre la base de 358,4 kbps que es el 70 % de la conexión, con esto se puede enviar audio con bitrates de 256 kbps y 64 kbps sin mucho problema.

Source Stream (fuente) es entendido como la fuente de audio (o vídeo) de donde se obtienen los datos para después distribuirlos en la red en forma de streaming, este fuente se genera desde un cliente como ices o idjc de los que se hablará más adelante.

Mount Point (Punto de Montaje) es una ruta por la cual se puede acceder a determinado streaming, debe ser configurada en el servidor de streaming, un ejemplo de un punto de montaje es `/radioglud.ogg` y para acceder a él desde un navegador web o un reproductor de audio la dirección se escribe de la forma `http://ip:puerto/radioglud.ogg`, no hay que preocuparse si no se ha entendido del todo ya que al hacer nuestro proceso de streaming en localhost el concepto llega sin mucho esfuerzo.

También vale la pena saber algunos conceptos de envío de información conocidas como *formas de transmisión de información* en una red, en lo que se consideran 3 como los principales conceptos, hay que decir que existen más pero éstos son los más relevantes y usados, el primero es *unicast* que consiste en que se envía la información uno a uno, donde obtiene la información el que lo necesita a través de requerimientos; el siguiente es *multicast* que necesita un rango de IPs ya reglamentado (conocidas como tipo D) para enviar simultáneamente los datos a todos los destinos escogidos a los que se les conoce como grupo multidifusión, no necesita enviar la información a cada destino por petición, sino que realiza una única transmisión de los fuentes a cada uno de los destinos y por eso no necesita una red con gran velocidad para realizar las transferencias; y el último de los tres es la forma *broadcast* que es parecido al anterior con la diferencia que se envía los datos a todos los puntos de la red o subred simultáneamente.

Icecast es unicast y hasta donde conozco no tiene soporte multicast, uno de los indicios principales para esta conclusión es el protocolo HTTP (Protocolo de Transferencia de HiperTexto) que es TCP (Protocolo de Control de Transmisión) con el que icecast distribuye el streaming, además de que supongo que como he leído desde hace un tiempo a pesar de que la multidifusión tiene muchas ventajas sobre el bandwidth (ancho de banda de la conexión de Internet) su tráfico es realmente complejo además de que su implementación a gran escala produce puntos de fallos que pueden ser utilizados para ataques de denegación de servicios. Como dato adicional solo los protocolos diseñados específicamente para trabajar con multicast son efectivos para su implementación y la mayoría de las aplicaciones conocidas lo hacen por UDP (Protocolo de Datagrama de Usuario).

Poder Social se presenta cuando las personas comparten.

V. Piezas de software usadas

A continuación se listan las aplicaciones que se han usado en Radio GLUD a través de su desarrollo, más adelante se dirá cuales de estas se usan actualmente.

- JACK: Es un servidor de sonido (daemon) de baja latencia, GNU GPL.
- Icecast: Es un proyecto y a la vez un programa servidor de streaming de medios, creado por la Xiph.Org Foundation, su propósito es crear un flujo de medios para una estación de radio, jukebox privado (rockola) y otros, a partir de un source client, GNU GPL v2.
- Ices: Es un programa generador de fuentes (source client) para un servidor de streaming, su pro-

pósito es generar un flujo de audio (stream) para el servidor de streaming del proyecto ICECAST, GNU GPL.

- IDJC: Cliente generador de fuentes para un servidor de streaming, es mucho más versátil que ices ya que permite mezclar distintos flujos de audio, provenientes de cualquier dispositivo o programa (incluso de si mismo) que pueden estar gestionados principalmente por el servidor de audio local. Su función es principalmente servir de estudio de audio casero, GNU GPL v2.
- QjackCtl: Es una aplicación QT, que nos da la posibilidad de controlar algunos parámetros de JACK (audio server daemon) de una forma simple y gráfica, GNU GPL v2.
- JACK Rack: Es un programa con un conjunto de efectos para JACK a través de complementos (plugins) su procesamiento del audio es a baja latencia, GNU GPL.
- JAMin: Es una interfaz muy poderosa de compresión y ecualización de audio, GNU GPL.
- Liquidsoap: Es un lenguaje de programación de audio para generar source streams, al ser un interprete con un lenguaje script dedicado la versatilidad y la adaptabilidad lo hace lo suficientemente poderoso para acomodarse a cualquier emisora, por complejo que se quiera realizar el(los) source stream(s), GNU GPL.
- jme (jmediaelement): Es una aplicación web, reproductor de audio y vídeo a través de elementos de HTML5.
- Otras aplicaciones de audio muy útiles en <http://jackaudio.org/applications>.

Jack Audio Connection Kit multipataforma y lo mejor que el software libre ha podido hacer por el mundo de la música.

Alternativas para emisión de audio y ventajas y desventajas de estas

Son muchas las aplicaciones que se utilizan en el entorno comercial para la emisión de medios, aunque contamos con pocas opciones de software libre, todas son de gran calidad y muy estables. Se listan las alternativas privativas que conozco como para dar ciertas observaciones al respecto, las que no conozco debido a que nunca las he usado, ni escuchado, ni desearía hacerlo aparecen con un “—”.

