Universidad Luterana Salvadoreña

Por una educación sin fronteras

FACULTAD CIENCIAS DEL HOMBRE Y LA NATURALEZA
LICENCIATURA EN CIENCIAS DE LA COMPUTACION

Cátedra: ***Redes I***

Tema: ***"Configuración de servicios con IPv6"***

Catedrático: ***Ing. Manuel Flores Villatoro***

Presentado por:

No.	Apellidos	Nombres	Carnet	Porcentaje de participación
1	Hernández Peña	Isabel Guadalupe	HP02110790	100 %
2	Vásquez Cerón	Mauricio Edgardo	VC02110805	100 %

San Salvador, 18 de mayo de 2013

Contenido

1. Introducción

Hoy en día las tecnologías de comunicación han evolucionado a pasos agigantados, ya que en la actualiadad es posible obtener una conexión a internet desde casi cualquier dispositivo, como lo son los smartphones, tables, teléfonos móviles, etc., es por ello que hoy en día ya no se cuentan con ip's versión 4 (IPv4) disponibles para ser asignadas a los nuevos dispositivos, por esta razón es que nace la versión 6 de este protocolo.

En el presente documento se trataran aspectos fundamentales tales como que es IPv4 e IPv6, porqué surge la versión IPv6, cual es su finalidad y cual es la forma y/o los protocolos que deben seguirce para la implementación de esta nueva versión.

También mostraremos y detallaremos dos ejemplos de instalación y configuración de servicios que puedan responder a ambas versiones de IP (IPv4 e IPv6), los cuales son un servidor HTTP (servidor web) y un servidor FTP (para transferencia de archivos).

2. Objetivos

- ***Objetivo General***

  - *Configuración de dos servicios que puedan responder a sus respectivas direcciones IPv4 e IPv6.*

- ***Objetivos Específicos***

  - *Introducción a conceptos básicos sobre IPv4 e IPv6.*
  - *Instalación y configuración de dos servicios.*

3. IPv4

*El **Internet Protocol version 4 (IPv4)** (en español: Protocolo de Internet versión 4) es la cuarta versión del protocolo Internet Protocol (IP), y la primera en ser implementada a gran escala. Definida en el RFC 791.*

IPv4 usa direcciones de 32 bits, limitándola a 2^{32} = 4.294.967.296 direcciones únicas, muchas de las cuales están dedicadas a redes locales (LANs). Por el crecimiento enorme que ha tenido Internet (mucho más de lo que esperaba, cuando se diseñó IPv4), combinado con el hecho de que hay desperdicio de direcciones en muchos casos, ya hace varios años se vio que escaseaban las direcciones IPv4.

Esta limitación ayudó a estimular el impulso hacia IPv6, que está actualmente en las primeras fases de implantación, y se espera que termine reemplazando a IPv4.

Las direcciones disponibles en la reserva global de IANA pertenecientes al protocolo IPv4 se

agotaron el jueves 3 de febrero de 2011 oficialmente. Los Registros Regionales de Internet deben, desde ahora, manejarse con sus propias reservas, que se estima, alcanzaran hasta septiembre de 2011

Actualmente no quedan direcciones IPv4 disponibles para compra, por ende se está en la forzosa y prioritaria obligacion de migrar a IPv6, Los sistemas operativos Windows Vista, 7, 8, Unix/like (Gnu/linux, Unix, Mac OSX), BSD entre otros, tienen soporte nativo para IPv6, mientras que Windows XP requiere utilizar el prompt y digitar ipv6 install, para instalarlo, y sistemas anteriores no tienen soporte para este.

Desperdicio de direcciones

El desperdicio de direcciones IPv4 se debe a varios factores:

Uno de los principales es que inicialmente no se consideró el enorme crecimiento que iba a tener Internet; se asignaron bloques de direcciones grandes (de 16,271 millones de direcciones) a países, e incluso a empresas.

Otro motivo de desperdicio es que en la mayoría de las redes, exceptuando las más pequeñas, resulta conveniente dividir la red en subredes. Dentro de cada subred, la primera y la última dirección no son utilizables; de todos modos no siempre se utilizan todas las direcciones restantes. Por ejemplo, si en una subred se quieren acomodar 80 hosts, se necesita una subred de 128 direcciones (se tiene que redondear a la siguiente potencia de base 2); en este ejemplo, las 48 direcciones restantes ya no se utilizan.

4. IPv6

El Internet Protocol version 6 (IPv6) (en español: Protocolo de Internet versión 6) es una versión del protocolo Internet Protocol (IP), definida en el RFC 2460 y diseñada para reemplazar a Internet Protocol version 4 (IPv4) RFC 791, que actualmente está implementado en la gran mayoría de dispositivos que acceden a Internet.

Diseñado por Steve Deering de Xerox PARC y Craig Mudge, IPv6 sujeto a todas las normativas que fuera configurado está destinado a sustituir a IPv4, cuyo límite en el número de direcciones de

red admisibles está empezando a restringir el crecimiento de Internet y su uso, especialmente en China, India, y otros países asiáticos densamente poblados. El nuevo estándar mejorará el servicio globalmente; por ejemplo, proporcionará a futuras celdas telefónicas y dispositivos móviles sus direcciones propias y permanentes.

A principios de 2010, quedaban menos del 10% de IPs sin asignar.[1] *En la semana del 3 de febrero del 2011, la IANA (Agencia Internacional de Asignación de Números de Internet, por sus siglas en inglés) entregó el último bloque de direcciones disponibles (33 millones) a la organización encargada de asignar IPs en Asia, un mercado que está en auge y no tardará en consumirlas todas.*

IPv4 posibilita 4,294,967,296 (2^{32}) direcciones de host diferentes, un número inadecuado para dar una dirección a cada persona del planeta, y mucho menos a cada vehículo, teléfono, PDA, etcétera. En cambio, IPv6 admite 340.282.366.920.938.463.463.374.607.431.768.211.456 (2^{128} o 340 sextillones de direcciones) —cerca de 6,7 × 10^{17} (670 mil billones) de direcciones por cada milímetro cuadrado de la superficie de La Tierra.

Otra vía para la popularización del protocolo es la adopción de este por parte de instituciones. El gobierno de los Estados Unidos ordenó el despliegue de IPv6 por todas sus agencias federales en el año 2008

Motivación y orígenes de las IP

Durante la primera década de operación de Internet basado en TCP/IP, a fines de los 80s, se hizo aparente que se necesitaba desarrollar métodos para conservar el espacio de direcciones. A principios de los 90s, incluso después de la introducción del rediseño de redes sin clase, se hizo claro que no sería suficiente para prevenir el agotamiento de las direcciones IPv4 y que se necesitaban cambios adicionales. A comienzos de 1992, circulaban varias propuestas de sistemas y a finales de 1992, la IETF anunció el llamado para white papers (RFC 1550) y la creación de los grupos de trabajo de "IP de próxima generación" ("IP Next Generation") o (IPng).

IPng fue propuesto por el Internet Engineering Task Force (IETF) el 25 de julio de 1994, con la formación de varios grupos de trabajo IPng. Hasta 1996, se publicaron varios RFCs definiendo IPv6, empezando con el RFC 2460.

La discusión técnica, el desarrollo e introducción de IPv6 no fue sin controversia. Incluso el diseño ha sido criticado por la falta de interoperabilidad con IPv4 y otros aspectos, por ejemplo por el científico de computación D. J. Bernstein.[3]

Incidentalmente, IPng (IP Next Generation) no pudo usar la versión número 5 (IPv5) como sucesor

de IPv4, ya que ésta había sido asignada a un protocolo experimental orientado al flujo de streaming que intentaba soportar voz, video y audio.
Se espera ampliamente que IPv6 sea soportado en conjunto con IPv4 en el futuro cercano. Los nodos solo-IPv4 no son capaces de comunicarse directamente con los nodos IPv6, y necesitarán ayuda de un intermediario.

Cambios y nuevas características

En muchos aspectos, IPv6 es una extensión conservadora de IPv4. La mayoría de los protocolos de transporte y aplicación necesitan pocos o ningún cambio para operar sobre IPv6; las excepciones son los protocolos de aplicación que integran direcciones de capa de red, como FTP o NTPv3, NTPv4.
IPv6 especifica un nuevo formato de paquete, diseñado para minimizar el procesamiento del encabezado de paquetes. Debido a que las cabeceras de los paquetes IPv4 e IPv6 son significativamente distintas, los dos protocolos no son interoperables.

