Universidad de las Ciencias Informáticas

Título: "Procedimiento para el procesamiento y almacenamiento de los protocolos más utilizados por los servicios AAA en redes IP."

Trabajo de Diploma para optar por el título de

Ingeniero en Ciencias Informáticas

Autor(es): Julio Igarza Rondón
Jorge Luis Palmero Horta

Tutor: MSc. Ing. Omar Yera Pérez

Asesor: MSc. Pedro Carlos Pérez Martinto

Junio de 2009

Ciudad de La Habana

"***Año del 50 aniversario del triunfo de la Revolución***"

"Investigar significa pagar la entrada por adelantado y entrar sin saber lo que se va a ver"

(Oppenheimer)

DATOS DE CONTACTO

Tutor: MSc. Ing. Omar Yera Pérez
E-mail: oyera@enet.cu

Graduado en 1984 en el ISPJAE en la especialidad Máquinas Computadoras. Máster en Ciencias. Ha trabajado en temas relacionados con las redes IP y el desarrollo de aplicaciones y sistemas durante más de 20 años.

Asesor: MSc. Pedro Carlos Pérez Martinto

E-mail: pcpmartinto@uci.cu

Graduado en 1990 en el Instituto Superior Pedagógico Enrique José Varona. Especialidad Defectología. Especializaciones: Oligofrenopedagogía y Tiflopedagogía. Es especialista y profesor principal de la asignatura Metodología de la Investigación científica en la UCI desde el curso 2006-2007.

AGRADECIMIENTOS

Jorge Luis:

Quiero agradecer a la Revolución por permitirme estudiar gratuitamente, a nuestro compañero Fidel Castro Ruz por tener la grandiosa idea de crear esta universidad, que cumplió con mis expectativas como estudiante, en ella dejo enormes y muy buenos recuerdos así como amistades. Mis agradecimientos van también dirigidos a aquellos que tuvieron que ver con mi carrera en el sector docente, profesores, dirigentes de la FEU y de la UJC, decana y demás integrantes del secretariado docente; no puedo dejar de mencionar a los cadetes insertados del MININT, al Estado Mayor, a Nenita, a Sander, que confiaron en que haría el esfuerzo, me recuperaría y lograría culminar este trabajo de diploma, a los integrantes de este proyecto que hemos estado bien unidos y hemos compartido buenos momentos además de bastantes preocupaciones, a los hermanos Yera que ambos asumieron la tutoría de este trabajo y me ayudaron mucho. Mis amigos tienen también mis agradecimientos, los amigos de siempre, los de ahora y a los que no son tanto como amigos pero me han ayudado a sobrellevar la lejanía del hogar, la carga docente entre otras dificultades, les agradezco mucho su paciencia y cariño.

Julio:

A mi mamá Mercedes, que es mi principal motivo de inspiración en la vida, por todo el gran sacrificio que ha hecho para llegar hasta donde he llegado, por su inmenso cariño, su educación y por saber guiarme por el mejor camino siendo madre y padre a la vez.

A mi hermana Daynelis, por haberme resistido desde pequeña y estar siempre a mi lado en todo momento.

A mis abuelos Eloísa y Tomás, a mis tíos los Alfredos, a Kenia, a la familia en general, por impregnarme tanto amor, respeto y dedicación entregada.

A mis padrinos Manuel y Teresita, que más que padrinos son mis segundos padres, y se que para ellos soy como un hijo, gracias por toda la ayuda brindada incondicionalmente.

Al compañero Fidel Castro, por haber tenido la tan ingeniosa idea de convertir el antiguo Centro Lourdes en la tal útil y necesaria universidad que es hoy, y al MININT por darme la posibilidad de estudiar en ella.

A todos mis amigos, a los hermanos Zilber, Heiler, Ariel, Osvey, Jorgito, gracias por aceptarme como soy, por todo el tiempo que hemos compartido, por los momentos tristes y alegres que hemos vivido juntos, por las largas horas de estudio, etc.

A nuestros tutores Omar y Adrian, por habernos ayudado y dedicarnos tiempo de trabajo y vida personal para la realización de este trabajo.

A todos los que de forma directa o indirecta han contribuido a que llegara a la cima de esta gran montaña.

Gracias.

DEDICATORIA

Jorge Luis:

Deseo agradecer a mi familia, a toda mi familia que no solo somos los que tenemos vínculos sanguíneos sino también a los que con amor y respeto nos hemos convertido también en madres e hijos, hermanos y hermanas, tíos y sobrinos, etc. y que se han preocupado siempre por mí, han exigido que estudie y que me guíe por el buen camino de la vida y me han ayudado a lograrlo, a mi tío Luis Horta Torres por estar siempre ahí para mí y mucho que lo necesité. He dejado a mi mamá para último con toda intención, a todos los antes mencionados les agradezco mucho lo que han hecho y han significado para mí, pero a mi mamá Lourdes Milagros Horta Torres no solo le agradezco sino que también le dedico este trabajo y toda mi carrera, por haber hecho de mí el hombre que soy hoy, por depositar todos sus esfuerzos en mí, y por quererme tanto, mamá te amo, quiero aprovechar esta dedicatoria para empezar a retribuirte todo lo que con grandes sacrificios has resguardado en mí.

Julio:

Primeramente este resultado está dedicado especialmente a mi mamá Mercedes, a mi hermana Daynelis, a toda mi familia y seres muy queridos que aunque no de sangre, también forman parte de ella, por la espléndida educación que han sembrado en mí. A los grandes y verdaderos amigos que con su sinceridad y respeto forman y formarán parte de la historia de mi vida. Al líder de líderes, compañero Fidel Castro por la manera en que sabido desarrollar la educación cubana. A todos los profesores que han contribuido a mi desarrollo como persona, estudiante y futuro profesional.

RESUMEN

El Ministerio del Interior (MININT) se encuentra insertado en la revolución tecnológica que se lleva a cabo en el campo de la Informática y las Comunicaciones en nuestro país, utilizando los recursos que le son imprescindibles para realizar su labor y trabajando arduamente en garantizar la seguridad de la información que por esta vía se intercambia. La utilización de elementos informáticos que permiten la correcta autenticación, la contabilidad detallada y la autorización del acceso de sus usuarios a las redes IP es una prioridad.

