

# Diseño e Implementación de un Prototipo de Meta Distribución del Sistema Operativo Linux Bajo Licencia GPL Orientada al Servicio AAA (RADIUS) Integrando un Módulo de Administración WEB

Xavier Alexander Calderón Hinojosa, Evelyn Andrea Calderón Suasnavas, Mario David López Ramírez, Escuela Politécnica Nacional (EPN), Quito - Ecuador

**Resumen** – Este documento describe el diseño y creación de un prototipo de Meta Distribución Linux que facilita la implementación de servicios de autenticación, autorización y registro en una red de datos, a través de una herramienta de administración Web para FreeRADIUS desarrollada en PHP y AJAX. El documento incluye una descripción de la arquitectura de los servicios AAA (Autenticación, Autorización y Auditoría), el funcionamiento del protocolo RADIUS y generalidades del proyecto Metadistros Linux. Se detalla también las funcionalidades de la herramienta de administración web desarrollada, “SAAAS!”, los resultados de las pruebas realizadas con el servidor en un ambiente de laboratorio sencillo y las conclusiones y recomendaciones que se desprenden de este proyecto.

**Índices** – Seguridad en Redes, Servicios AAA, RADIUS, Distribuciones Linux, Desarrollo de Software.

## I. INTRODUCCIÓN

En la actualidad, el uso de esquemas de seguridad 802.1x y servicios AAA en redes de área local es de vital importancia. Por esta razón los administradores de red se ven en la necesidad de buscar herramientas que permitan una implementación sencilla y eficaz de estos servicios. El protocolo RADIUS es el protocolo AAA más utilizado, por lo que se ha decidido crear una solución de software libre basada en Linux que simplifique su implementación.

El Servidor de Autenticación, Autorización y Auditoría Simplificado “SAAAS!”, es una herramienta para redes 802.1X que facilita la administración y configuración de un servidor de autenticación que maneja el protocolo RADIUS. El prototipo se encuentra enfocado al personal encargado de redes de pequeña y mediana escala con poca experiencia en administración de servidores Linux pero familiarizado

---

X. Calderón, es Profesor Principal en la Facultad de Ingeniería Eléctrica y Electrónica de La Escuela Politécnica Nacional, Quito-Ecuador, (e-mail: xavier.calderon@epn.edu.ec).

E. Calderón, es Ingeniera Graduada en Electrónica y Redes de Información de La Escuela Politécnica Nacional, Quito-Ecuador, (e-mail: eve29online@hotmail.com)

M. López, es Ingeniero Graduado en Electrónica y Redes de Información de La Escuela Politécnica Nacional, Quito-Ecuador, (e-mail: md\_nano@hotmail.es)

con el estándar 802.1x y el funcionamiento del protocolo RADIUS.

## II. SERVICIOS AAA

El acrónimo AAA proviene de los términos en inglés Authentication (Autenticación), Authorization (Autorización), y Accounting (Registro). Durante el proceso de Autenticación en una red de datos, el dispositivo o usuario que solicita acceso al sistema de información debe responder a la pregunta: “¿Quién o qué es?”. Una vez terminado este proceso se inicia la etapa de Autorización en la que se debe responder a la pregunta “¿Qué se le permite hacer?”. Finalmente, se lleva a cabo el proceso de Registro o Auditoría, en el que se responde a la pregunta “¿Qué hizo?”.

### A. Arquitectura AAA

La arquitectura AAA ofrece un marco de gestión común para autenticar, autorizar y medir el consumo de recursos durante la conexión de un usuario, mediante la monitorización y generación de informes.

#### 1) Usuario

El usuario es el dispositivo que requiere ingresar a la red. Un usuario puede ser un computador personal o dispositivos móviles como una laptop o un PDA, un cliente VPN, etc.

#### 2) Punto de Ejecución de Políticas (PEP)

Es la entidad lógica responsable de aplicar los términos del acceso de un usuario al sistema de información. El dispositivo que realiza esta tarea dentro de una red de datos es el NAS, Network Access Server, que puede ser un Access Point, un conmutador, un firewall, un router o un concentrador VPN.

#### 3) Punto de Información de Políticas (PIP)

Es el repositorio de la información que puede ser usado durante la evaluación de una solicitud de acceso a la red. El PIP podría ser una base de datos de identificadores de usuarios, un Active Directory de Windows o un servidor LDAP.