Autoconfiguración de direcciones libres de estado (SLAAC)

Los nodos IPv6 pueden configurarse a sí mismos automáticamente cuando son conectados a una red ruteada en IPv6 usando los mensajes de descubrimiento de routers de ICMPv6. La primera vez que son conectados a una red, el nodo envía una solicitud de router de link-local usando multicast (router solicitación) pidiendo los parámetros de configuración; y si los routers están configurados para esto, responderán este requerimiento con un "anuncio de router" (router advertisement) que contiene los parámetros de configuración de capa de red.
Si la autoconfiguración de direcciones libres de estado no es adecuada para una aplicación, es posible utilizar Dynamic Host Configuration Protocol para IPv6 (DHCPv6) o bien los nodos pueden ser configurados en forma estática.
Los routers presentan un caso especial de requerimientos para la configuración de direcciones, ya que muchas veces son la fuente para información de autoconfiguración, como anuncios de prefijos de red y anuncios de router. La configuración sin estado para routers se logra con un protocolo especial de renumeración de routers.

Multicast

Multicast, la habilidad de enviar un paquete único a destinos múltiples es parte de la especificación base de IPv6. Esto es diferente a IPv4, donde es opcional (aunque usualmente

implementado).

IPv6 no implementa broadcast, que es la habilidad de enviar un paquete a todos los nodos del enlace conectado. El mismo efecto puede lograrse enviando un paquete al grupo de multicast de enlace-local todos los nodos (all hosts). Por lo tanto, no existe el concepto de una dirección de broadcast y así la dirección más alta de la red (la dirección de broadcast en una red IPv4) es considerada una dirección normal en IPv6.

Muchos ambientes no tienen, sin embargo, configuradas sus redes para rutear paquetes multicast, por lo que en éstas será posible hacer "multicasting" en la red local, pero no necesariamente en forma global.

El multicast IPv6 comparte protocolos y características comunes con IPv4, pero también incorpora cambios y mejoras. Incluso cuando se le asigne a una organización el más pequeño de los prefijos de ruteo global IPv6, ésta también recibe la posibilidad de usar uno de los 4.2 billones de grupos multicast IPv6 ruteables de fuente específica para asignarlos para aplicaciones multicast intra-dominio o entre-dominios (RFC 3306). En IPv4 era muy difícil para una organización conseguir incluso un único grupo multicast ruteable entre-dominios y la implementación de las soluciones entre-dominios eran anticuadas (RFC 2908). IPv6 también soporta nuevas soluciones multicast, incluyendo Embedded Rendezvous Point (RFC 3956), el que simplifica el despliegue de soluciones entre dominios.

Seguridad de Nivel de Red obligatoria

Internet Protocol Security (IPsec), el protocolo para cifrado y autenticación IP forma parte integral del protocolo base en IPv6. El soporte IPsec es obligatorio en IPv6; a diferencia de IPv4, donde es opcional o fue un agregado posterior (pero usualmente implementado). Sin embargo, actualmente no se está usando normalmente IPsec excepto para asegurar el tráfico entre routers de BGP IPv6, aunque también se puede utilizar en OSPFv3 y en movilidad IPv6 (ver Movilidad IPv6)

Procesamiento simplificado en los routers

Se hicieron varias simplificaciones en la cabecera de los paquetes, así como en el proceso de reenvío de paquetes para hacer el procesamiento de los paquetes más simple y por ello más eficiente. En concreto,

- El encabezado del paquete en IPv6 es más simple que el utilizado en IPv4, así los campos que son raramente utilizados han sido movidos a opciones separadas; en efecto, aunque las direcciones en IPv6 son 4 veces más largas, el encabezado IPv6 (sin opciones) es solamente

el doble de largo que el encabezado IPv4 (sin opciones).

- *Los routers IPv6 no hacen fragmentación. Los nodos IPv6 requieren ya sea hacer descubrimiento de MTU, realizar fragmentación extremo a extremo o enviar paquetes menores al MTU mínimo de IPv6 de 1280 bytes.*
- *El encabezado IPv6 no está protegido por una suma de comprobación (checksum); la protección de integridad se asume asegurada tanto por el checksum de capa de enlace y por un checksum de nivel superior (TCP, UDP, etc.). En efecto, los routers IPv6 no necesitan recalcular la suma de comprobación cada vez que algún campo del encabezado (como el contador de saltos o Tiempo de Vida) cambian. Esta mejora puede ser menos necesaria en routers que utilizan hardware dedicado para computar este cálculo y así pueden hacerlo a velocidad de línea (wirespeed), pero es relevante para routers por software.*
- *El campo Tiempo de Vida de IPv4, conocido como TTL (Time To Live), pasa a llamarse Límite de saltos, reflejando el hecho de que ya no se espera que los routers computen el tiempo en segundos que tarda en atravesarlo (que en cualquier caso siempre resulta menor de 1 segundo). Se simplifica como el número de saltos entre routers que se permita realizar al paquete IPv6.*

Movilidad

A diferencia de IPv4 móvil (MIPv4), IPv6 móvil (MIPv6) evita el ruteo triangular y por lo tanto es tan eficiente como el IPv6 normal. Los routers IPv6 pueden soportar también Movilidad de Red (NEMO, por Network Mobility) (RFC 3963), que permite que redes enteras se muevan a nuevos puntos de conexión de routers sin reasignación de numeración. Sin embargo, ni MIPv6 ni MIPv4 o NEMO son ampliamente difundidos o utilizados hoy, por lo que esta ventaja es más bien teórica.

Soporte mejorado para las extensiones y opciones

Los cambios en la manera en que se codifican las opciones de la cabecera IP permiten límites menos rigurosos en la longitud de opciones, y mayor flexibilidad para introducir nuevas opciones en el futuro.

Jumbogramas

IPv4 limita los paquetes a 64 KiB de carga útil. IPv6 tiene soporte opcional para que los paquetes puedan superar este límite, los llamados jumbogramas, que pueden ser de hasta 4 GiB. El uso de jumbogramas puede mejorar mucho la eficiencia en redes de altos MTU. El uso de jumbogramas

está indicado en el encabezado opcional Jumbo Payload Option.

Direccionamiento IPv6

El cambio más grande de IPv4 a IPv6 es la longitud de las direcciones de red. Las direcciones IPv6, definidas en el RFC 2373 y RFC 2374 pero fue redefinida en abril de 2003 en la RFC 3513, son de 128 bits; esto corresponde a 32 dígitos hexadecimales, que se utilizan normalmente para escribir las direcciones IPv6, como se describe en la siguiente sección.

El número de direcciones IPv6 posibles es de $2^{128} \approx 3.4 \times 10^{38}$. Este número puede también representarse como 16^{32}, con 32 dígitos hexadecimales, cada uno de los cuales puede tomar 16 valores (véase combinatoria).

En muchas ocasiones las direcciones IPv6 están compuestas por dos partes lógicas: un prefijo de 64 bits y otra parte de 64 bits que corresponde al identificador de interfaz, que casi siempre se genera automáticamente a partir de la dirección MAC de la interfaz a la que está asignada la dirección.

Notación para las direcciones IPv6

Las direcciones IPv6, de 128 bits de longitud, se escriben como ocho grupos de cuatro dígitos hexadecimales. Por ejemplo,

```
2001:0db8:85a3:08d3:1319:8a2e:0370:7334
```

es una dirección IPv6 válida.

Se puede comprimir un grupo de cuatro dígitos si éste es nulo (es decir, toma el valor "0000"). Por ejemplo,

```
2001:0db8:85a3:0000:1319:8a2e:0370:7344
                ----
  2001:0db8:85a3::1319:8a2e:0370:7344
```

Siguiendo esta regla, si más de dos grupos consecutivos son nulos, también pueden comprimirse como "::". Si la dirección tiene más de una serie de grupos nulos consecutivos la compresión sólo se permite en uno de ellos. Así, las siguientes son representaciones posibles de una misma dirección:

```
2001:0DB8:0000:0000:0000:0000:1428:57ab
2001:0DB8:0000:0000:0000::1428:57ab
2001:0DB8:0:0:0:0:1428:57ab
```

2001:0DB8:0::0:1428:57ab
2001:0DB8::1428:57ab

son todas válidas y significan lo mismo, pero
2001::25de::cade
    --   --

no es válida porque no queda claro cuántos grupos nulos hay en cada lado.
Los ceros iniciales en un grupo también se pueden omitir:
2001:0DB8:02de::0e13
2001:DB8:2de::e13

Si la dirección es una dirección IPv4 empotrada, los últimos 32 bits pueden escribirse en base decimal, así:
::ffff:192.168.89.9
::ffff:c0a8:5909

No se debe confundir con:
::192.168.89.9
::c0a8:5909

El formato `::ffff:1.2.3.4` *se denomina dirección IPv4 mapeada, y el formato* `::1.2.3.4` *dirección IPv4 compatible.*
Las direcciones IPv4 pueden ser transformadas fácilmente al formato IPv6. Por ejemplo, si la dirección decimal IPv4 es 135.75.43.52 (en hexadecimal, 0x874B2B34), puede ser convertida a 0000:0000:0000:0000:0000:0000:874B:2B34 o ::874B:2B34. Entonces, uno puede usar la notación mixta dirección IPv4 compatible, en cuyo caso la dirección debería ser ::135.75.43.52. Este tipo de dirección IPv4 compatible casi no está siendo utilizada en la práctica, aunque los estándares no la han declarado obsoleta.
Cuando lo que se desea es identificar un rango de direcciones diferenciable por medio de los primeros bits, se añade este número de bits tras el carácter de barra "/". Por ejemplo:
2001:0DB8::1428:57AB/96 sería equivalente a 2001:0DB8::
2001:0DB8::874B:2B34/96 sería equivalente a 2001:0DB8:: y por supuesto también a 2001:0DB8::1428:57AB/96

Identificación de los tipos de direcciones

Los tipos de direcciones IPv6 pueden identificarse tomando en cuenta los rangos definidos por los primeros bits de cada dirección.