Semi-equivalentes entre software libre y privativo	
Alternativa libre	Alternativa privativa
Icecast	SHOUTcast
Icecast	oddcast o edcast
Ices, Darkice	Winamp, VLC
IDJC	Winamp, Zazaradio
QjackCtl	—
JACK Rack	—
JAMin	Winamp
liquidsoap	—
jme (Reproductor HTML5)	Reproductor en Flash
WebcamStudio	WebcamMax

Cuadro 1: Equivalentes lo más parecido posible a las herramientas libres

Hace algún tiempo vi una conferencia en el CPCO4 (Campus Party Colombia), la verdad no recuerdo el nombre de la persona y no la encontré por la web por eso no la cito, en esta se enseñó como hacer una emisora online con software privativo. Algunas de las quejas del expositor en la conferencia eran, “shoutcast solo funciona con winamp”; “winamp solo trabaja con la tarjeta de audio escogida como principal”; “la configuración de archivos de texto con notepad es feo (eso puede ser debido a que pone el texto todo de un solo color y posiblemente no se aprecia la indentación como debe ser)” y dijo también “estoy más perdido que el hijo de limber” y por las constantes fallas en el proceso decía “eso le pasa al chavo, al coyote y a mí”. Mis observaciones fueron que el equipo se colgaba fácilmente, la razón es que posiblemente el software era muy pesado, empezando por el sistema operativo.

Una de las cosas que si se hacían bien con el software privativo era cuando usaba herramientas como oddcast y zazaradio aunque en sus versiones gratis poseen algunas limitaciones en cuanto a funcionalidad. Otra aplicación que usaba era WebcamMax que sirve para generar un fuente de vídeo con imágenes u otros vídeos para posteriormente unirla con el audio y enviarla a un canal en *ustream*, buscando un rato en la web encontré la alternativa libre *WebcamStudio* aunque por estar escrita para Java 1.6 ha resultado hasta el momento imposible hacerlo funcionar con nuevas versiones de Java, solo queda esperar a que la versión alpha siga desarrollándose y llegue a ser estable.

Queda rescatar que tu libertad está sobre todas las cosas, así que aunque algunas veces el software privativo sea más funcional no hay que usarlo porque luego te esclavizaras a los desarrolladores de dicha herramienta.

VI. ¿Qué música puedo poner?, licencias asociadas

Cuando hacemos nuestra propia emisora en línea es necesario conocer algunas leyes colombianas e internacionales que nos impiden poner música de cualquier tipo sin considerar que existen los derechos de autor (patrimoniales y conexos) y que hay que posiblemente pagar por ellos a un ente recaudador de derechos como Sayco y Acinpro (al parecer va a ser cambiada por otro ente debido a la corrupción del mismo), de lo contrario se está cometiendo un delito.

Los derechos de autor tienen distintas clases, entre los que interesan para el desarrollo de la pregunta están:

- Los derechos Morales: son los derechos ligados al autor y perduran por la eternidad, estos derechos son irrenunciables e imprescriptibles (ni ceder, vender o transferir).
- Los derechos Patrimoniales: son los derechos que definen la explotación de la obra (retribuciones por uso, reproducción, difusión) hasta cierto límite después de la muerte del autor, después de este plazo se convierte en lo que se llama dominio público.
- Los derechos Conexos: son los derechos que cobijan a personas distintas al autor (artistas, intérpretes, traductores, editores, productores y otros)

Comparativa de tarifas de Sayco y Acinpro			
Lugar	Categoría	Estrato	Precio
Piojó	6	1	\$278.533
Natagaima	4	2	\$371.472
Ibagué	2	2	\$566.700
Bogotá	E	3	\$1.133.400

Cuadro 2: Datos tomados de http://www.saycoacinpro.org.co/test_tarifas.php con el código 102 (Establecimientos de cualquier naturaleza que para fines de su objeto social vendan productos para llevar y consumir o usar en sitios diferentes al del establecimiento comercial. Ejemplo: Almacenes de cadena, Supermercados, Hipermercados, Bancos, Estación.) y Capacidad/Aforo de 50 personas

Tal vez para una emisora libre de publicidad o una emisora estudiantil, sería muy difícil invertir recursos para la música y tener que suspender el uso de piezas musicales cuando es bien conocido que la música (ver arriba los argumentos) hace parte importante del proceso de comunidad que es lo que se busca en un grupo como GLUD.

Existen algunos tipos de productos musicales debidos al cumplimiento de los derechos anteriormente nombrados y que se podrán utilizar en nuestra emisora teniendo en cuenta que uso es el que se le va a dar

- Música libre: Es la música que cumple los mismos principios del software libre.
- Música con algunos derechos reservados: Es la música que no permiten las libertades de obra derivada y/o uso comercial.

Dentro de la música que se puede usar está la de Dominio Público, mucha de la mejor música del planeta se encuentra bajo esta licencia, entre ellas obras de arte como la música clásica; también encontramos piezas de audio que el autor ha decidido registrar con licencias como Creative Commons, Coloriuris o ArtLibre.