La presente investigación aporta la documentación detallada de los protocolos AAA, RADIUS y DIAMETER, el estudio del modelo TCP/IP y el diseño de un procedimiento que permita capturar, procesar y almacenar en línea, los datos que se intercambian utilizando dichos protocolos. Los resultados de este trabajo pueden utilizarse en el desarrollo de aplicaciones que permitan el seguimiento y el análisis de las trazas que se obtienen de dichos protocolos a partir de su almacenamiento en bases de datos.

TABLA DE CONTENIDOS

1

INTRODUCCIÓN

Desde la década de los 90 la situación de la industria de las telecomunicaciones ha cambiado de manera sustancial. Se han producido constantes actualizaciones de tecnología y un decrecimiento importante del costo de la red. Las líneas fijas tradicionales están siendo sustituidas por la telefonía móvil y la telefonía IP (Internet Protocol). En general, la revolución que está ocurriendo en las redes y servicios de telecomunicaciones, hace que aparezcan nuevos retos en el trabajo de seguridad: se unen los servicios tradicionales de telefonía, televisión y datos, aumenta la complejidad de los servicios y su valor, y se reducen los tiempos para introducir un nuevo servicio al mercado.[1]

El mejoramiento de las telecomunicaciones implica un aceleramiento en el proceso de extenderla a todo el país. En la sociedad actual, más que ninguna otra anterior, estamos viviendo una etapa donde la información se ha convertido en algo de vital importancia. El mundo de las tecnologías depende de un modo u otro de la información que se pueda obtener y procesar.

Con el surgimiento del Plan Informatización de la Sociedad, se ha ampliado grandemente el sistema de redes nacionales, ejemplo de ello lo constituye la idea de crear Joven Club de Computación en todo el país, desde el primero, inaugurado en septiembre de 1987 hasta el último, hoy contamos con al menos uno en cada municipio del territorio nacional. Sin dejar de mencionar la red de Infomed, en funcionamiento desde 1992, la cual le presta variados servicios de información especializados en el sector de la salud a profesionales, médicos y paramédicos.

Uno de los acontecimientos más importantes fue la creación de la Empresa de Telecomunicaciones Sociedad Anónima (ETECSA) en 1994, como una empresa mixta con capital extranjero, hasta entonces el país tenía una red telefónica escasa y anticuada tecnológicamente.

Pero sin dudas el paso más grande dado en la década del 90 fue la conectividad cubana a la gran red de redes, Internet, exactamente en el año 1994, lo cual nos permitió abrirnos camino al mundo de los grandes volúmenes de información.

[1] Baluja García, MSc. Walter y Dra. Caridad Anias Calderón. Propuesta de Arquitectura de Seguridad para las redes de Telecomunicaciones.

En los últimos años se ha venido desplegando una red nacional de fibra óptica, con capacidad de trafico para desarrollar la red nacional de telecomunicaciones y, por supuesto, el fortalecimiento de intranet y el progresivo empleo de internet para los distintos propósitos que se propone abarcar el Plan de Informatización de la Sociedad, que debe dar resultados positivos en cuanto a la extensión y profundización del acceso y uso de las redes de computadoras.

Existe un proyecto organizado por el Ministerio de la Informática y las Comunicaciones (MIC) de desarrollo de la Red Cuba, que tecnológicamente se basa en establecer una gran red IP que integre servicios que tradicionalmente estaban separados como por ejemplo la red telefónica y la red de datos. De tal manera, se pretende que exista una red todo IP, que agrupe todos los servicios en una misma red. Así se están realizando varias inversiones relacionadas con el aumento de las conectividades exteriores.

En estos momentos contamos con más de 100 redes nacionales y más de 60 000 usuarios con acceso telefónico autorizado a organismos de la administración del Estado, del sector empresarial y corporativo de la economía, así como a entidades de otras esferas de la sociedad, como la prensa y la cultura, la medicina, etc. que se han ido incorporando a la utilización de Internet como medio de difusión global.

El MININT se encuentra insertado en esta revolución tecnológica desde los primemos momentos, utilizando los recursos que le son imprescindibles para su labor y trabajando arduamente en garantizar la seguridad de la información que por esta vía se intercambia. La utilización de elementos informáticos que permiten la correcta autenticación, la contabilidad detallada y la autorización del acceso de los usuarios a las redes IP es un ejemplo de dicha afirmación.

No obstante lo anterior y a pesar de que muchos de los dispositivos que realizan el AAA almacenan trazas de los accesos que autorizan, el MININT no cuenta con los mecanismos que le permitan realizar el procesamiento en línea y el almacenamiento de todos los datos de interés que se intercambian dentro de los protocolos AAA, para utilizarlos en análisis posteriores en función de la seguridad de las redes que opera.

Problema a resolver

¿Cómo realizar el procesamiento y almacenamiento de los datos de autenticación, autorización y contabilidad que se intercambian en las redes IP atendidas por el MININT?

Objeto de estudio

Datos intercambiados por los protocolos AAA, que son utilizados en el proceso de autenticación en las redes IP atendidas por el MININT.

Objetivo general

Definir un procedimiento que permita el procesamiento y el almacenamiento de los datos transportados por los protocolos AAA empleados en las redes IP que son atendidas por el MININT, utilizando las normas establecidas en los RFC, que regulan el funcionamiento del modelo TCP/IP, para el control de los accesos a las redes de Datos.

Campo de acción

El procesamiento y almacenamiento de los datos de interés que se obtienen de los protocolos AAA utilizados en las redes IP.

Idea a defender

Un procedimiento, elaborado sobre la base del funcionamiento del modelo TCP/IP, para realizar el procesamiento y almacenamiento de los principales datos transportados por los protocolos AAA, permitirá al MININT aplicar herramientas de procesamiento y análisis propias en función de la seguridad de las redes IP que atiende.

Posibles resultados

- El estudio del modelo TCP/IP y la documentación detallada del principio de funcionamiento de los protocolos AAA RADIUS y DIAMETER.
- El diseño de un procedimiento que permita capturar, procesar y almacenar en línea, los datos más importantes que se intercambian utilizando dichos protocolos.