::/128

La dirección con todo ceros se utiliza para indicar la ausencia de dirección, y no se asigna ningún nodo.

::1/127

La dirección de loopback es una dirección que puede usar un nodo para enviarse paquetes a sí mismo (corresponde con 127.0.0.1 de IPv4). No puede asignarse a ninguna interfaz física.

::1.2.3.4/96

La dirección IPv4 compatible se usa como un mecanismo de transición en las redes duales IPv4/IPv6. Es un mecanismo que no se usa.

::ffff:0:0/96

La dirección IPv4 mapeada se usa como mecanismo de transición en terminales duales.

fe80::/10

El prefijo de enlace local (en inglés link local) específica que la dirección sólo es válida en el enlace físico local.

fec0::

El prefijo de emplazamiento local (en inglés site-local prefix) específica que la dirección sólo es válida dentro de una organización local. La RFC 3879 lo declaró obsoleto, estableciendo que los sistemas futuros no deben implementar ningún soporte para este tipo de dirección especial. Se deben sustituir por direcciones Local IPv6 Unicast.

ff00::/8

El prefijo de multicast. Se usa para las direcciones multicast.

Hay que resaltar que no existen las direcciones de difusión (en inglés broadcast) en IPv6, aunque la funcionalidad que prestan puede emularse utilizando la dirección multicast FF01::1/128,

denominada todos los nodos (en inglés all nodes)

Paquete IPv6

Un paquete en IPv6 está compuesto principalmente de dos partes: la cabecera (que tiene una parte fija y otra con las opciones) y la carga útil (los datos).
= ! 0 ! 0 | colspan="4"| Versión | colspan="8"| Clase de Tráfico | colspan="20"| Etiqueta de Flujo |- ! 4 ! 32 | colspan="16"| Longitud del campo de datos | colspan="8"| Cabecera Siguiente | colspan="8"| Límite de Saltos |- ! 8 ! 64 | colspan="32" rowspan="4"| Dirección de Origen |- ! C ! 96 |- ! 10 ! 128 |- ! 14 ! 160 |- ! 18 ! 192 | colspan="32" rowspan="4"| Dirección de Destino |- ! 1C ! 224 |- ! 20 ! 256 |- ! 24 ! 288 |}

Hay dos versiones de IPv6 levemente diferentes. La ahora obsoleta versión inicial, descrita en el RFC 1883, difiere de la actual versión propuesta de estándar, descrita en el RFC 2460, en dos campos: hay 4 bits que han sido reasignados desde "etiqueta de flujo" (flow label) a "clase de tráfico" (traffic class). El resto de diferencias son menores.
En IPv6 la fragmentación se realiza sólo en el nodo origen del paquete, al contrario que en IPv4 en donde los routers pueden fragmentar un paquete. En IPv6, las opciones también desaparecen de la cabecera estándar y son especificadas por el campo "Cabecera Siguiente" (Next Header), similar en funcionalidad en IPv4 al campo Protocolo. Un ejemplo: en IPv4 uno añadiría la opción "ruta fijada desde origen" (Strict Source and Record Routing) a la cabecera IPv4 si quiere forzar una cierta ruta para el paquete, pero en IPv6 uno modificaría el campo "Cabecera Siguiente" indicando que viene una cabecera de encaminamiento. La cabecera de encaminamiento podrá entonces especificar la información adicional de encaminamiento para el paquete, e indicar que, por ejemplo, la cabecera TCP será la siguiente. Este procedimiento es análogo al de AH y ESP en IPsec para IPv4 (que aplica a IPv6 de igual modo, por supuesto).

Cabeceras de extensión

El uso de un formato flexible de cabeceras de extensión opcionales es una idea innovadora que permite ir añadiendo funcionalidades de forma paulatina. Este diseño aporta gran eficacia y flexibilidad ya que se pueden definir en cualquier momento a medida que se vayan necesitando entre la cabecera fija y la carga útil.
Hasta el momento, existen 8 tipos de cabeceras de extensión, donde la cabecera fija y las de extensión opcionales incluyen el campo de cabecera siguiente que identifica el tipo de cabeceras de

extensión que viene a continuación o el identificador del protocolo de nivel superior. Luego las cabeceras de extensión se van encadenando utilizando el campo de cabecera siguiente que aparece tanto en la cabecera fija como en cada una de las citadas cabeceras de extensión. Como resultado de la secuencia anterior, dichas cabeceras de extensión se tienen que procesar en el mismo orden en el que aparecen en el datagrama. La Cabecera principal, tiene a diferencia de la cabecera de la versión IPv4 un tamaño fijo de 40 octetos.Específica para asignarlos para aplicaciones multicast intra-dominio o entre-dominios (RFC 3306). En IPv4 era muy difícil para una organización como ésta.

Mecanismos de transición a IPv6

Ante el agotamiento de las direcciones IPv4, y los problemas que este está ocasionando ya, sobre todo en los países emergentes de Asia como India o China, el cambio a IPv6 ya ha comenzado. Se espera que convivan ambos protocolos durante un año, aunque se piensa que la implantación mundial y total en internet de IPv6 se hará realidad hacia finales de 2012, dada la celeridad con la que se están agotando las direcciones IPv4. La red no podrá aguantar mucho más sin el cambio, y de no realizarse pronto este las consecuencias podrían ser muy graves. Existe una serie de mecanismos que permitirán la convivencia y la migración progresiva tanto de las redes como de los equipos de usuario. En general, los mecanismos de transición pueden clasificarse en tres grupos:

- *Doble pila*
- *Túneles*
- *Traducción*

La doble pila hace referencia a una solución de nivel IP con doble pila (RFC 4213), que implementa las pilas de ambos protocolos, IPv4 e IPv6, en cada nodo de la red. Cada nodo con doble pila en la red tendrá dos direcciones de red, una IPv4 y otra IPv6.

- *A favor: Fácil de desplegar y extensamente soportado.*
- *En contra: La topología de red requiere dos tablas de encaminamiento y dos procesos de encaminamiento. Cada nodo en la red necesita tener actualizadas las dos pilas.*

Los túneles permiten conectarse a redes IPv6 "saltando" sobre redes IPv4. Estos túneles trabajan encapsulando los paquetes IPv6 en paquetes IPv4 teniendo como siguiente capa IP el protocolo número 41, y de ahí el nombre proto-41. De esta manera, se pueden enviar paquetes IPv6 sobre una infraestructura IPv4. Hay muchas tecnologías de túneles disponibles. La principal diferencia

está en el método que usan los nodos encapsuladores para determinar la dirección a la salida del túnel.

La traducción es necesaria cuando un nodo que sólo soporta IPv4 intenta comunicar con un nodo que sólo soporta IPv6. Los mecanismos de traducción se pueden dividir en dos grupos basados en si la información de estado está guardada o no:

- *Con estado: NAT-PT (RFC 2766), TCP-UDP Relay (RFC 3142), Socks-based Gateway (RFC 3089)*
- *Sin estado: Bump-in-the-Stack, Bump-in-the-API (RFC 276)*

Despliegue de IPv6

Varios de los mecanismos mencionados más arriba se han implementado para acelerar el despliegue de IPv6. Los distintos servicios de control de Internet han ido incorporando soporte para IPv6, así como los controladores de los dominios de nivel superior (o ccTLD, en inglés).