Las licencias más usadas actualmente para contenidos libres o semilibres son las Creative Commons que son 6 licencias que surgen de la conjugación de las características o funciones que son, Atribución (by), No comercial (nc), Sin derivar (nd) y Compartir igual (sa). Las licencias libres[2] son CC-by y CC-by-sa, las otras 4 que son CC-by-nd, CC-by-nc, CC-by-nd-nc y CC-by-nc-sa son las que se consideran semi-libres o con algunos derechos reservados. Pueden buscar su licencia ideal en la dirección <http://creativecommons.org/choose/> y posteriormente poner los meta-datos en tu sitio web.

Estas licencias CC (Creative Commons) según cuenta

la historia surgió debido a la aparición del Internet (hay que darse cuenta la importancia de esta en lo que que consideramos que es la libertad por lo menos en la tecnología) y el espíritu de compartir que hicieron que una persona Eric Eldred fuera subiendo a la red obras que quedaban en Dominio Público cada año, en la escena apareció Mary Bono viuda de Sonny Bono un actor estadounidense que quería que los derechos de autor duraran para siempre, y luego de su muerte, Mary Bono tomó el cargo que él tenía en la cámara de representantes y comenzó a promover lo que ella consideraba la última voluntad de su marido.

Todo este lío resulta ser anticonstitucional pero igual se aumentó los años que se protegía el derecho patrimonial del autor, situación que causó descontento (para Eric Eldred y sus seguidores) ya que para ese año y otros venideros no habría liberación de obras; en este dilema aparece Lawrence Lessig con un argumento contundente “en la constitución de USA decía que se garantizaba el derecho monopolístico por un tiempo limitado para asegurar y promover el progreso”, hubieron discusiones hasta que lo denominado “The Mickey Mouse Protection Act”[3] impuso con su poderío una influencia lo bastante grande para que el tiempo del derecho patrimonial se extendiera sin regulación aparente cada vez que llegaba la hora de que Mickey Mouse pasara a dominio público[5] en 1976 decidieron parar estos cambios casi anuales.

Mickey Mouse ahora se esconde bajo una marca registrada que impide usarlo por ser prácticamente el logo de la compañía The Walt Disney Company y en el transcurso de todo esto Lawrence Lessig creó lo que actualmente es la organización sin fines de lucro Creative Commons como oposición directa al copyright.

Internet y el espíritu de compartir!!!

Ilustración general del método usado en Radio GLUD

Método usado en Radio GLUD

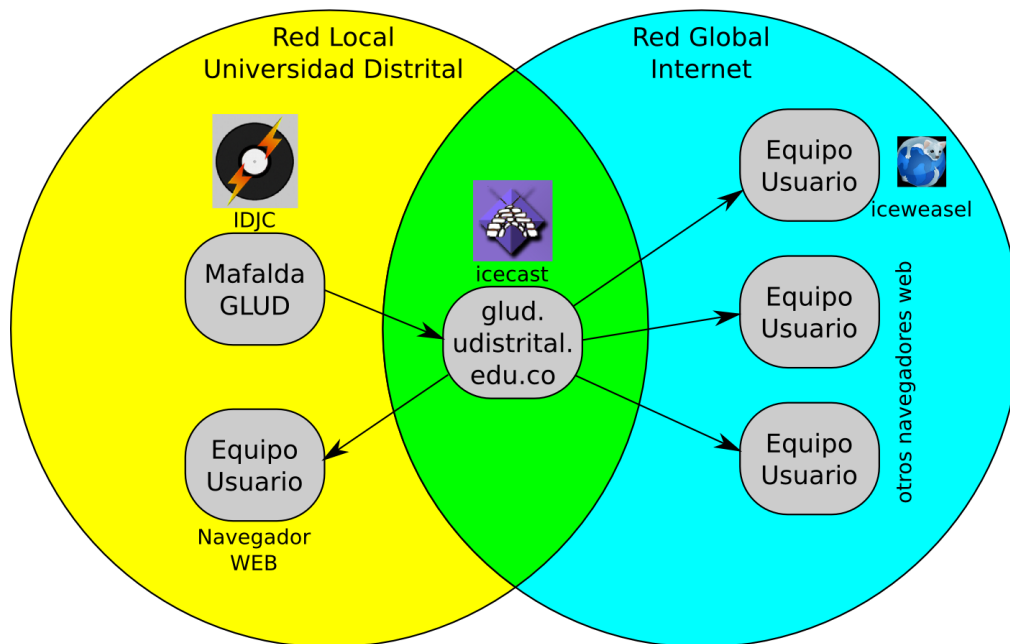


FIGURA 4: Gráfico del método usado en Radio GLUD, consiste en 2 computadores donde el client streaming que genera el source stream es Mafalda GLUD y el servidor de streaming es glud.udistrital.edu.co

Como se puede ver, el método (ha de entenderse método como el esquema general o a grandes rasgos de la gestión, configuración y flujo del audio) implementado hasta ahora no incluye el uso de liquidsoap, y hay dos fuentes (generada por IDJC) y 2 puntos de montaje con baja (64 kbps) y alta (128 kbps) calidad de sonido (configurando el bitrate de las fuentes en IDJC) en el formato ogg vorbis. Se recomienda hacer bien los cálculos para que se pueda dar la suficiente calidad si son pocos o muchos los usuarios.[5] La desventaja de éste es que claramente no hay un sistema de respaldo y en caso de que el IDJC falle, nuestro punto de montaje desaparecerá, se recomienda tener otros fuentes que deberían ser administrados a través de liquidsoap.