#### 4) Punto de Decisión de Políticas (PDP)

Es la entidad lógica encargada de la toma de decisiones dentro de la arquitectura AAA. El PDP recoge las solicitudes de acceso del usuario a través del PEP y la información requerida para tomar la decisión de acceso del PIP y es la entidad que toma la decisión final de la admisión o

denegación del acceso a la red y los recursos de la misma.

### 5) Sistema de Registro y Auditoría

El Sistema de Registro y Auditoría almacena la información referente a la actividad que tienen los usuarios durante los procesos de autenticación y autorización y el uso de recursos dentro de la red.

### B. Funcionamiento de la Arquitectura AAA

El funcionamiento de la Arquitectura AAA se detalla en la Fig. 1.

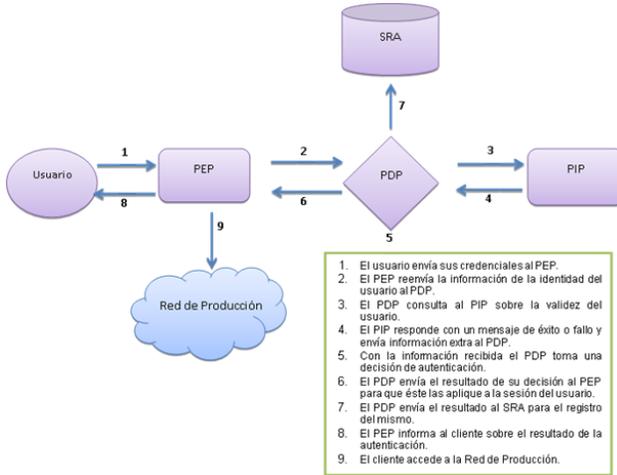


Fig. 1. Funcionamiento de la Arquitectura AAA

Cuando un usuario intenta conectarse a una red protegida mediante el servicio AAA, el dispositivo PEP le solicita información que lo identifique. El PEP envía la información de la identidad del usuario al PDP, el cual realiza una consulta a la PIP configurada para determinar si la información que envió el cliente es válida. La PIP responde con un mensaje de éxito o fracaso de la validación de credenciales y envía información adicional del cliente al PDP.

El PDP evalúa la información sobre el usuario obtenida a través del PEP y el PIP y toma una decisión de autenticación para crear un perfil de autorización para el usuario. Este resultado se envía al Sistema de Registro y Auditoría y al PEP.

En base a la información recibida, el PEP envía un mensaje de autenticación exitosa o fallida al usuario y pone en práctica el perfil de autorización del mismo.

### III. EL PROTOCOLO RADIUS

RADIUS, Remote Authentication Dial-In User Server/Servicio de Autenticación Remota para Usuario de Acceso Telefónico, es un protocolo estándar que se utiliza para proporcionar servicios de autenticación, autorización y registro para el acceso y uso de recursos de red. Actualmente, el funcionamiento del protocolo para los procesos de autenticación y autorización se describe en el RFC 2865, y los procesos para registro o auditoría se encuentran descritos en el RFC 2866.

### A. Formato del Paquete

El paquete RADIUS es de longitud variable, consta de cinco campos y se encapsula dentro de UDP. La Fig. 2 muestra el formato del paquete.



Fig. 2. Formato del Paquete RADIUS

#### 1) Código:

Es un octeto que identifica el tipo de paquete RADIUS que se transmite.

#### 2) Identificador:

Es un octeto que numera al paquete RADIUS para ayudar al servidor a emparejar peticiones con respuestas.

#### 3) Longitud:

Son dos octetos que indican la longitud total del paquete, incluyendo el campo Atributos. La longitud mínima de un paquete RADIUS es 20 octetos y la máxima es 4096 octetos.

#### 4) Autenticador:

Son 16 octetos que contienen la información que el cliente y el servidor RADIUS utilizan para autenticarse mutuamente.

#### 5) Atributos:

Este campo es de longitud variable y contiene la información necesaria para los procesos de autenticación, autorización y ciertos detalles de configuración.

### IV. META DISTRIBUCIONES LINUX

#### A. REMASTERSYS

REMASTERSYS es una herramienta que puede utilizarse para crear un LIVE CD de un respaldo completo de una distribución Debian que pueda instalarse en otro equipo; o se puede crear una copia distribible y personalizada de Debian siguiendo el concepto de Meta Distribuciones.