Métodos y procedimientos

Métodos teóricos

- Históricos
    - Analítico: Estudiar cada frame del protocolo AAA RADIUS y DIAMETER para establecer su composición dentro de los protocolos de transporte UDP y TCP respectivamente.
    - Sistémico: Se realizará la modelación de la forma de trabajo del protocolo AAA RADIUS y DIAMETER a partir de la relación entre los frames que conforman el proceso de autorización, autenticación y contabilidad.

Métodos empíricos

- Observación científica.
    - Observación documental: Se realizará a partir del estudio de la bibliografía y de los RFC que reglamentan la forma de utilización de los protocolos AAA objetos de estudio.
    - Observación de campo: Se observará el protocolo AAA RADIUS en alguna de las redes IP atendidas por el MININT, obteniendo capturas reales. Se podrán utilizar capturas que pertenezcan a otros lugares donde también se utilice dicho protocolo.

Este trabajo cuenta en su estructura con introducción, desarrollo de 3 capítulos, conclusiones, recomendaciones, bibliografía, anexos y glosario.

El capitulo 1 presenta la fundamentación teórica y el estado del arte, que incluye las principales características de los modelos en capas OSI y TCP/IP, la importancia de los servicios AAA, incluidos los protocolos que encierra en su familia y que aplicaciones tienen.

El capitulo 2 abarca la descripción del procedimiento para analizar los protocolos de las capas del modelo TCP/IP, para lo cual se hizo un estudio de los protocolos Ethernet, IP, TCP, UDP, DIAMETER y RADIUS. De estos se puntualizó en sus aspectos primordiales, como son el formato de sus mensajes, campos que los componen y como se produce la comunicación entre ellos. Además de una propuesta de almacenamiento de los campos más importantes de los protocolos analizados.

El capitulo 3 contiene lo relacionado con las pruebas hechas al procedimiento del protocolo RADIUS para comprobar la veracidad del mismo. Se utilizó para analizar los protocolos de red una herramienta llamada Wireshark.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Desde el surgimiento de las primeras computadoras se hizo necesario el intercambio de información entre ordenadores. Muchas son las variantes para lograr el mutuo envío de ficheros entre las computadoras; dependiendo del tipo de dato a transmitir se utiliza la forma más adecuada para logarlo exitosamente. Como entre las personas existen los idiomas para hablar y poder entenderse, las máquinas necesitaron de algún lenguaje en común que permitiera establecer una comunicación, es así como aparece el término protocolo, que no es más que las reglas y procedimientos que utilizan los dispositivos de red para interactuar entre sí.

La comunicación surge al existir un mensaje o algún tipo de información que debe ser enviada desde una persona o dispositivo a otro. Las personas intercambian informaciones mediante distintos métodos de comunicación. Estos métodos están ligados por tres elementos en común. El primero de estos elementos es el origen del mensaje o emisor. Los orígenes de los mensajes son las personas o los dispositivos electrónicos interesados en enviar un mensaje a otras personas o dispositivos. El segundo elemento de la comunicación es el destino o receptor del mensaje o la información. El destino recibe el mensaje y lo interpreta. Un tercer elemento, llamado canal, está formado por los medios que proporcionan el camino por el que el mensaje viaja desde el origen hasta el destino. Estos mensajes pueden ser enviados a través de una red de datos o de información convirtiéndolos primero en dígitos binarios o bits. Después, estos bits se codifican en una señal transmisible por el medio apropiado. En las redes de computadoras, el medio generalmente es un tipo de cable o una transmisión inalámbrica.

En cuanto a la comunicación de mensajes, se puede decir que se pueden enviar e-mails, videos, etc., mediante una red desde un emisor hasta un receptor como un stream de bits masivo y continuo, si esto ocurre de la manera anteriormente descrita, entonces se producen retrasos importantes debido al gran tamaño de los streams de datos, además ningún otro dispositivo puede enviar o recibir mensajes en esa red mientras se efectúe la transferencia de datos. Otro inconveniente es que si falla algún enlace en la infraestructura de red interconectada mientras se realizaba el intercambio de datos se pierde completamente el mensaje.

Existe otra manera más eficiente de enviar datos, esta consiste en fragmentar los streams, convirtiéndolos en partes más pequeñas y manejables. Este proceso de fragmentar los streams se denomina segmentación. Debido a que con la segmentación se envían partes independientes de menor tamaño del origen al destino, es posible entrelazar varias conversaciones en la red. Para poder entrelazar las piezas de conversaciones diferentes es necesario un mecanismo o proceso que realice esta función, este proceso lleva por nombre multiplexación. La segmentación brinda una mayor seguridad en la red, ya que los paquetes no deben tomar el mismo camino para ir del origen al destino. En otras palabras si ocurren fallas en la red, los segmentos del mensaje enviado pueden tomar canales alternativos para llegar al destino, evitando así que colapse la transferencia, en caso de no recibir todos los segmentos solo se deben redireccionar aquellos que no hayan llegado al receptor. La desventaja principal de este proceso es el nivel de complejidad que aumenta considerablemente, lo que no seria muy relevante para mensajes pequeños, pero en caso de mensajes de envergadura sería un gran costo de tiempo utilizar esta vía de transferencia.

1.1 Modelo OSI y modelo TCP/IP

Para visualizar el funcionamiento de los protocolos, frecuentemente se emplea un modelo en capas, de esta forma se muestra la interacción de las capas sobre y debajo del protocolo. Se deben llevar a cabo muchos procedimientos separados para conseguir un intercambio fiable de datos entre dos computadoras.

De modo que se obtiene un software de comunicaciones muy complejo. Con un modelo en capas o niveles resulta más sencillo agrupar funciones relacionadas e implementar el software de comunicaciones modular.

Las capas están jerarquizadas. Cada capa se construye sobre su predecesora. El número de capas y, en cada una de ellas, sus servicios y funciones son variables con cada tipo de red. Sin embargo, en cualquier red, la misión de cada capa es proveer servicios a las capas superiores haciéndoles transparentes el modo en que esos servicios se llevan a cabo. De esta manera, cada capa debe ocuparse exclusivamente de su nivel inmediatamente inferior, a quien solicita servicios, y del nivel inmediatamente superior, a quien devuelve resultados.