5. Servidor HTTP Apache

*El **servidor HTTP Apache** es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de internet. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. En inglés, a patchy server (un servidor "parcheado") suena igual que Apache Server.*

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

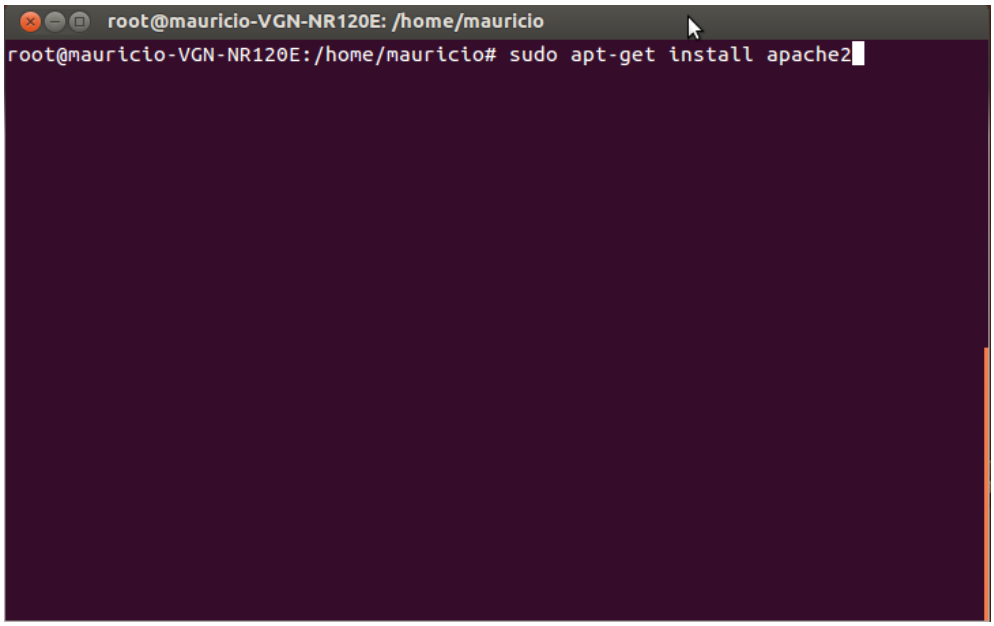
Apache presenta entre otras características altamente configurables, bases de datos de

autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años. (Estadísticas históricas y de uso diario proporcionadas por Netcraft).

Instalación:

Para instalar Apache2, desde consola ejecutamos el siguiente comando "sudo apt-get install apache2"

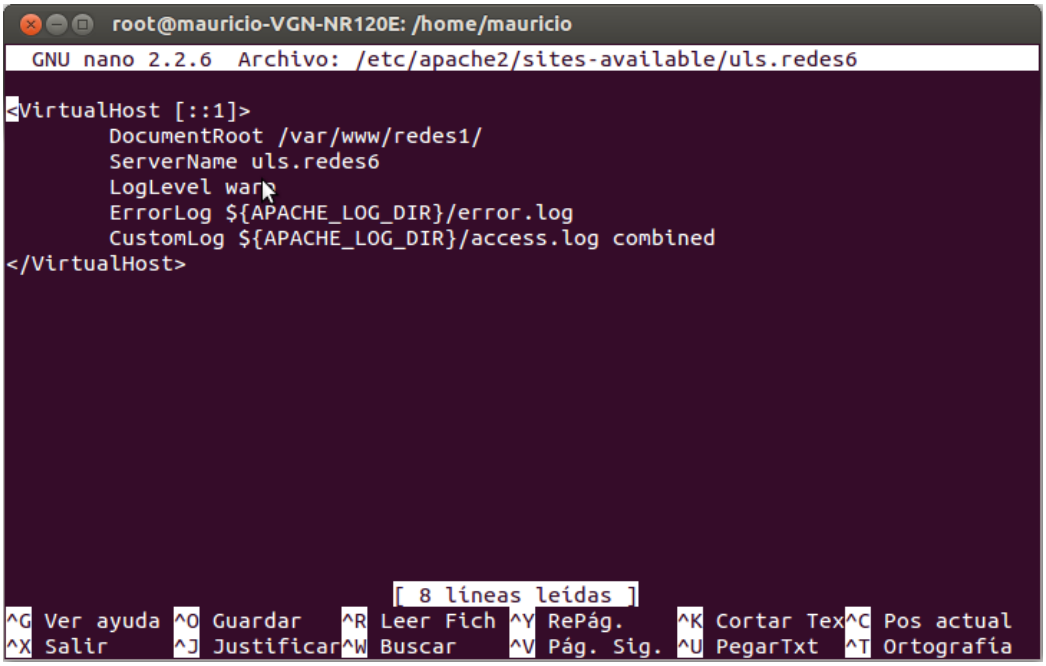


Al terminar la instalación, simplemente abrimos el navegador y escribimos "localhost" y nos mostrara la pagina de inicio de nuestro servidor

Configuración

A este nivel nuestro servidor solo responderá a IPv4, por lo que en los pasos mostraremos como configurar nuestro servidor para que responda a IPv4 e IPv6, como primer paso crearemos un host virtual:

Crear el archivo /etc/apache2/sites-available/nombre_de_tu_servidor con el siguiente contenido:



Luego activamos nuestro host virtual con el siguiente comando: "# a2ensite nombre_de_tu_servidor"

Luego reiniciamos el servicio de nuestro servidor web, ejecutando desde consola el siguiente comando: "/etc/init.d/apache2 restart"

Y listo ya tenemos nuestro servidor respondiendo a su dirección IPv4 e IPv6

Imagen 1.0, pagina web de prueba en version IPv4 (127.0.0.1/redes1)

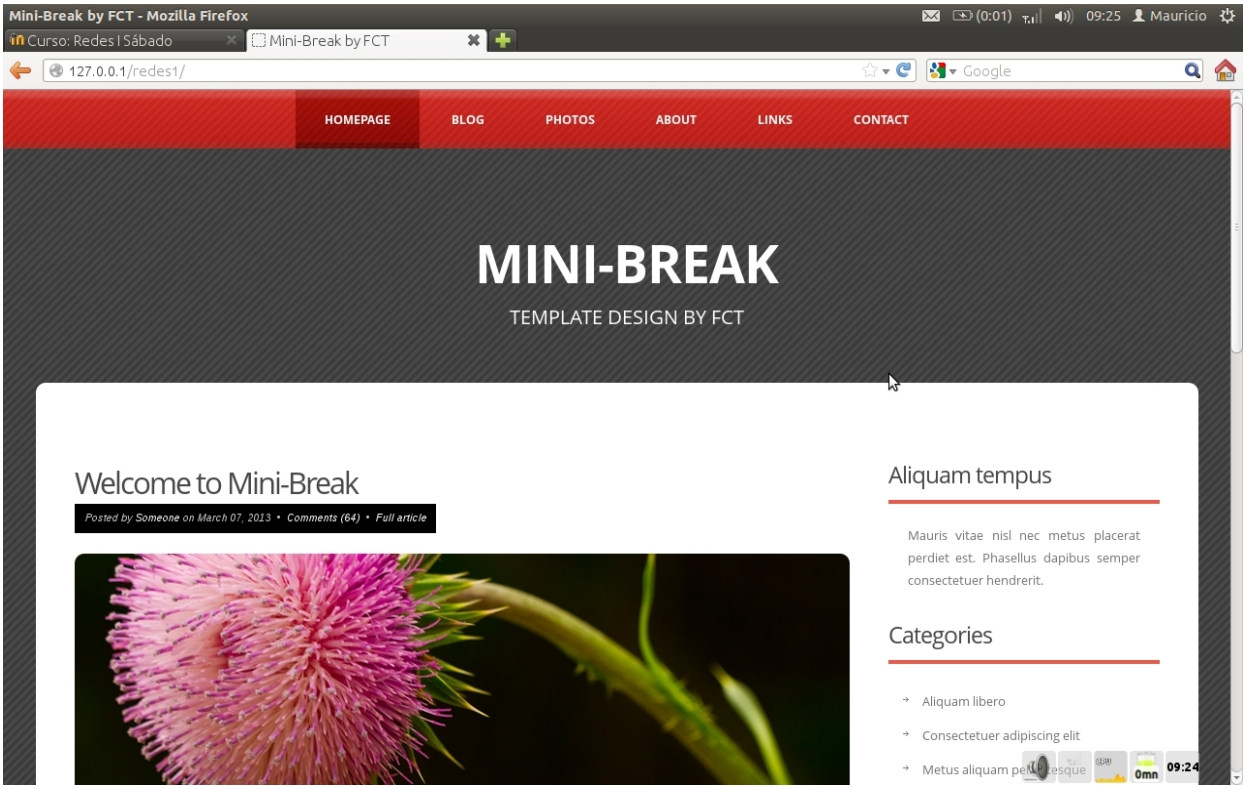
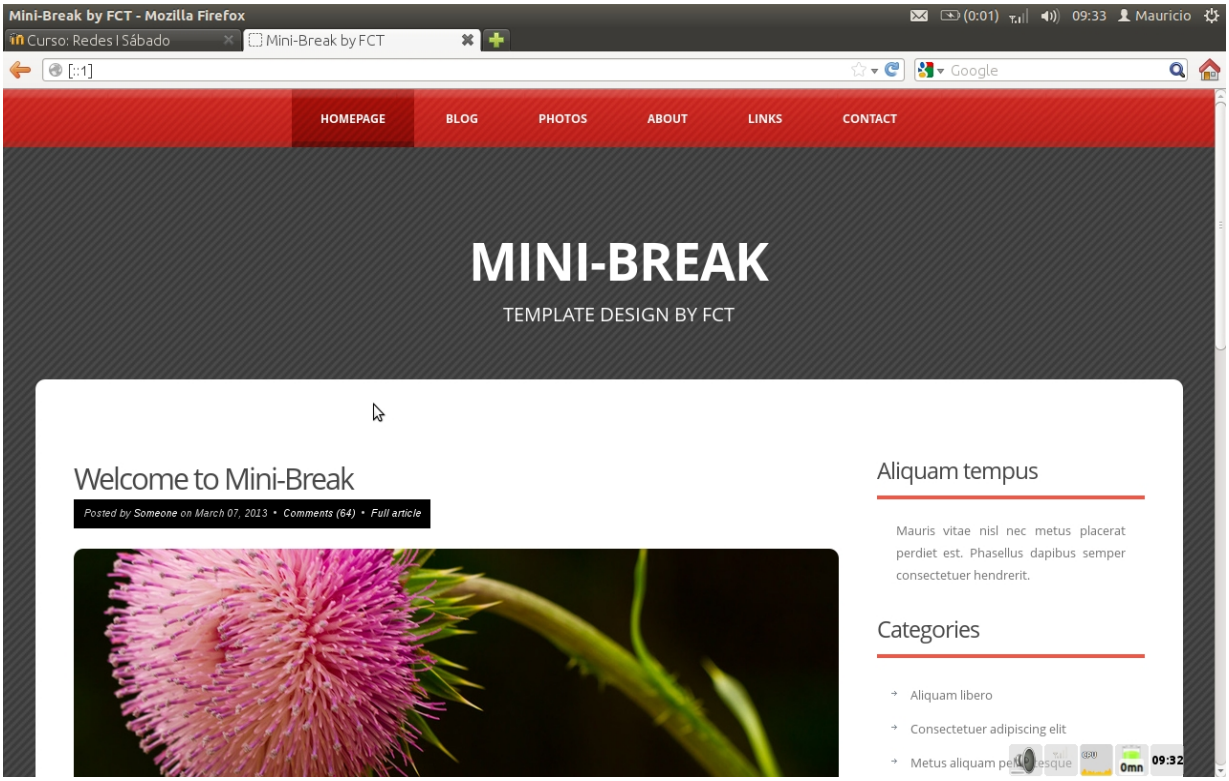