##### 1) InitransFS:

Es un sistema de archivos raíz mínimo embebido en el kernel de Linux desde la versión 2.7, que se utiliza para montar un pequeño sistema Linux temporal sobre la memoria RAM para hacer detecciones de hardware básicas antes del arranque del proceso init. REMASTERSYS utiliza initransFS para el arranque LIVE de la distribución que inicia desde el CD.

##### 2) Isolinux:

Es el nombre de un gestor de arranque para Linux/i386 que funciona sobre ISO 9660/El Torito CD-ROM permitiendo crear CDs autoejecutables.

##### 3) UnionFS y AuFS:

Son sistemas de archivos de Linux que permiten la unión del contenido de varios directorios aunque el contenido esté separado físicamente (sistemas de archivos diferentes). REMASTERSYS los usar para superponer un sistema de

archivos modificable sobre un CD que es un medio de solo lectura, de tal forma que el usuario lo vea como un sistema de archivos en el que es posible modificar, eliminar y crear contenido.

#### 4) SquashFS:

Es un sistema de archivos de sólo lectura para Linux que comprime directorios o sistemas de archivos completos. REMASTERSYS usa la imagen SquashFS para que la copia de una distribución Linux en producción quepa en un CD

### V. LA APLICACIÓN WEB “SAAAS!”

El Servidor de Autenticación, Autorización y Auditoría Simplificado “SAAAS!”, es una consola Web de administración y configuración para un servidor FreeRADIUS anidado en Debian Linux.

#### A. Requerimientos de la Solución

##### 1) Sistema Operativo:

“SAAAS!” ha sido diseñado para correr sobre un servidor DEBIAN GNU/Linux, por lo que no funcionará de manera correcta si se intenta anidarlos en otro sistema operativo.

##### 2) FreeRADIUS:

Es un paquete de código abierto distribuido bajo licencia GNU GPL usado para la implementación del protocolo RADIUS. “SAAAS!” se desarrolló para trabajar con la versión 2.1.8 de FreeRADIUS. Para habilitar todas las funciones de “SAAAS!” se debe compilar de manera especial los paquetes de FreeRADIUS disponibles para Debian.

##### 3) Otros paquetes:

Para el correcto funcionamiento de la aplicación “SAAAS!”, los paquetes OpenSSL, MySQL, Samba, Kerberos, Cron, Apache, PHP5 y CURL, deben estar instalados.

#### B. Consideraciones del Diseño.

“SAAAS!” aplica en su diseño los conceptos de la Web 2.0 a través del uso de AJAX (Javascript Asíncrono + XML). El esquema base utilizado en el código de “SAAAS!” se muestra en la Fig. 3 y la Fig. 4 muestra una comparación entre el modelo tradicional de una aplicación Web y el propuesto por AJAX.

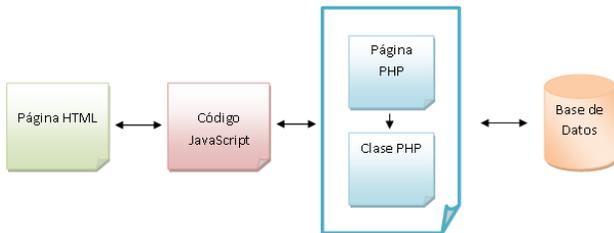


Fig. 3 Lógica de la Aplicación Web

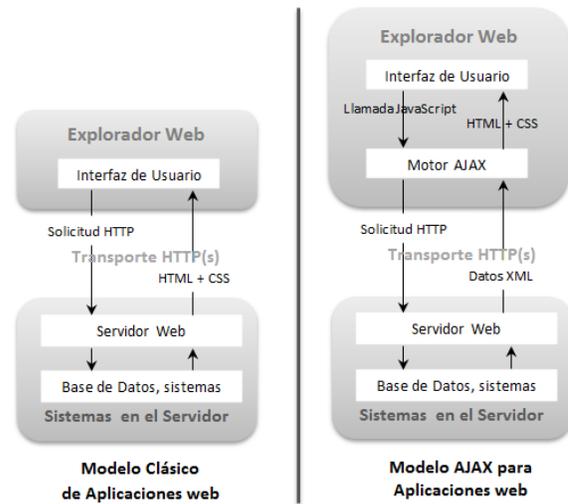


Fig. 4 El modelo tradicional de aplicación Web VS. El modelo propuesto por AJAX

#### C. Secciones de la Aplicación Web.

La aplicación Web “SAAAS!” está dividida en 6 secciones:

##### 1) Configuración Básica del Servidor:

En esta sección se detalla la información básica del servidor (nombre de host, dirección IP, estado de la memoria, etc.) y también permite realizar la configuración de las interfaces de red (dirección IP, gateway), el nombre del equipo, el DNS, etc.