El Modelo de referencia de Interconexión de Sistemas Abiertos o Modelo OSI (Open Systems Interconection). En el Modelo OSI pueden modelarse o referenciarse diversos dispositivos que reglamenta la Unión Internacional de Telecomunicaciones (International Telecommunication Union, ITU), con el objetivo de organizar todos los sistemas y componentes requeridos en la transmisión de datos. Cada dispositivo de cómputo y telecomunicaciones podrá ser referenciado en este modelo con características específicas en cada nivel.

Un modelo de protocolo proporciona un modelo que coincide fielmente con la estructura de una suite de protocolo en particular. El conjunto jerárquico de protocolos relacionados en una suite representa típicamente toda la funcionalidad requerida para interconectar la red humana con la red de datos. El Modelo TCP/IP es un modelo de protocolo porque describe las funciones que se producen en cada capa de los protocolos dentro del conjunto TCP/IP.

Un modelo de referencia proporciona una referencia común para mantener consistencia en todos los tipos de protocolos y servicios de red. Un modelo de referencia no está pensado para ser una especificación de implementación ni para proporcionar un nivel de detalle suficiente para definir de forma precisa los servicios de la arquitectura de red. El propósito principal de un modelo de referencia es asistir en la comprensión más clara de las funciones y los procesos involucrados.

El Modelo de Interconexión de Sistema Abierto (OSI) es el modelo de referencia de internetwork más ampliamente conocido. Se utiliza para el diseño de redes de datos, especificaciones de funcionamiento y resolución de problemas. Este modelo fue propuesto como una aproximación teórica y también como fase de inicio en la evolución de las redes de ordenadores. Por lo tanto, el modelo OSI es más fácil de entender, pero realmente, el que se usa es el modelo TCP/IP en el cual se basa este trabajo. (Ver anexo 1)

En este modelo, la información se pasa de una capa a otra, comenzando en la capa de Aplicación en el host de transmisión, siguiendo por la jerarquía hacia la capa más inferior, pasando por el canal de comunicaciones al host de destino, donde la información vuelve a la jerarquía y termina en la capa de Aplicación.

La capa de Aplicación, es la capa superior de los modelos OSI y TCP/IP. Es la capa que proporciona la interfaz entre las aplicaciones que utilizamos para comunicarnos y la red subyacente en la cual se transmiten los mensajes. Los protocolos de capa de aplicación se utilizan para intercambiar los datos entre los programas que se ejecutan en los hosts de origen y destino. Existen muchos protocolos de capa de aplicación y siempre se desarrollan protocolos nuevos.

Aunque el grupo de protocolos TCP/IP se desarrolló antes de la definición del modelo OSI, la funcionalidad de los protocolos de capa de aplicación de TCP/IP se adaptan aproximadamente a la estructura de las tres capas superiores del modelo OSI: Capas de Aplicación, Presentación y Sesión.

Funciones de los protocolo de la capa de Aplicación

Los protocolos de la capa 4, son utilizados tanto por los dispositivos del origen como por los del destino. Para que ocurra una exitosa comunicación deben coincidir los protocolos implementados en el host origen y destino. Muchos y diversos tipos de aplicaciones se comunican a través de las redes de datos. Por lo tanto, los servicios de la capa de Aplicación deben implementar protocolos múltiples para proporcionar la variedad deseada de experiencias de comunicación.

Cada protocolo tiene un fin específico y contiene las características requeridas para cumplir con dicho propósito. Deben seguirse los detalles del protocolo correspondiente a cada capa, así las funciones en una capa se comunican correctamente con los servicios en la capa inferior. Las aplicaciones y los servicios también pueden utilizar protocolos múltiples durante el curso de una comunicación simple.

Son muchos los protocolos en la capa de aplicación que interactúan y proporciona el intercambio de información del usuario. Estos protocolos especifican la información de control y formato necesaria para muchas de las funciones de comunicación de Internet más comunes, alguno de los protocolos son:

- El protocolo Servicio de nombres de dominio (DNS, Domain Name Service) se utiliza para resolver nombres de Internet en direcciones IP.
- El protocolo de transferencia de hipertexto (HTTP, Hypertext Transfer Protocol) se utiliza para transferir archivos que forman las páginas Web de la World Wide Web.

- El Protocolo simple de transferencia de correo (SMTP) se utiliza para la transferencia de mensajes de correo y adjuntos.
- Telnet, un protocolo de emulación de terminal, se utiliza para proporcionar acceso remoto a servidores y a dispositivos de red.
- El Protocolo de transferencia de archivos (FTP, File Transfer Protocol) se utiliza para la transferencia interactiva de archivos entre sistemas.
- Protocolo de Autenticación, Autorización y Contabilidad de usuarios y recursos de red (TACACS, TACACS+, DIAMETER Y RADIUS).

1.2 Protocolo AAA

En el mundo de la informática, las siglas AAA, que en inglés significan Authentication, Authorization and Accounting, corresponden a un tipo de protocolo, o mejor dicho, a una familia de protocolos, pues no se refiere a uno en particular, sino a un grupo, que realizan las funciones de autenticación, autorización y contabilización.

Estos protocolos operan en la capa 4 o de Aplicación del Modelo TCP/IP.

Determinar quien o que está autorizado a obtener acceso a la red es una cuestión que nace desde la concepción del internet, poder decidir quién tiene acceso es fundamental, la autenticación, autorización y el manejo de cuentas es una metodología que pretende resolver tres preguntas fundamentales del acceso a la red; ¿Quién o Qué eres?, ¿Qué tienes permitido hacer? y ¿Qué hiciste?; estas preguntas de seguridad son fundamentales.

El término AAA surge muy cerca del nacimiento de RADIUS (Remote Authentication Dial-In User Service) y actualmente es el protocolo AAA más aceptado. Existen otros protocolos y tecnologías que utilizan AAA para determinar los accesos a la red. AAA permite la movilidad y la seguridad dinámica, ya que sin AAA la red es estática y todos los accesos deben estar previamente definidos.