“Aquel que sacrifica la libertad por seguridad, no merece ninguna de las dos” – Benjamin Franklin

El ideal para todos estos tipos de sistemas se encuentra en la wiki de RadioNú la dirección de dicho esquema se encuentra en http://wiki.radiognu.org/como_funciona_una_radio éste esquema es la proyección a futuro de Radio GLUD; pongo la imagen para que se pueda apreciar rápidamente.

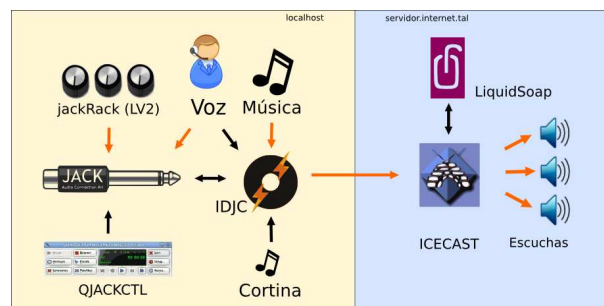


FIGURA 5: Esquema de como debería ser el flujo de datos en un sistema de streaming, extraído de http://wiki.radiognu.org/como_funciona_una_radio

Si desean saber más de las tecnologías usadas en Radio GLUD, pueden visitar la dirección <http://radio/InformeFinaldeActividades-RadioGLUD1.odt> donde se explica detalladamente todo lo que se ha hecho y donde se puede descargar.

Para finalizar esta sección quiero dar crédito a los proyecto RadioNú y Radio Liberación que ha dado mucho a la comunidad hispano-hablante ya que con su documentación han dado un paso increíble para los que no sabemos leer muy bien Inglés (o definitivamente no sabemos).

VII. Instrucciones para la primera emisión en una red local

Ahora que ha terminado toda la teoría se hace necesario ir a lo técnico, pero antes hay que aclarar la notación que se va a utilizar. La distribución en que se hizo fue Fedora GNU/Linux 16 Verne con gnome 3, para evitar todos estos pasos se podría instalar una distribución como Musix GNU+Linux que es una distro dedicada a la música; la notación \$ comando re-

presenta los comandos en una terminal como usuario normal (sin permisos de root o administrador); la notación `# comando` representa la ejecución de los comandos especificados como root, para ello solo basta con digitar `$ su` y escribir la clave de root, en caso de que se maneje con la utilidad `sudo` habría que digitarse `$ sudo su` y la clave del usuario común; en particular debido a que `sudo` es una herramienta muy conveniente para copiar y pegar comandos en terminal sin cambiar de usuario, es la que se utiliza en este mini-tutorial, pero si no se tiene, pueden loquearse como root y ejecutar solamente los comandos que están con `sudo`.

También es importante decir que los pasos no están tan detallados como se quisiera, pero se espera que con la teoría se pueda deducir lo que se dice que se haga o buscar en Internet para que quede claro, por ejemplo como se modifican los archivos con el editor nano, el lenguaje usado en los archivos o incluso como se abre una terminal y que es, ya que lastimosamente no hace parte de lo que se quiere en este artículo, pero por suerte hay mucha información en la red sobre estos temas.



Jorge Ulises Useche Cuellar

Estudiante de Ingeniería Electrónica, Interesado en el desarrollo de tecnologías libres que reduzcan la brecha social que existe hoy en día en Colombia al ser un país subdesarrollado. Viniendo de otro sector de Colombia encontró una simpatía inmediata al conocer el software libre e invierte su tiempo conociendo y desarrollando para la comunidad de software libre que le ha dado tanto.

Referencias

- [1] Copia textual del párrafo 3 del artículo Demonio (informática). Wikipedia la enciclopedia libre. Recuperado de [es.wikipedia.org/wiki/Demonio_\(informática\)](https://es.wikipedia.org/wiki/Demonio_(informática)) el 19 de agosto de 2012.
 - [2] Artículo Música Libre. Wikipedia la enciclopedia libre. Recuperado de es.wikipedia.org/wiki/Música_libre el 10 de agosto de 2012.
 - [3] Artículo Mickey Mouse. Wikipedia la enciclopedia libre. Recuperado de es.wikipedia.org/wiki/Mickey_Mouse el 19 de agosto de 2012.
 - [4] Artículo Copyright Term Extension Act. Wikipedia the free encyclopedia. Recuperado de en.wikipedia.org/wiki/Sonny_Bono_Copyright_Term_Extension_Act el 19 de agosto de 2012.
 - [5] Artículo Estándares para Calidad y Ancho de Banda. RadioNú. Recuperado de wiki.radiognu.org/calidad el 22 de agosto de 2012.
-

Importancia del Software Libre en la Seguridad Informática

Importance of Free Software in Computer Security.