##### 2) Administración de Certificados:

Ciertos protocolos de autenticación necesitan de la generación de certificados digitales para funcionar. La sección Administración de Certificados del servidor “SAAAS!” permite la creación de una estructura de certificados local para la creación de certificados auto firmados y la configuración de los protocolos utilizados en la etapa de autenticación del servicio AAA.

##### 3) Configuración del Servidor:

Esta sección permite la configuración de clientes NAS del Servidor RADIUS y administrar el estado de los servicios necesarios para que la aplicación “SAAAS!” funcione correctamente.

##### 4) Credenciales:

En esta sección se visualiza y se manipula la configuración de los repositorios de credenciales. Permite la configuración de una base de datos interna, Active Directory y un servidor LDAP externos como repositorios de credenciales.

##### 5) Políticas de Acceso:

Esta sección permite la configuración de políticas de acceso basadas en grupos.

##### 6) Visor de Eventos:

En esta sección se visualizan reportes del funcionamiento del servidor y de los procesos de autenticación y autorización de los usuarios, también se permite la creación de copias de seguridad para la recuperación de fallas mediante la restauración del sistema.

## VI. LA META DISTRIBUCIÓN “SAAAS!”

El Servidor AAA Simplificado “SAAAS!”, es un prototipo de Meta Distribución basada en Debian Squeeze, cuyo objetivo es facilitar la implementación de servicios AAA en entornos de red a través de la consola de administración de FreeRADIUS del mismo nombre. El prototipo ha sido optimizado para la implementación del servicio FreeRADIUS y el correcto funcionamiento de la aplicación Web “SAAAS!”

### A. Proyectos Similares.

La Tabla 1 muestra una comparación entre PHPRADmin, Cisco Secure ACS, y “SAAAS!”.

TABLA I  
COMPARACIÓN DE “SAAAS!” CON PROYECTOS SIMILARES

Característica	PhPRADmin	Cisco Secure ACS	SAAAS!
Tipo de Solución	Sistema Operativo basado en Fedora.	Sistema Operativo basado en Centos.	Sistema Operativo basado en Debian.
Licenciamiento	GNU/GPL	Propietario	GNU/GPL
Protocolos AAA	RADIUS	RADIUS y TACACS+	RADIUS
Manejo de Credenciales	Base de datos local y externa. Directorio LDAP.	Base de datos local. Directorio LDAP. Active Directory.	Base de datos local. Directorio LDAP. Active Directory.
Procesos de Autorización	Permite la configuración de políticas de acceso basada en usuarios.	Permite la configuración de políticas de accesos basada en reglas y condiciones.	Provee interfaz de administración amigable para la configuración de políticas de acceso.
Configuración de clientes NAS	En archivo de configuración.	En una base de datos.	En una base de datos.
Meta Distribución	NO	NO	SÍ

### B. Creación del Prototipo

El primer paso para la creación de la Meta Distribución es la instalación del Sistema Operativo base, en este caso Debian Squeeze. Ya que el objetivo de “SAAAS!” es funcionar como un servidor AAA administrado en su totalidad a través de un interfaz Web, se omite el sistema de ventanas X y GNOME en la instalación de la distribución base

#### 1) Instalación de FreeRADIUS:

El paquete de FreeRADIUS disponible para Debian no soporta los protocolos de autenticación EAP-TLS, EAP-TTLS y EAP-PEAP debido a los problemas de licenciamiento con OpenSSL. Es necesario modificar el archivo `..debian/rules` del código fuente para arreglar este inconveniente.

#### 2) Adaptación del GRUB2:

Se modificó el paquete GRUB2 para que muestre el logotipo de “SAAAS!” durante el arranque del sistema, en lugar del tema por defecto de Debian Linux, lo cual se logró a

través del uso de temas disponibles sólo en la versión 2 de GRUB.