Imagen 2.0, pagina web de prueba en version IPv6 ([::1])



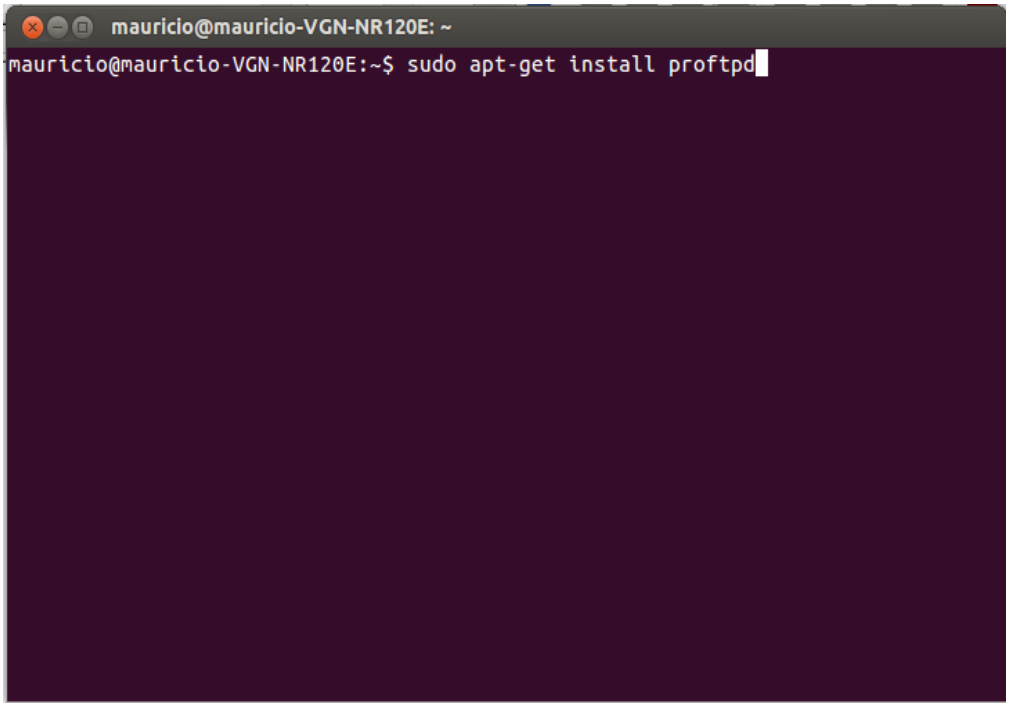
6. Servidor FTP(File Transfer Protocol)

FTP *(siglas en inglés de* ***F****ile* ***T****ransfer* ***P****rotocol, 'Protocolo de Transferencia de Archivos') en informática, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente-servidor. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo.*

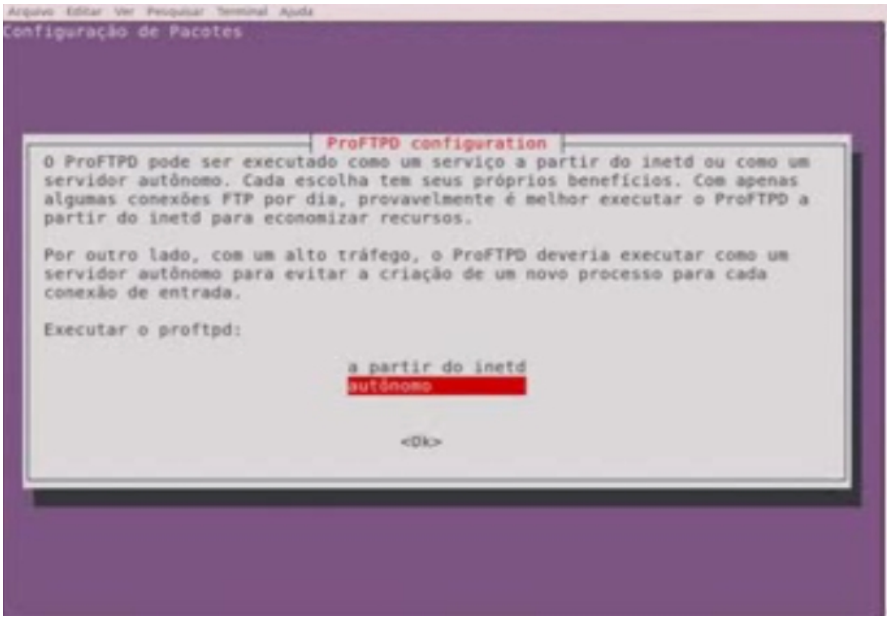
El servicio FTP es ofrecido por la capa de aplicación del modelo de capas de red TCP/IP al usuario, utilizando normalmente el puerto de red 20 y el 21. Un problema básico de FTP es que está pensado para ofrecer la máxima velocidad en la conexión, pero no la máxima seguridad, ya que todo el intercambio de información, desde el login y password del usuario en el servidor hasta la transferencia de cualquier archivo, se realiza en texto plano sin ningún tipo de cifrado, con lo que un posible atacante puede capturar este tráfico, acceder al servidor y/o apropiarse de los archivos transferidos.
Para solucionar este problema son de gran utilidad aplicaciones como scp y sftp, incluidas en el paquete SSH, que permiten transferir archivos pero cifrando todo el tráfico.