Diego Osorio, Andrés Acosta, Leidy Aldana.
Estudiantes de pregrado en Ingeniería de Sistemas

Resumen

El software libre hace parte de una construcción social que beneficia a la comunidad; en el campo de la Seguridad Informática juega un papel importante con entornos y herramientas potentes que han revolucionado este campo.

Palabras clave: Confiabilidad, Integridad, Disponibilidad, Libertad, Seguridad.

Introducción

Compartir conocimiento ha aportado en la evolución de las sociedades, de forma similar, compartir código ha aportado a la construcción de software de mayor calidad; más allá del código abierto este artículo hace referencia al Software Libre, el cual permite la libertad del usuario y la libertad del software, permite modificar de acuerdo a necesidades propias, permite ayudar a una persona o a una comunidad; el software es libre al tener cuatro libertades, enunciadas en *Software Libre para una Sociedad Libre* (Stallman, 2004, página 236):

0. La libertad cero es la libertad de ejecutar el programa con cualquier propósito.
1. La libertad uno es la libertad de ayudarte a ti mismo cambiando el programa para que se ajuste a tus necesidades.
2. La libertad dos es la libertad de ayudar al prójimo distribuyendo copias del programa.
3. La libertad tres es la de ayudar a distribuir a tu comunidad publicando una versión mejorada de modo que los otros puedan beneficiarse de tu trabajo.

Gracias a estas cuatro libertades el Software Libre ha contribuido y puede contribuir en el aprendizaje, en la construcción y el soporte de grandes empresas o en la enseñanza; de hecho puede ser una alternativa que resulta más interesante, alejada del tedio y más próxima

al área científica. Cabe resaltar que libre es diferente de gratis (libre como pensamiento libre más no como barra libre), que el software libre no quita el reconocimiento de la persona que contribuye a crear software o otro aporte hecho a la construcción de conocimiento libre.

El Software Libre tiene la ventaja de ser estudiado, ejecutado, modificado y distribuido.

Su importancia se ve reflejada en la mejora continua, en que brinda la oportunidad de aprender apartir de aportes realizados por otras personas, se ve reflejada en el hecho de que el conocimiento debería ser libre. Para algunas personas más allá de estas ventajas mencionadas anteriormente, la elección de utilizar Software Libre es una cuestión ética, de ciencia, de no permanecer tanto en la ignorancia, tomando como ejemplo el área de la seguridad informática (área que hace parte de la mayoría de las tecnologías de la información y la comunicación), hay programas no libres que cuentan con código oculto utilizado para conseguir información del usuario o ejecutar procesos sin permiso del usuario.

Se denomina por seguridad informática a las medidas y controles que garantizan la confiabilidad, integridad y disponibilidad de los activos del sistema de información; incluyendo hardware y software, firmware e información procesada, almacenada y comunicada.

El área de la seguridad informática influye en la mayoría de las áreas en la actualidad, como en el desarrollo web, en la inteligencia artificial, en el manejo de grandes volúmenes de información; desde una perspectiva más cotidiana influye en el simple hecho de enviar un correo electrónico (para que llegue sin modificaciones exclusivamente al destinatario), de hacer una transacción bancaria en línea, o realizar la conexión a una red wi-fi sin que la información de nuestros dispositivos sea vulnerable.

La seguridad informática son las medidas y controles que garantizan la confiabilidad, integridad y disponibilidad de los activos del sistema de información.

De esta forma, este artículo busca mostrar al lector la importancia del trabajo conjunto de estos dos campos a través de ventajas, de herramientas, de alternativas.

Herramientas

Las herramientas utilizadas en la seguridad informática que son de código abierto, permiten que la comunidad examine el código, lo entiendan e incluso lo modifiquen, lo que permite a los usuarios aprender acerca del software, ajustarlo a sus necesidades e incluso contribuir. Esto lo hace más versátil e incluso agrega la confiabilidad de usarlo sabiendo que muchas personas han examinado el código y nos da la certeza que hace lo que debe, que en este no se encuentra código malicioso que pueda afectar a quien lo use y que ya ha sido probado.

El fácil acceso a todo tipo de usuarios, dónde no se tiene que pagar por utilizar, y el gran aumento del uso de las tecnologías de la información y el mundo digital, permite que la comunidad que utiliza software libre en la seguridad informática sea cada vez más, en los últimos años ha ganado mucha popularidad, incluso la demanda laboral ha aumentado drásticamente en los últimos años al igual que el uso de software libre a nivel profesional en la seguridad informática y personas especializadas en él.

Al ser probado constantemente por diferentes personas, con diferentes enfoques hace que se encuentren mayor número de problemas y fallos en menor tiempo que un software que sólo es mantenido por una empresa. Al haber más personas que pueden contribuir a mejorarlo, se ve una evolución constante haciendo las herramientas de software libre actualizadas, versátiles y altamente competitivas.