#### 3) Adaptación de ISOLINUX:

Se modificaron los archivos de configuración de ISOLINUX, el gestor de arranque del LiveCD para personalizar el arranque “live” de la Meta Distribución “SAAAS”. La configuración de este menú se encuentra en el archivo `menu.cfg` el cual llama al archivo `live.cfg` donde se configura la acción que se realiza al escoger los diferentes ítems del menú, y al archivo `stdmenu.cfg` en el que se maneja el entorno de visualización, los colores e imagen de fondo que se muestra al arrancar el CD.

#### 4) Adaptación de REMASTERSYS:

Para la creación de la Meta Distribución se modificó los scripts del paquete REMASTERSYS. El script `/usr/bin/remastersys` es el encargado de crear la imagen `squashfs` del sistema operativo y convertirla en un `.iso`. Fue necesario comentar las líneas del script que borraban el archivo `/etc/hostname` y las claves generadas en el servidor para la conexión mediante SSH, opciones que generaban errores al momento de cargar el LiveCD.

El script `/usr/bin/remastersys-installer` se ejecuta desde el LiveCD y es el encargado de la instalación del sistema operativo en el disco duro. Para personalizar la instalación de “SAAAS!” se modificó este script y se lo renombró de tal forma que al ejecutar el comando `setup` desde el LiveCD el usuario tiene acceso al diálogo de instalación de la Meta Distribución.

## VII. PRUEBAS DE LABORATORIO

Para las pruebas de funcionamiento del servidor “SAAAS!” se utilizó la topología ilustrada en la Fig. 5. La tabla II detalla el rol de cada dispositivo en la topología de acuerdo a la arquitectura AAA.

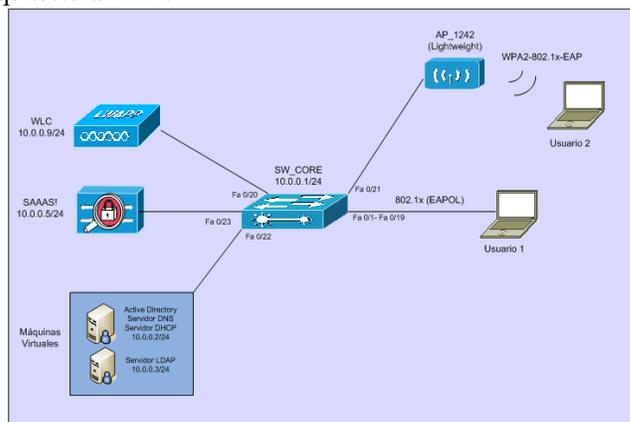


Fig. 4 Topología de red utilizada durante las pruebas.

TABLA II  
DESCRIPCIÓN DE LA TOPOLOGÍA DE RED PARA PRUEBAS DE LABORATORIO

Característica	Descripción	Rol AAA
Servidor “SAAAS!”	Provee el servicio de FreeRADIUS y la interfaz de administración “SAAAS!”.	PDP, PIP y Sistema de Registro y Auditoría

<b>Servidor Windows 2003</b>	Provee los servicios de Active Directory, DNS y DHCP para la red de pruebas.	PIP
<b>Servidor LDAP</b>	Servidor Debian que provee el servicio LDAP embebido en Zimbra.	PIP
<b>WLC</b>	Controladora Inalámbrica que actúa como cliente RADIUS	PDP
<b>Switch de Core</b>	Switch capa 3 que realiza el enrutamiento entre VLANs y funciona como cliente RADIUS	PDP

#### A. Escenario I: Esquema AAA utilizando la Base de Datos Local de "SAAAS!" como repositorio de credenciales.

En este escenario se implementa un esquema AAA básico en los que interviene un usuario (computador portátil), un cliente (switch de acceso) y un servidor AAA (SAAAS!). El usuario se registra en la red LAN por medio de 802.1x y después de la autenticación se aplican políticas de acceso basadas en su perfil de usuario. Las credenciales que utilizan son almacenadas localmente en una base de datos MySQL del servidor "SAAAS!".

#### B. Escenario II: Esquema AAA utilizando un repositorio de credenciales externo.

En este escenario se implementa un esquema AAA básico en los que interviene un usuario (computador portátil), un cliente (switch de acceso), un servidor de Active Directory (Windows 2003) y un servidor AAA (SAAAS!). En este escenario el usuario se registra en la red LAN por medio de 802.1x, las credenciales del usuario residen en un Active Directory.