Actualmente existen cada vez más dispositivos móviles y diferentes ambientes de red que requieren la movilidad y el dinamismo que ofrece la AAA, el crecimiento de las redes inalámbricas, los hotspot que

ofrecen servicios de red son el tipo de redes que requieren identificar quién ingresa, a que tiene privilegios y que hicieron.

Definiendo cada una de las funciones a las que se refiere el término, la Autenticación es el proceso por el que una entidad prueba su identidad ante otra. Normalmente la primera entidad es un cliente (usuario, ordenador, etc.) y la segunda un servidor (ordenador). La autenticación se consigue mediante la presentación de una propuesta de identidad (un nombre de usuario) y la demostración de estar en posesión de las credenciales que permiten comprobarla. Ejemplos posibles de estas credenciales son las contraseñas, los testigos de un sólo uso (one-time tokens), los Certificados Digitales, ó los números de teléfono en la identificación de llamadas.

La Autorización se refiere a la concesión de privilegios específicos (incluyendo "ninguno") a una entidad o usuario basándose en su identidad (autenticada), los privilegios que solicita, y el estado actual del sistema. Las autorizaciones pueden también estar basadas en restricciones, tales como restricciones horarias, sobre la localización de la entidad solicitante, la prohibición de realizar logins múltiples simultáneos del mismo usuario, etc. La mayor parte de las veces el privilegio concedido consiste en el uso de un determinado tipo de servicio. Ejemplos de algunos tipos de servicio son: filtrado de direcciones IP, asignación de direcciones, asignación de rutas, asignación de parámetros de Calidad de Servicio, asignación de Ancho de banda, y encriptación.

La Contabilización se refiere al seguimiento del consumo de los recursos de red por los usuarios. Esta información puede usarse posteriormente para la administración, planificación, facturación, u otros propósitos. La contabilización en tiempo real es aquella en la que los datos generados se entregan al mismo tiempo que se produce el consumo de los recursos. En contraposición la contabilización por lotes (en inglés "batch accounting") consiste en la grabación de los datos de consumo para su entrega en algún momento posterior. La información típica que un proceso de contabilización registra es la identidad del usuario, el tipo de servicio que se le proporciona, cuándo comenzó a usarlo, y cuándo terminó.

AAA es una propiedad que todo sistema seguro debe contemplar; sin embargo, no ocurre de este modo. Muchas personas piensan en AAA sólo para conexiones y accesos remotos; no obstante, es posible implementarla a nivel local. De hecho, sistemas operativos como Unix y Windows poseen los elementos

necesarios para implementar estas características. Es deber de cada administrador configurar los sistemas de manera que cuenten con las tres AAA.

La familia de los protocolos AAA incluye al TACACS, TACACS+, DIAMETER y RADIUS. Además se apoyan para su desempeño en otros protocolos como el PPP[2] (Point-to-Point Protocol), EAP[3] (Extensible Authentication Protocol) y LDAP[4] (Lightweight Directory Access Protocol).

1.3 Aplicación del protocolo AAA

Las aplicaciones de la AAA pueden ser cualquiera que requiera acceso a una red o un servicio de red.

Aunque la mayoría de las implementaciones utilizan el paso por contraseña también existen otras que llevan a cabo el proceso completo como pueden ser los accesos remotos a una red wireless o una red VoIP, también las redes móviles celulares como GSM.

Por ejemplo un usuario conectándose remotamente a una Red Privada Virtual (Virtual Private Network, VPN) mediante un acceso Dial-up.

Una conexión a una red inalámbrica corporativa donde los usuarios tienen que estar registrados en las bases de empleados de la organización.

Las redes VoIP requieren del SIP (Session Initiation Protocol) este es utilizado para verificar la cuenta tras cada llamada que se realiza.

Otra aplicación puede ser la de invitado, donde solo ciertos privilegios deben ser entregados de manera temporal donde se conceden dependiendo del tiempo de inicio, y este requiere de un repositorio de invitados donde cada uno se identifica para conocer su estado actual y de esta manera determinar cuándo se debe cerrar la sesión.

---

[2] Protocolo Punto a Punto, asociado a la pila TCP/IP de uso en Internet.
[3] Protocolo de Autenticación Extensible.
[4] Protocolo Ligero de Acceso a Directorios.

También es utilizada para convivencia de dos diferentes tipos de usuarios aquellos que son invitados y los permanentes los cuales pueden estar identificados por sus roles dentro de la red.

La mayoría de las aplicaciones pueden utilizar diferentes protocolos de comunicación así como diferentes métodos de aplicación del AAA. Existen aplicaciones que pueden darse a través de los proveedores de servicios de internet, como pueden ser las conexiones Dial-up (marcado telefónico), DSL (Digital Subscriber Line), cable, 3G (Tercera Generación en telefonía móvil), wireless (comunicación inalámbrica) o metro wireless.

1.4 Familia de protocolos AAA

1.4.1 Protocolo TACACS

TACACS (acrónimo de Terminal Access Controller Access Control System , en inglés, y en español "Sistema de Control de Acceso mediante Control del Acceso desde Terminales") es un protocolo de autenticación remota que se usa para comunicarse con un servidor de autenticación comúnmente usado en redes Unix.

TACACS permite a un servidor de acceso remoto comunicarse con un servidor de autenticación para determinar si el usuario tiene acceso a la red. TACACS está documentado en el RFC 1492.

Protocolo de autenticación, que suministra autenticación de acceso remoto y servicios relacionados, por ejemplo, registro de eventos. Las contraseñas de usuario se administran en una base de datos central en lugar de administrarse en routers individuales, suministrando una solución de seguridad de red fácilmente escalable.

1.4.2 Protocolo TACACS+

TACACS+ (acrónimo en inglés de Terminal Access Controller Access Control System, Sistema de Control de Acceso del Controlador de Acceso a Terminales) es un protocolo de autenticación remota que se usa

para gestionar el acceso (proporciona servicios separados de autenticación, autorización y registro) a servidores y dispositivos de comunicaciones. Suministra soporte adicional para autenticación, autorización y contabilidad.