Es bastante la información que se encuentra en Internet de las herramientas de software libre en el campo de la seguridad informática, en los principales blogs se discute y comparte especialmente sobre software de código abierto. Tanto organizaciones como personas individuales comparten constantemente documentación, guías, tutoriales, vídeos, experiencias, preguntas, trucos, etc. generando una gran cantidad de material en múltiples idiomas permitiendo que casi cualquier persona con iniciativa y con algunos conocimientos básicos de la informática pueda aprender a utilizar estas herramientas.

Kali Linux

Es una distribución de software libre la cual esta en constante evolución con nuevas características que se agregan constantemente, esta distribución esta diseñada específicamente para pruebas de penetración profesional y auditoria de seguridad.

Cambios principales de esta distribución

Usuario único, acceso root por diseño:

Es diseñado para trabajar en un único escenario root (usuario principal o administrador del sistema) ya que la mayoría de las herramientas utilizadas en pruebas de penetración requieren privilegios escalonados.

Servicios de red deshabilitados de forma predeterminada:

Kali Linux contiene Demonios (scripts en segundo plano) que desactivan los servicios de red de forma predeterminada. Estos Demonios nos permiten instalar varios servicios en Kali Linux, a la vez que nos aseguramos de que nuestra distribución permanezca segura de forma predeterminada, sin importar qué paquetes estén instalados. Los servicios adicionales, como Bluetooth, también están en la lista negra de forma predeterminada.

Núcleo Linux personalizado:

Kali Linux utiliza un kernel en sentido ascendente, con parches para inyección inalámbrica.

Un conjunto mínimo y confiable de repositorios:

Dado los objetivos y metas de Kali Linux para mantener la integridad del sistema como un todo. Con ese objetivo en mente, el conjunto de fuentes de software ascendente que Kali utiliza se mantiene en un mínimo absoluto. Muchos nuevos usuarios de Kali están tentados a agregar repositorios adicionales a sus *sources.list*, pero al hacerlo corre un riesgo muy serio de romper la instalación de Kali Linux.

¿Es Kali Linux adecuado para usted?

Los desarrolladores de Kali linux recomiendan que esta distribución debe ser usada por probadores de penetración profesional y especialistas en seguridad, ya que no es una distribución para soluciones generales

de GNU/LINUX como el desarrollo de aplicaciones, diseño web, juegos, etc.

El uso indebido de herramientas de seguridad y de pruebas de penetración dentro de una red, en particular sin una autorización específica, puede causar daños irreparables y tener consecuencias significativas, personales y/o legales. "No entender lo que estabas haciendo" no va a funcionar como una excusa.

Sin embargo, si usted es un probador de penetración profesional o está estudiando pruebas de penetración con el objetivo de convertirse en un profesional certificado, no hay mejor juego de herramientas - a cualquier precio - que Kali Linux.

Algunas Herramientas de seguridad y hacking.

Nmap ("Mapeador de redes")

Es una herramienta de código abierto para la exploración de redes y sondeo de seguridad de puertos, se diseña para analizar rápidamente grandes redes, utiliza paquetes IP "crudos" para poder determinar que equipos se encuentran disponibles en la red, esta herramienta generalmente se usa para auditoria de seguridad, administradores de redes, y es útil para realizar tareas rutinarias, como puede ser el inventariado de la red, la planificación de actualización de servicios y la motorización del tiempo que los equipos o servicios se mantiene activos en la red. El código fuente lo puedes encontrar en: github.com/nmap/nmap

John the Ripper

Es un programa libre de código abierto de criptografía, el cual aplica fuerza bruta para descifrar contraseñas, esta herramienta nos permite romper algoritmos de cifrado o hash, como DES, SHA-1 y otros. El código fuente lo puedes encontrar en: github.com/magnumripper/JohnTheRipper

Nikto

Nikto es un software de código abierto (GPL), diseñado para evaluar la seguridad de los servidores web. Este software encuentra varios archivos predeterminados e inseguros, configuraciones y programas en cualquier tipo de servidor web. El código fuente lo puedes encontrar en: github.com/sullo/nikto

Wireshark

Wireshark es un software de código abierto (GPL) el cual lo usan administradores de redes para analizar el tráfico de la red, capturar paquetes de diferentes tipos de protocolos que existen. El código fuente lo puedes encontrar en: github.com/wireshark/wireshark

Metasploit

Metasploit es un proyecto de código abierto para seguridad informática el cual proporciona información acerca de vulnerabilidades de seguridad y nos ayuda a realizar tests de penetración "Pentesting" junto con el desarrollo de firmas para sistemas de detección de intrusos. El código fuente lo puedes encontrar en: github.com/rapid7/metasploit-framework

Referencias

- [1] Stallman, Richard. *Software Libre para una Sociedad Libre*, disponible en pdf. Madrid, España: Traficantes de sueños, 2004.
- [2] 2017 Offensive Security. *Documentación oficial de Kali Linux*, <https://docs.kali.org/>.
- [3] *Guía de referencia de Nmap*, <https://nmap.org/man/es/>.
- [4] RAPID1. *John the Ripper password cracker*, <http://www.openwall.com/john/>.
- [5] Wireshark Foundation. *Documentación oficial de Wireshark*, <https://www.wireshark.org/docs/>.