### VIII. CONCLUSIONES

El resultado final de este proyecto es el Servidor de Autenticación, Autorización y Auditoría Simplificado "SAAAS!", un prototipo de Meta Distribución Linux basada en Debian Squeeze que permite la implementación de servicios AAA mediante RADIUS en redes de datos de mediana escala que requieren soluciones avanzadas de seguridad de la red con bajos costos de implementación.

Para el desarrollo de este proyecto se utilizó el protocolo RADIUS como base de la arquitectura AAA. La decisión de utilizar este protocolo se basó en la alta compatibilidad de los equipos de red actuales con el mismo y en la existencia de la herramienta FreeRADIUS, que es un conjunto de paquetes estable desarrollado para plataformas Linux/Unix.

Se considera a "SAAAS!" una Meta Distribución por ser un sistema operativo basado en Debian Linux, personalizado para entornos AAA y cuyo proceso de arranque tiene lugar desde un CD-ROM (modo LIVE), si el usuario decide instalar "SAAAS!", esta instalación mantendrá todas configuraciones realizadas en el modo Live (cambios en la configuración de FreeRADIUS, instalación de nuevos paquetes, modificaciones en la base de datos). Además, se mantienen las herramientas necesarias para poder generar otra distribución a partir de esta instalación.

Para el desarrollo del módulo de administración Web "SAAAS!" se usó AJAX (JavaScript Asíncrono + XML), para que la aplicación sea más interactiva y amigable para el usuario siguiendo la tendencia de la Web 2.0; ya que permite que una página Web solicite y reciba información del servidor en segundo plano de manera asíncrona sin interferir con la presentación de la página.

El módulo de administración Web "SAAAS!" provee una solución rápida y eficaz para la implementación de servicios AAA en una red de datos al permitir un manejo flexible de repositorios externos y locales de credenciales para la autenticación de usuarios en la red. La aplicación Web ha sido probada con una base de datos local MySQL, un servidor Active Directory sobre Windows 2003 y un servidor OpenLDAP.

Al momento de integrar soluciones de código abierto con servicios propietarios se debe tomar en consideración protocolos comunes y desarrollos de integración parcial. En el caso de este proyecto para la integración con Active Directory (Microsoft) se tuvo que ocupar dos herramientas: el servicio LDAP y el paquete NTLM controlado por el servicio Samba, que en conjunto permiten tener una gran compatibilidad con los atributos y métodos de autenticación de Active Directory.

El servidor "SAAAS!" almacena los eventos de autenticación y autorización de los usuarios o dispositivos que han intentado acceder a la red y los muestra a través de la aplicación Web, permitiendo que el administrador del servicio lleve a cabo un seguimiento que más tarde puede utilizarse en la administración de los recursos de red.

Se puede implementar "SAAAS!" en redes pequeñas y medianas en las que se requiere limitar el acceso de los usuarios a información confidencial o a ciertos servicios de red, como Cooperativas de Ahorro en las que el acceso a herramientas financieras debe ser limitado para pocos usuarios, o clínicas pequeñas, en las que sólo los médicos tratantes deben tener acceso al historial de los pacientes.

### REFERENCIAS

- [1] Anvin, Peter. Syslinux. *Comboot/menu.c32*. [En línea] 27 de Enero de 2011. [Citado el: 20 de Junio de 2011.] <http://syslinux.zytor.com/wiki/index.php/Comboot/menu.c32>.
- [2] Bennett, Colin. *GNU GRUB Graphical Menu*. [En línea] 29 de Noviembre de 2009. [Citado el: 21 de Enero de 2011.] <http://grub.gibibit.com/Journal>.
- [3] Charles P. Wright, Erez Zadok. *Linux Journal. Kernel Kerner-Unionfs:Bringing Filesystems together*. [En línea] 1 de Diciembre de 2004. [Citado el: 4 de Junio de 2011.] <http://www.linuxjournal.com/article/7714?page=0,0>.
- [4] *¿Cómo hacer una metadistro?* [En línea] 11 de Diciembre de 2007. [Citado el: 2 de Junio de 2011.] <http://tuxplc-linuxgroup.blogspot.com/2007/12/cmo-hacer-una-metadistro.html>.
- [5] Comunidad de Usuarios Guadalinex. *Initransfs:¿Qué es?* [En línea] 12 de Julio de 2010. [Citado el: 4 de Junio de 2011.] [http://forja.guadalinex.org/webs/guadalinexv5/doku.php?id=intra\\_mfs:que\\_es](http://forja.guadalinex.org/webs/guadalinexv5/doku.php?id=intra_mfs:que_es).
- [6] Garrett, Jesse James, *Ajax: A New Approach to Web Applications*. 2005, <http://www.adaptivepath.com>
- [7] González, Sergio Gonzáles. *Creando un CDROM autoejecutable con ISOLINUX*. [En línea] [Citado el: 04 de 06 de 2011.] <http://linuxupc.upc.es/~kampa/documentos/isolinux/index.html>.