Es un protocolo propiedad de CISCO, que sustituye los protocolos TACACS y XTACACS, que sólo proporcionaban autenticación. Cisco añadió seguridad al estándar y la posibilidad de dividir el servidor AAA en tres servidores por separado. Debido a que es propiedad de Cisco, el estándar está perdiendo popularidad entre los proveedores. Es un protocolo de tipo cliente/servidor, donde un cliente (Network Access Server, NAS) envía una petición que es atendida por un servidor AAA. El protocolo está basado en TCP.

1.4.3 Protocolo DIAMETER

Diseñado para ser usado como protocolo AAA entre un Proveedor de Servicios de Internet (Internet Service Provider, ISP) y redes corporativas, es compatible con RADIUS y está fundado en el uso de (Attribute Value Pairs, AVP). Un AVP consta de tres campos: el código, la longitud y los datos. El protocolo nunca es usado como tal; sólo ofrece los requerimientos mínimos, requiere el uso de extensiones que son específicos a la aplicación. Existen extensiones para IP Móvil. Se considera un protocolo tipo peer-to-peer, en el sentido de que cualquier nodo puede iniciar una petición.

Este protocolo surgió como una solución de autenticación para redes implementadas bajo el protocolo PPP. No obstante, guarda diferencias marcadas que pretenden principalmente brindar extensamente servicios AAA. Dentro de sus facilidades más sobresalientes se encuentra la entrega y manejo de AVP’s, es decir, unidades de información AAA, unidades para negociación de recursos, unidades para notificaciones de errores, unidades de extensibilidad del protocolo a través de la adición de AVP’s y prestación de servicios básicos para aplicaciones, como por ejemplo, el manejo de sesiones de usuario.

El protocolo lleva a cabo la entrega de información por medio de los AVP’s, los cuales son adicionados a los mensajes DIAMETER dependiendo del tipo de solicitud requerida como, por ejemplo, el transporte de la información de autenticación del usuario, con el fin de realizar la búsqueda y verificación de éste en el

servidor DIAMETER. Adicionalmente, se considera el transporte de la información de autorización entre clientes y servidores para conceder la solicitud de acceso al usuario y el intercambio de información sobre utilización de recursos, con el fin de desempeñar tareas de auditoría, tales como mantenimiento y gestión de las sesiones, capacidad de los enlaces, registro de errores, etc.

Por último, se define la configuración de una jerarquía DIAMETER para llevar a cabo funciones de relevo de servidores para disponibilidad de servicios, resolución de solicitudes descentralizadas y redireccionamiento de mensajes. El funcionamiento del protocolo DIAMETER cumple con las mismas condiciones que el protocolo RADIUS, en cuanto al número y tipo de mensajes. Sin embargo, sus componentes de red brindan algunas características adicionales que demuestran serias diferencias. En primer lugar, el cliente DIAMETER es, por definición, un dispositivo que se encarga de administrar el control de acceso a un segmento de red específico, basado en el soporte y reconocimiento de las aplicaciones designadas para la red inalámbrica que opera bajo el protocolo DIAMETER; este cliente es el mismo punto de acceso. De igual forma es denominado este dispositivo en el protocolo RADIUS, sin embargo se limita tan sólo a retransmitir las solicitudes y respuestas desde y hacia los extremos de red.

Por su parte, el servidor DIAMETER es el dispositivo que se encarga de administrar y atender todas las solicitudes para la autenticación, autorización y auditoría, procedentes desde cualquier usuario en un dominio específico, lo cual ofrece mayores facilidades en la atención de requerimientos originados en redes remotas, mediante la ejecución de agentes, mientras RADIUS prescinde por completo de estas entidades.

Por lo tanto, es necesario que todo nombre de usuario este separado por el carácter “@” del nombre del dominio, con el fin de determinar si dicha solicitud puede ser atendida localmente o debe ser redireccionada o enrutada. RADIUS también requiere de este formato de nombre de dominio para poner en funcionamiento a sus servidores remotos, aunque la ventaja de DIAMETER es que este parámetro puede ser enviado por piggybacking en el campo para el nombre del DNS, en el mensaje de información sobre la ubicación remota, lo cual contribuye en la disminución del número de paquetes necesarios para el intercambio de datos.

Finalmente, los agentes DIAMETER para el redireccionamiento, relevo, Proxy y traducción, se encargan de manera respectiva o conjunta de la distribución de sistemas de administración en grupos para las funciones de asociación segura. Dentro de esta función se encuentran las tareas de concentración de solicitudes desde varios puntos de acceso inalámbricos localizados en la misma área o distribuidos, el procesamiento de valor agregado para ciertas solicitudes o respuestas, la distribución balanceada del tráfico, la organización de las solicitudes hacia las diferentes entidades autenticadoras cuando algunas redes de naturaleza más compleja pueden necesitarlo y mantener un registro salto por salto de las transacciones. Esto se hace con el fin de llevar un completo monitoreo de los eventos y transacciones que ocurren dentro de la red y bajo el protocolo.

El protocolo DIAMETER provee las siguientes facilidades:

- Entrega AVP's.
- Capacidad de negociación.
- Notificación de errores.
- Posibilidad de expansión, al poder agregar nuevos comandos y AVP's.
- Servicios básicos para aplicaciones, como pueden ser manejo de sesiones y contabilidad.

Actualmente, el protocolo DIAMETER no posee un desarrollo comercial muy avanzado, dada su novedad y el trabajo implícito que involucraría la transacción de servicios soportados por el protocolo RADIUS en redes de corto o amplio cubrimiento. Debido a esto muchas de las pruebas realizadas no pueden ser analizadas desde un punto de vista comparativo, ya que el paquete cliente/servidor DIAMETER disponible actualmente es un proyecto de código abierto, conocido como OpenDiameter, el cual a pesar de llevar una evolución ciertamente avanzada, son varias las deficiencias que aún guarda, como por ejemplo la ausencia de agentes Proxy, agentes de relevo, agentes de redireccionamiento, agentes de traducción y aplicaciones de movilidad.