Instalación:

Para instalar nuestro servidor ftp (para nuestro ejemlo usaremos proftpd), ejecutamos desde consola el siguiente comando: "sudo apt-get install proftpd"

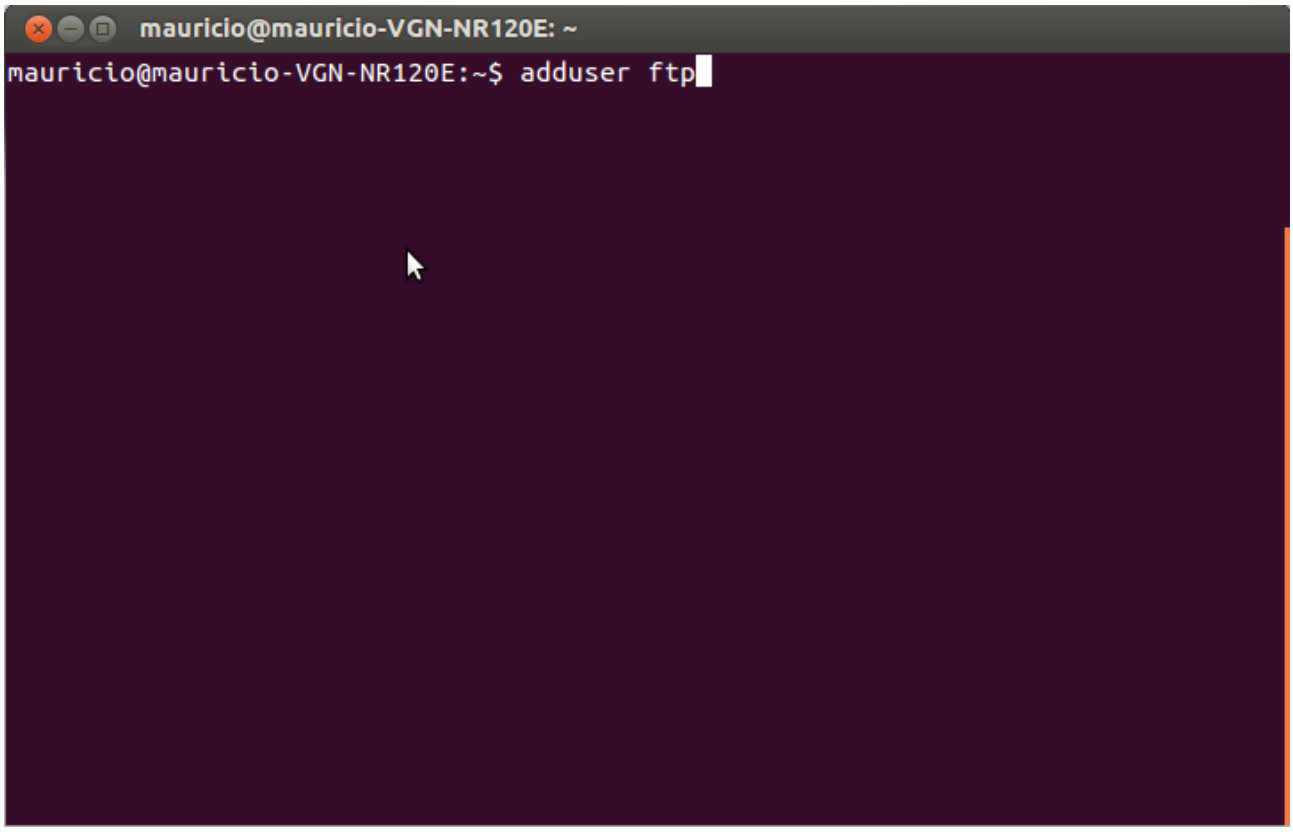


Al mostrar este mensaje seleccionar la opción anónimo

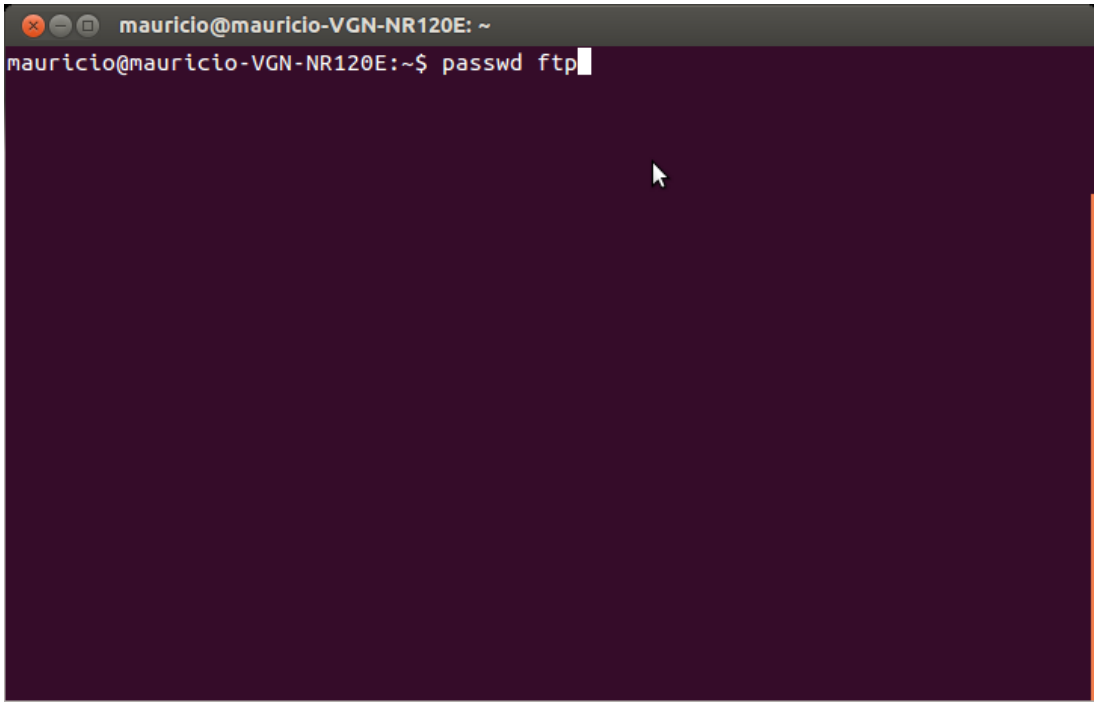


Configuración

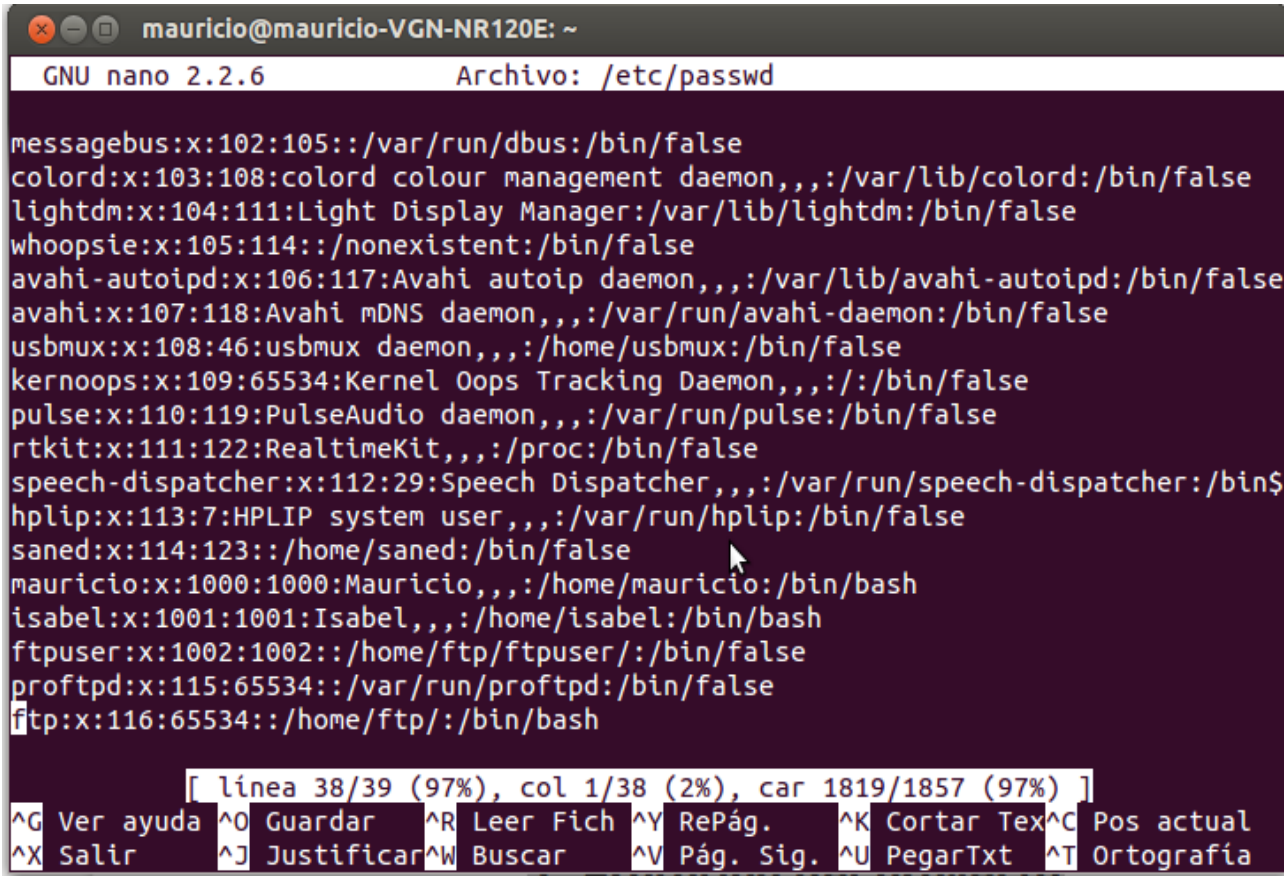
Agreganos un usuario donde se alojaran los archivos, ejecutamos el comando: "adduser ftp"