Diego Osorio Estudiante de Ingeniería de Sistemas, actual director del GLUD. Interesado en el Software Libre, el desarrollo de Software y la Seguridad Informática.
daosoriog@gmail.com

Andrés Acosta Estudiante de Ingeniería de Sistemas.
andres.taamap@gmail.com

Leidy Aldana
Estudiante de Ingeniería de Sistemas, interesada por aprender continuamente; quien ve en el software libre una perspectiva más agradable e interesante de su carrera.
LeidyMarcelaAldana@gmail.com

El aporte del software libre a la accesibilidad en plataformas tecnológicas.

The contribution of free software to accessibility in technology platforms.

Jesús David Romero, Estudiante de Ingeniería de Sistemas

Resumen

Este trabajo hace un estado de arte sobre los avances en el software libre, centrándose en las herramientas que contribuyen a la accesibilidad de las plataformas tecnológicas.

calización geográfica y capacidades de los usuarios".

Software Libre

" Enseñar a los niños el uso del software libre en las escuelas, formará individuos con sentido de libertad. "

Richard Stallman

Palabras clave: Software libre, accesibilidad, plataformas tecnológicas, avance, discapacidad.

Introducción

La accesibilidad es una tendencia que pretende brindar a las personas que presenten algún tipo de discapacidad, ya sea visual, auditiva o cognitiva una experiencia parcialmente completa a los contenidos audiovisuales presentes en las diferentes plataformas tecnológicas, aliado a la ideología emanada por el software libre las oportunidades de avance que hay en este campo son tan limitadas como la imaginación de los colaboradores, y la oportunidad de generar cada vez mayor inmersión en estos contenidos creando una equidad en el mundo de las tecnologías.

Accesibilidad en las tecnologías de la información

La accesibilidad se basa en el diseño de producto y entornos de fácil uso para el mayor número de personas, enfocándose a la población más vulnerable en cuanto al uso de herramientas se refiere. Existen siete principios que apuntan al tema de la accesibilidad: la igualdad de uso, la flexibilidad, diseño simple e intuitivo, información perceptible, diseño tolerante a errores, diseño con escaso esfuerzo físico y tamaño y espacio para el acceso y uso.

Específicamente la WGAC tiene tres principios: Debe ser perceptible, operable, robusto.

La accesibilidad recoge varias clases de discapacidades: visual, cognitiva, auditiva y neurológica. Recoge también a personas que no poseen ningún tipo de discapacidad pero que por alguna condición no pueden disfrutar por completo su experiencia en la web (por ejemplo, la conexión lenta, computadores de bajos recursos, pérdida momentánea de un brazo, edad avanzada).

La W3C define accesibilidad como: *Acceso de todos a la Web, independientemente del tipo de hardware, software, infraestructura de red, idioma, cultura, lo-*

El software libre es el software que respeta la libertad de los usuarios y de la comunidad, en otras palabras es el derecho y la libertad que tienen los usuarios de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software; entendiendo así el software libre como una cuestión de libertad y no de precio ya que dada su escritura en ingles da para equivocaciones y malas interpretaciones.

Se dice que un software es libre cuando cumple con las siguientes 4 libertades:

- (Libertad 0) La libertad de ejecutar el programa como se desea, con cualquier propósito.
- (Libertad 1) La libertad de estudiar cómo funciona el programa, y cambiarlo para que haga lo que usted quiera. El acceso al código fuente es una condición necesaria para ello.
- (Libertad 2) La libertad de redistribuir copias para ayudar a su prójimo.
- (Libertad 3) La libertad de distribuir copias de sus versiones modificadas a terceros. Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello.

Ventajas del Software libre

Cuando se habla de libertad en el software libre y no se tiene profundidad en el tema, se puede quedar en el campo de la filosofía abstracta y poco práctica, por eso a continuación se esbozan algunas ventajas identificadas en el campo de la implementación de esta praxis filosófica y revolucionaria.