El mecanismo EAP-TTLS se aplica de la misma forma para el protocolo DIAMETER, la docena de mensajes intercambiados llevan a cabo una transacción idéntica en el reconocimiento de certificados y credenciales, pero con la diferencia de emplear como algoritmo interno de cifrado al mecanismo EAP-MD5, ya que la aplicación OpenDiameter utilizada pertenece a un proyecto de arquitectura abierta, que reemplaza el método MS-CHAPv2 de Microsoft con MD5. Además los atributos intercambiados de cada

paquete DIAMETER evidentemente son diferentes para cada mensaje, llevando a encontrar las mayores diferencias en los valores de cabecera de AVP's y datos efectivos.

Finalmente, en el segundo de los mecanismos de autenticación implementados, el esquema PEAP con EAP-MD5 como algoritmo interno de cifrado para el envío de las credenciales de usuario, los mensajes intercambiados entre el usuario y el servidor son veinte en total.

1.4.4 Protocolo RADIUS

Surgió inicialmente como una solución para la administración en el control de acceso para usuarios que soportaban su conexión mediante enlaces seriales y módems, facilitando el control y supervisión de la seguridad, la autorización, la auditoría, verificación de nombres de usuarios y contraseñas, así como una detallada información de configuración sobre el tipo de servicio que se pretende entregar al usuario. Los elementos característicos que posee RADIUS le han permitido guardar un alto grado de compatibilidad con la arquitectura dispuesta por las redes inalámbricas IEEE 802.11, una razón primordial por la cual es éste el servidor recomendado, según la norma, para prestar los servicios de autenticación en redes inalámbricas.

El protocolo RADIUS sigue un modelo cliente/servidor, donde el papel de servidor es desempeñado por RADIUS, y un elemento de red designado como NAS (Network Access Server), toma la función de cliente de RADIUS; el NAS tiene la responsabilidad de servir como puente o mediador entre los mensajes entrantes y salientes desde y hacia el servidor, es decir, se encarga de retransmitir las solicitudes de conexión, autenticación de usuarios y en general toda la información necesaria para el usuario. RADIUS utiliza una sola base de datos, donde se encuentra almacenada toda la información de autenticación.

Se aclara que el NAS no solicita servicios ni acepta respuestas a los servicios solicitados. En el escenario de una red inalámbrica IEEE 802.11, el punto de acceso toma el papel de NAS. Las transacciones realizadas entre el cliente y el servidor RADIUS son autenticadas mediante la utilización de un secreto compartido, que nunca viaja por la red, además del intercambio entre estos dos puntos de una serie de contraseñas de usuarios, con el fin de minimizar la captura de la contraseña verdadera por parte de algún

intruso en la red. En el escenario de una red inalámbrica IEEE 802.11 la administración de llaves, con el fin de garantizar la confidencialidad de los mensajes, se hace mediante un sistema de derivación jerárquico de llaves que autentican la información por cada intercambio de mensaje, dicho sistema se describe en el protocolo IEEE 802.11i. En cualquier escenario donde se encuentre implementado RADIUS, todo usuario debe presentar ante el cliente o NAS una información de autenticación, dada por un nombre de usuario y una contraseña. Además se requiere de un protocolo de enlace de datos que soporte dentro de sus tramas información de autenticación.

Una vez el usuario ha presentado la información de autenticación ante el cliente, éste crea una Solicitud-de-Acceso o Access-Request ante el servidor RADIUS; la solicitud consta, como base, del nombre y contraseña del usuario, de la identificación del cliente y del puerto por el cual el usuario está accediendo. El mensaje de Solicitud-de-Acceso es enviado al servidor RADIUS el cual, tan pronto recibe el mensaje, inicia la validación en primera instancia del cliente; si el servidor no tiene un secreto compartido para la solicitud procedente desde el cliente, el paquete debe ser silenciosamente descartado. Si por el contrario el cliente es correctamente validado, el servidor RADIUS busca en la base de datos de usuarios la correspondencia entre el nombre de usuario y la contraseña especificada en la Solicitud-de-Acceso. La base de datos en el servidor contiene adicionalmente una lista de requisitos que deben ser conocidos y especificados inicialmente para permitir el acceso del usuario; por ejemplo en el estándar IEEE 802.11i la configuración de los perfiles de usuarios están considerados de tal forma que el usuario sólo pueda acceder por un puerto específico y a través de un único cliente, mediante la implementación del protocolo IEEE 802.1X.

El desempeño del protocolo RADIUS se mide de acuerdo con el comportamiento demostrado en la autenticación de usuarios mediante los mecanismos EAP-TTLS (Tunneled Transport Layer Security) y PEAP (Protected EAP), con algoritmos internos de cifrado PAP (Password Authentication Protocol), MS-CHAPv2 (Microsoft Challenge Handshake Authentication Protocol) y EAP-MS-CHAPv2, para que finalmente con estadísticas tales como la media, el coeficiente de variación y la desviación estándar, se establezcan las fortalezas y debilidades internas en el comportamiento del protocolo, al igual que las ventajas y desventajas comparativas de uno contra otro.

Los mecanismos de autenticación analizados son en primer lugar EAP-TTLS con algoritmo interno de cifrado PAP y posteriormente con MS-CHAPv2, los cuales son versiones populares y de uso más generalizado para el intercambio de credenciales. El segundo mecanismo es PEAP con algoritmo interno de cifrado EAP-MS-CHAPv2, el cual es el único mecanismo de autenticación dispuesto para plataformas Windows XP Service Pack 2.

Durante la autenticación EAP-TTLS con PAP, el primer paquete Access-Request transporta en el atributo EAP-Message la identidad del equipo móvil, es decir, el nombre de usuario; no obstante para la autenticación EAP-TTLS es posible ocultar el nombre verdadero con un nombre falso programado por el cliente, de tal forma que el analizador de protocolo muestre el nombre falso de usuario en el atributo User-Name.