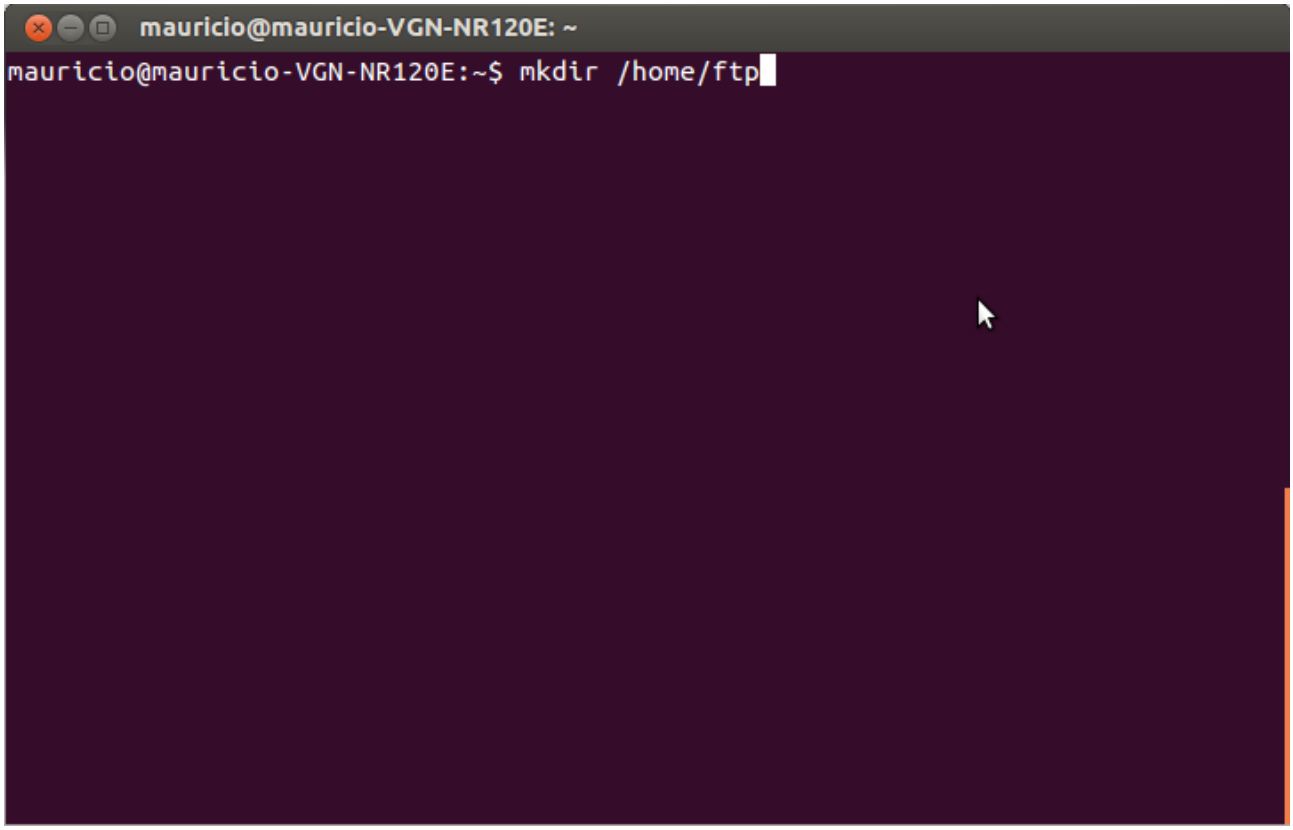
luego le asignamos su contraseña ejecutando el siguiente comando: "passwd ftp"



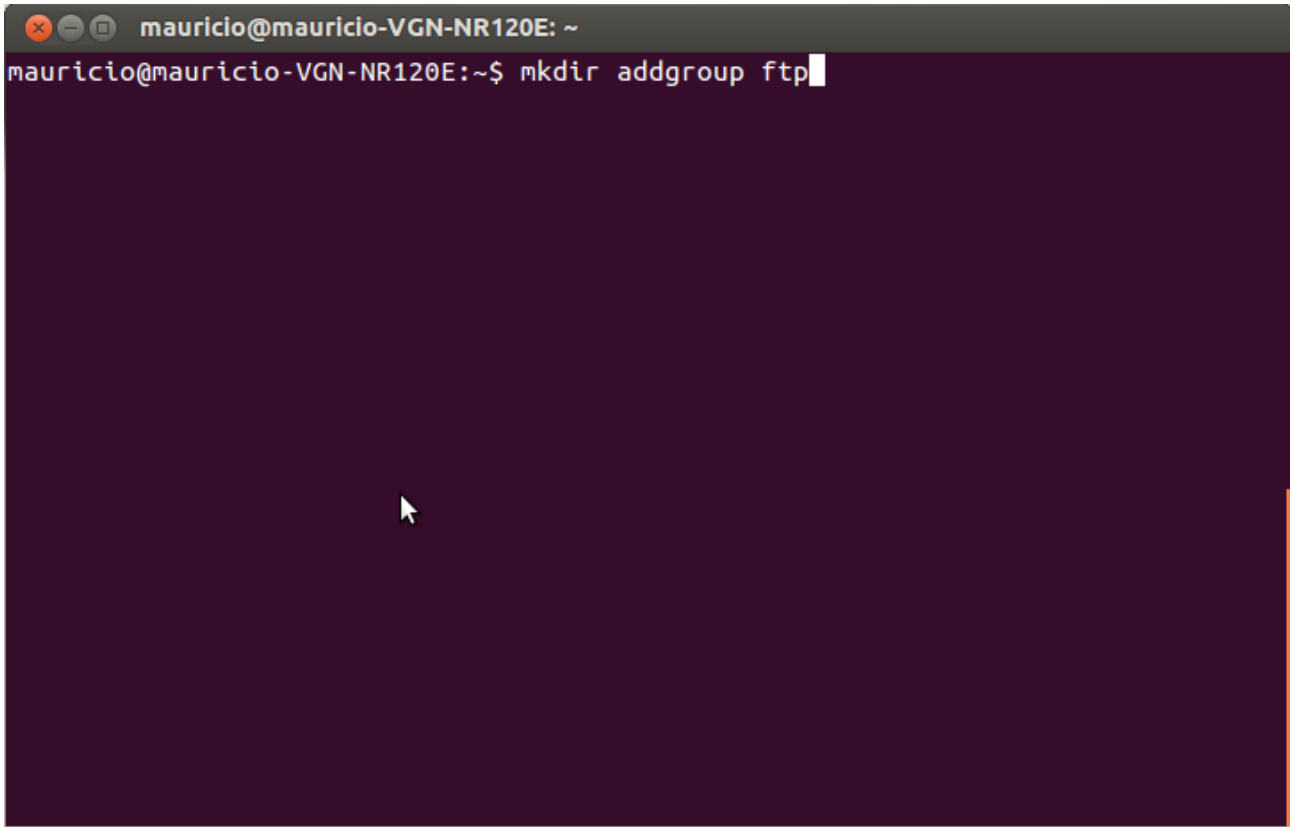
Luego editamos el archivo /etc/passwd, buscamos la linea: ftp:x:116:65534::/srv/ftp:/bin/false y la sustituimos por esta ftp:x:116:65534::/home/ftp/:/bin/bash



Luego creamos los directorios para nuestro usuario ftp, ejecutamos el comando: "mkdir /home/ftp"



Luego creamos un grupo para nuestro servidor, ejecutamos el comando: "addgroup ftp"



Luego cambiamos la configuración de nustro servidor, para ello editamos el archivo /etc/proftpd/proftpd.conf

y sustituimos su contenido por este.