- **Revisión pública:** Al estar varias personas revisando sin ningún tipo de restricción el código, la detección de errores y bugs va a ser mucho más rápida.
- **Distribución de conocimientos:** Aunque se es enfático en que software libre no significa software gratuito, si hay casos en que efectivamente este no presenta precio para adquirirlo, permitiendo que la adquisición de conocimientos sea mucho más dinámica y más nutrida.
- **Innovación:** En el libro *La catedral y el bazar* de Erick Raymond, se muestra la diferencia entre un modelo de construcción a modo catedral, hermético y vertical y un modelo bazar, variado horizontal y “bullicioso” haciendo analogía con el software privativo y el software libre respectivamente, este modelo bazar donde hay un caos constante y un ir y venir de ideas e iniciativas permite efectivamente que de todo este caos surgen constantemente mejoras y nuevos desarrollos al software inicial haciendo que la innovación esté al orden del día.
- **Beneficio económico:** Como ya se mencionó antes en efecto mucho software libre es totalmente gratuito, permitiendo que en empresas la inversión en informática no sea tan alta.
- **Seguridad:** Esta ventaja va de la mano de la revisión pública pero se hace el énfasis ya que con la cantidad de datos que se manejan en la web la seguridad debe ser un indicador primordial.
- **Mozilla Firefox,** es un conocido proyecto de software libre que tiene una gran cantidad de extensiones para proporcionar accesibilidad en el contenido web.
- **Brity,** es un proceso que se ejecuta en segundo plano y permite la entrada de teclados braille para usarlos en consolas de texto de Linux y Unix.
- **Festival,** es un sintetizador de voces, disponible en inglés y en español para diferentes distribuciones. Con documentación completa para la implementación de nuevas voces.
- **Gnome-Speech,** es una librería API general que permite la programación de software basado en Gnome para producir voz, actualmente solo está disponible la interfaz de Festival.
- **Gnopernicus,** brinda a los usuarios con discapacidad visual o visión limitada usar las aplicaciones en Gnome mediante herramientas como lupas magnificadoras, sintetizador de voz y lector de braille.
- **Screader,** es un sintetizador de voz que reproduce texto y caracteres de la consola.
- **SVGATextMode,** ajusta el tamaño de las líneas de texto en consola en tarjetas SVGA para Linux en modo texto. Modifica el tamaño de la fuente, el cursor, la sincronización de horizontal y vertical, etc. En modo texto se puede sacar todo el partido a la tarjeta de video y del monitor.
- **KeyTouch,** permite generar funciones extras en el teclado para ciertas operaciones, aunque el proyecto no va generado principalmente a personas con discapacidad, si se muestra un gran interés por esta población.

Proyectos de Software libre y accesibilidad

Discapacidad visual

- El proyecto Orca, para el sistema operativo Linux, combina herramientas de síntesis de voz para que el ordenador lea en voz alta lo que aparece en la pantalla, con la posibilidad de trabajar con Braille y magnificación de pantalla. Es parte de la plataforma “Gnome” de Linux. <http://www.gnome.org/projects/orca>
- Lazaroux, este proyecto va enfocado a personas con discapacidad visual de habla hispana, es una distribución de Linux que ofrece varias herramientas y aplicaciones para facilitar la accesibilidad y un motor totalmente en castellano. No es necesaria su instalación para su utilización.

Discapacidad Motriz

- **Dasher,** es una interfaz de texto que permite mediante acciones predictivas del ratón reemplazar la escritura mediante el teclado; utiliza inteligencia artificial y estadísticas de grupos de palabras e idiomas.
- **XVoice,** permite el control total de las aplicaciones mediante la voz utilizando IBM’s ViaVoice for Linux, permite desde controlar programas hasta comandos de usuarios.

Herramientas de educación

- Clic, es un conocido software educativo que permite realizar diversas actividades: puzzles, asociaciones, crucigramas, sopas de letras, etc. Posee varias oportunidades de personalización. Jclic es la última versión, desarrollado como sw libre y disponible para MAC OS, Solaris, Windows y Linux, algunas características de click no estaban presentes en jclick pero gracias a su naturaleza de sw libre, varios grupos de colaboradores ya están incluyéndolas.

8. Screader leb.net
9. SVGATextMode freshmeat.net
10. KeyTouch keytouch.sourceforge.net
11. Dasher www.inference.phy.cam.ac.uk
12. XVoyce xvoice.sourceforge.net
13. Proyecto Clic clic.xtec.es
14. Britty mielke.cc
15. Proyecto orca www.gnome.org

Conclusiones

La accesibilidad es una tendencia que busca la equidad y el disfrute igualitario de los usuarios frente al contenido informático, dado su auge se vuelve necesario (no por capricho) la elaboración de múltiples herramientas que respondan a estas necesidades, aparte de la necesidad de su desarrollo se requiere la democratización de este conocimiento ya que es un derecho, el derecho a informarse y la falta de estrategias para difundir este conocimiento no puede ser la excusa. Es aquí donde entra el sw libre permitiendo mediante el libre y constante discernimiento la creación de nuevas y más y mejores herramientas que proporcionen al usuario final ese derecho al disfrute total de la información. Esto lo demuestra el hecho de que a pesar de que a esta iniciativa se sumó el sw libre un poco tarde ya ha hecho avances bastante significativos, que hacen pensar que en un futuro no muy lejano llegará a superar a los sw privativos, tanto en funcionalidad como en ética.

Jesús David Romero

Estudiante de Ingeniería de Sistemas, miembro activo del Grupo GNU Linux Universidad Distrital

Referencias

Bibliografía

Pautas, métodos y herramientas de evaluación de accesibilidad web Autor: Cinthia De Oleo Moreta, Luis Rodríguez Baena. Revista Universidad de Manizales. 21 de junio del 2013.

Cibergrafía

1. Software libre y accesibilidad: Matías Sánchez Caballero. 15 de julio del 2010. Tomado de www.nosolousabilidad.com
2. El sistema operativo GNU, ¿Qué es el Software Libre? www.gnu.org
3. Qué es la accesibilidad web www.w3c.es
4. Mozilla Firefox, accesibilidad web www.mozilla.org
5. Festival www.cstr.ed.ac.uk
6. GnomeSpeech www.escomposlinux.org
7. Gnopernicus www.escomposlinux.org