- [8] ICS Yaco, S.L y Emergya S. Coop. And. El Proyecto Guadalínx. *Infraestructura del Proyecto de Software Libre impulsado por la Junta de Andalucía*. [En línea] 23 de Enero de 2004. [Citado el: 5 de Marzo de 2011.] <http://speeches.ofset.org/jrfernandez/rml2004/fuentes/proyguadalínx.pdf>.
- [9] Pavlov, Artemiy I. *SquashFS HOWTO*. [En línea] 24 de Julio de 2008. [Citado el: 4 de Junio de 2011.] <http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/SquashFS-HOWTO.pdf>.
- [10] Rigney, C., y otros, y otros. *Remote Authentication Dial In User Service (RADIUS)*. RFC 2865. Junio de 2000.
- [11] Rigney, C. *RADIUS Accounting*. RFC 2866. Junio de 2000
- [12] Tirado, Rafael Martín de Agar. CALIU:Asociación de Usuario de GNU/Linux en Lengua Catalana. *Analizando en profundidad el proyecto METADISTROS*. [En línea] 8 de Septiembre de 2003. [Citado el: 1 de Junio de 2011.] [http://ftp.caliu.cat/pub/caliu/distros/doc/sistemas\\_dw\\_0.3.0.pdf](http://ftp.caliu.cat/pub/caliu/distros/doc/sistemas_dw_0.3.0.pdf).
- [13] Vollbrecht, J., y otros, y otros. *AAA Authorization Framework*. RFC 2904. Agosto de 2000.



**Mario López**, nació el 14 de Febrero de 1988 en Quito-Ecuador, se graduó en el Colegio Técnico Aeronáutico de Aviación Civil, especialidad Físico-Matemático en el año 2005. En el 2011 obtiene el título de Ingeniero en Electrónica y Redes de Información en la Escuela Politécnica Nacional.. Actualmente trabaja como Jefe de Seguridad Informática en el Ministerio de Defensa del Ecuador.

(e-mail: [md\\_nano@hotmail.es](mailto:md_nano@hotmail.es))

## BIOGRAFIAS



**Xavier Calderón**, nació el 16 de Agosto de 1972 en Quito-Ecuador, se graduó en el Colegio La Salle, especialidad Físico-Matemático en el año 1990. En el 2011 obtiene el título de Ingeniero en Electrónica y Telecomunicaciones en la Escuela Politécnica Nacional y en el 2002 se gradúa como Máster en Tecnologías de la Información

en Fabricación en la Universidad Politécnica de Madrid. Actualmente trabaja en la Escuela Politécnica Nacional donde es profesor principal a tiempo completo, Miembro del Consejo de Departamento (Departamento de Electrónica y Redes de Información). Jefe del Laboratorio de Informática de la Facultad de Ingeniería Eléctrica y Electrónica y Director Proyecto Semilla.

(e-mail: [xavier.calderon@epn.edu.ec](mailto:xavier.calderon@epn.edu.ec))



**Evelyn Calderón**, nació el 29 de Agosto de 1987 en Quito-Ecuador, se graduó en la Unidad Educativa Experimental Manuela Cañizares, especialidad Físico-Matemático en el año 2005. En el 2011 obtiene el título de Ingeniera en Electrónica y Redes de Información en la Escuela Politécnica Nacional. Actualmente

trabaja como Ingeniera Preventa en la empresa ANDEAN-TRADE SA.

(e-mail: [eve29online@hotmail.com](mailto:eve29online@hotmail.com))