codigo

```
#
# /etc/proftpd/proftpd.conf -- This is a basic ProFTPD configuration file.
# To really apply changes, reload proftpd after modifications, if
# it runs in daemon mode. It is not required in inetd/xinetd mode.
#

# Includes DSO modules
Include /etc/proftpd/modules.conf

# Set off to disable IPv6 support which is annoying on IPv4 only boxes.
UseIPv6                 on
# If set on you can experience a longer connection delay in many cases.
IdentLookups            off

ServerName              "Servidor FTP"
ServerType              standalone
DeferWelcome            off

MultilineRFC2228        on
DefaultServer           on
ShowSymlinks            on

TimeoutNoTransfer       600
TimeoutStalled          600
TimeoutIdle             1200
```

```
DisplayLogin                    welcome.msg
DisplayChdir                    .message true
ListOptions                "-l"

DenyFilter                 \*.*/

# Use this to jail all users in their homes
DefaultRoot                ~

# Users require a valid shell listed in /etc/shells to login.
# Use this directive to release that constrain.
# RequireValidShell         off

# Port 21 is the standard FTP port.
Port                       21

# In some cases you have to specify passive ports range to by-pass
# firewall limitations. Ephemeral ports can be used for that, but
# feel free to use a more narrow range.
# PassivePorts                    49152 65534

# If your host was NATted, this option is useful in order to
# allow passive tranfers to work. You have to use your public
# address and opening the passive ports used on your firewall as well.
# MasqueradeAddress               1.2.3.4

# This is useful for masquerading address with dynamic IPs:
# refresh any configured MasqueradeAddress directives every 8 hours
<IfModule mod_dynmasq.c>
# DynMasqRefresh 28800
</IfModule>

# To prevent DoS attacks, set the maximum number of child processes
# to 30.  If you need to allow more than 30 concurrent connections
```

```
# at once, simply increase this value.  Note that this ONLY works
# in standalone mode, in inetd mode you should use an inetd server
# that allows you to limit maximum number of processes per service
# (such as xinetd)
MaxInstances                    30

# Set the user and group that the server normally runs at.
User                            ftp
Group                           ftp

# Umask 022 is a good standard umask to prevent new files and dirs
# (second parm) from being group and world writable.
Umask                           022  022
# Normally, we want files to be overwriteable.
AllowOverwrite                  on

# Uncomment this if you are using NIS or LDAP via NSS to retrieve passwords:
# PersistentPasswd              off

# This is required to use both PAM-based authentication and local passwords
# AuthOrder                     mod_auth_pam.c* mod_auth_unix.c

# Be warned: use of this directive impacts CPU average load!
# Uncomment this if you like to see progress and transfer rate with ftpwho
# in downloads. That is not needed for uploads rates.
#
# UseSendFile                   off

TransferLog /var/log/proftpd/xferlog
SystemLog   /var/log/proftpd/proftpd.log

# Logging onto /var/log/lastlog is enabled but set to off by default
#UseLastlog on
```

 # In order to keep log file dates consistent after chroot, use timezone info
 # from /etc/localtime. If this is not set, and proftpd is configured to
 # chroot (e.g. DefaultRoot or <Anonymous>), it will use the non-daylight
 # savings timezone regardless of whether DST is in effect.
#SetEnv TZ :/etc/localtime

<IfModule mod_quotatab.c>
QuotaEngine off
</IfModule>

<IfModule mod_ratio.c>
Ratios off
</IfModule>

 # Delay engine reduces impact of the so-called Timing Attack described in
 # http://www.securityfocus.com/bid/11430/discuss
 # It is on by default.
<IfModule mod_delay.c>
DelayEngine on
</IfModule>

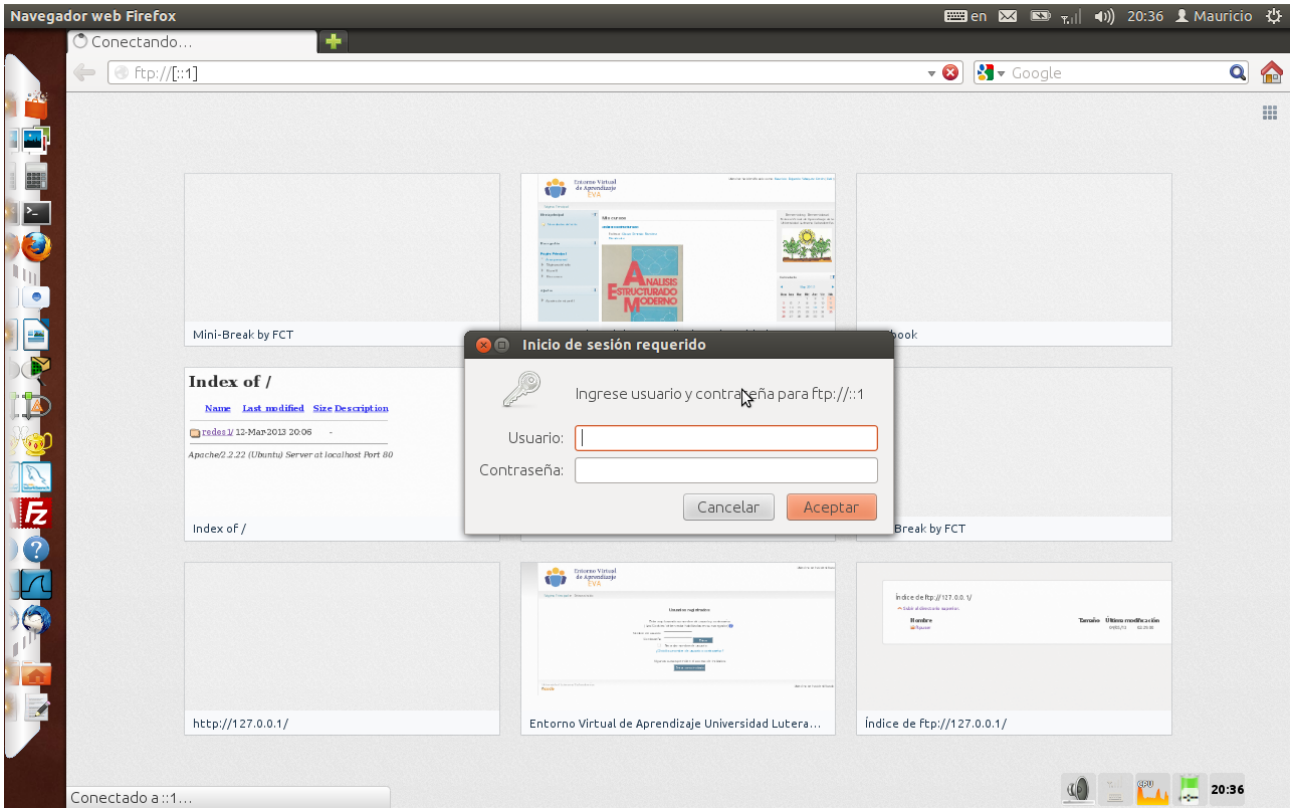
<IfModule mod_ctrls.c>
ControlsEngine off
ControlsMaxClients 2
ControlsLog /var/log/proftpd/controls.log
ControlsInterval 5
ControlsSocket /var/run/proftpd/proftpd.sock
</IfModule>

<IfModule mod_ctrls_admin.c>
AdminControlsEngine off
</IfModule>

```
#
# Alternative authentication frameworks
#
#Include /etc/proftpd/ldap.conf
#Include /etc/proftpd/sql.conf

#
# This is used for FTPS connections
#
#Include /etc/proftpd/tls.conf

#
# Useful to keep VirtualHost/VirtualRoot directives separated
#
#Include /etc/proftpd/virtuals.con

# A basic anonymous configuration, no upload directories.

# <Anonymous ~ftp>
#   User                ftp
#   Group               nogroup
#   # We want clients to be able to login with "anonymous" as well as "ftp"
#   UserAlias           anonymous ftp
#   # Cosmetic changes, all files belongs to ftp user
#   DirFakeUser  on ftp
#   DirFakeGroup on ftp
#
#   RequireValidShell   off
#
#   # Limit the maximum number of anonymous logins
#   MaxClients          10
#
#   # We want 'welcome.msg' displayed at login, and '.message' displayed
#   # in each newly chdired directory.
```

```
#  DisplayLogin                welcome.msg
#  DisplayChdir       .message
#
#  # Limit WRITE everywhere in the anonymous chroot
#  <Directory *>
#    <Limit WRITE>
#      DenyAll
#    </Limit>
#  </Directory>
#
#  # Uncomment this if you're brave.
#  # <Directory incoming>
#  #   # Umask 022 is a good standard umask to prevent new files and dirs
#  #   # (second parm) from being group and world writable.
#  #   Umask                       022  022
#  #            <Limit READ WRITE>
#  #            DenyAll
#  #            </Limit>
#  #            <Limit STOR>
#  #            AllowAll
#  #            </Limit>
#  # </Directory>
#
# </Anonymous>

# Include other custom configuration files
Include /etc/proftpd/conf.d/
```

Y listo ya tenemos nuestro servidor ftp funcional



7. Bibliografia

http://www.youtube.com/watch?v=9W8mjA8fw_E *(03 – Mayo – 2013)*

http://es.wikipedia.org/wiki/IPv4 *(11-Mayo-2013)*

http://es.wikipedia.org/wiki/IPv6 *(11-Mayo-2013)*

8. Diagrama de GANT

No.	Actividad	Semanas															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	*Recopilación y análisis de la información necesaria.*																
2	*Recopilación de Hardware y Software necesario.*																
3	*Pruebas en configuración de IPv4 y IPv6.*																
4	*Documentación y análisis de los resultados.*																
5	*Generación de reporte final.*																

Realizado
Retrasado
En curso
Programado