Si el nombre es verificado satisfactoriamente en el servidor, éste envía un paquete Access-Challenge definiendo el mecanismo que propone para la autenticación del usuario, en este caso TTLS. Si el mecanismo es soportado por el cliente, se inicia con el nuevo mensaje Access-Request la sesión SSL (Secure Socket Layer) implementando el protocolo TLS, enviando internamente el paquete de saludo al cliente (Hello Client) con las llaves de cifrado y métodos de compresión. El servidor en respuesta envía su certificado digital que permite al cliente verificar si es un servidor de confianza, junto con el certificado viaja la llave pública para el encriptado de los mensajes siguientes. El cliente con el envío de un Access-Request confirma la instalación de la llave pública en su equipo local, para que a continuación el servidor prepare un desafío encriptado con la llave ya presente en ambos extremos, validando de esta manera si la llave fue propiamente instalada en el cliente; de ser así éste se dispone a transmitir su llave pública de cifrado hacia el servidor, quien confirma su instalación con el envío de un mensaje encriptado. Si el mensaje es correcto, el usuario responde con un mensaje de aplicación que finaliza el establecimiento del túnel virtual en el canal de comunicación y el servidor puede emitir finalmente su mensaje Access-Accept con las llaves de encripción de datos, derivadas de la contraseña PAP y almacenadas en los atributos Vendor-Specific.

En la autenticación EAP- TTLS con MS-CHAPv2 se necesita intercambiar doce mensajes entre el servidor y el usuario para llevar a cabo de forma satisfactoria el proceso de autenticación, desde la primera solicitud de acceso hasta la respuesta de aceptación. El método de cifrado interno hereda el

funcionamiento del tradicional CHAP del protocolo PPP con una secuencia binaria de 16 bytes, generada a partir de la contraseña de usuario y los campos propios del paquete RADIUS; sin embargo Microsoft, quien posee la patente de este método, reforzó la generación de los datos cifrados con un módulo muy similar al trabajado por el método MD5 (Message Digest 5). Adicionalmente, la razón por la cual se necesita de dos paquetes más para completar la autenticación, es la utilización de dos llaves CHAP para ser instaladas en el equipo cliente de tal forma que al paquete con el mensaje de aplicación enviado por el cliente, el servidor responda con uno igual, pero utilizando la segunda llave CHAP. Si el mensaje es correctamente verificado por el usuario, el servidor envía un paquete Access-Request confirmando que el intercambio de las herramientas de cifrado funcionan correctamente; en este momento el servidor está listo para emitir su mensaje de aceptación.

El mecanismo PEAP es un desarrollo de Microsoft, por lo tanto sólo es compatible con plataformas de este tipo, incluso las opciones disponibles para algoritmos internos de cifrado son escasas, la primera de ellas es EAP-MS-CHAPv2 y la segunda es EAP-GTC (EAP Generis Tocken Card) que sólo funciona con equipos inalámbricos de la compañía Cisco Systems. Este mecanismo de autenticación intercambia una totalidad de veinte mensajes antes de completar el proceso de autorización para el acceso del usuario, donde las doce primeras transacciones cumplen los mismos objetivos de la autenticación EAP-TTLS con MS-CHAPv2, es decir, el intercambio del certificado digital del servidor, las credenciales del usuario y las llaves derivadas del algoritmo de cifrado; los paquetes adicionales son mensajes de aplicación que comprueban mediante desafío y respuesta la instalación de llaves MS-CHAP, tanto en el equipo servidor como en la estación del cliente.

Para proveer con seguridad a los mensajes RADIUS, el cliente y el servidor son configurados con una clave secreta común.

Conclusiones

Se ha realizado un estudio de los modelos OSI y TCP/IP, se compararon marcando sus diferencias y semejanzas, se analizaron las capas de ambos modelos y de esta forma se evidenciaron las ventajas y las utilidades que trae el uso de los modelos en capas, además que permite adquirir un mejor conocimiento de cómo se establece la comunicación entre los diferentes dispositivos de red, profundizando en el modelo TCP/IP que utilizaremos en nuestra investigación. Para conocer el funcionamiento de los servicios AAA se estudiaron sus funciones así como la de los principales protocolos de esta familia. Se profundizó en los protocolos DIAMETER y RADIUS que son nuestro objeto de estudio.

CAPÍTULO 2: DESCRIPCION DEL PROCEDIMIENTO PARA ANALIZAR LOS PROTOCOLOS AAA.

En este capítulo se presenta la descripción del procedimiento, para ello se enumeran varios pasos necesarios para una mejor definición del mismo. La exposición de cada uno de los protocolos que intervienen en las capas del modelo TCP/IP es un aspecto muy importante para lograr obtener un buen procedimiento de cada uno de los protocolos que intervienen en una transferencia de datos por la red.

2.1 Modelo TCP/IP

Con el fin de mejorar los servicios de transmisión y conmutación en una red, los diseñadores de internet reevaluaron las versiones iniciales de las redes conmutadas por paquetes. El objetivo perseguido por los diseñadores de internet es que en la red un paquete puede ser dividido en múltiples bloques de mensajes, donde los bloques individuales que contienen información de direccionamiento indican su punto de origen así como su destino final. Empleando la información de direccionamiento contenida en estos paquetes individuales se pueden enviar por la red utilizando diferentes vías o rutas así como reensamblar estos bloques conformando el mensaje original al llegar a su destino.

Fig. 1 Modelo TCP/IP

Capa 1 - Acceso al Medio o la Red, asimilable a la capa 1 (física) y 2 (enlace de datos) del modelo OSI. Ofrece la capacidad de acceder a cualquier red física, es decir, brinda los recursos que se deben implementar para transmitir datos a través de la red. Algunos de los protocolos que interactúan en esta capa son Ethernet, Token Ring.

Capa 2 - Internet, asimilable a la capa 3 (red) del modelo OSI. La capa de Internet es una de las capas más importantes. En ella se definen las direcciones IP, se anidan los datos pertenecientes a las capas de transporte y aplicación y se incluyen los elementos necesarios para realizar el correcto enrutamiento de los paquetes. A continuación se muestran varios protocolos que intervienen en esta capa, IP, RARP, IGMP, ARP, ICMP.

Capa 3 - Transporte, asimilable a la capa 4 (transporte) del modelo OSI. Permite que las aplicaciones que se ejecutan en equipos remotos puedan comunicarse. Es por ello que se ha implementado un sistema de numeración para poder asociar un tipo de aplicación con un tipo de datos. Estos identificadores se denominan puertos.

La capa de transporte contiene dos protocolos que permiten que dos aplicaciones puedan intercambiar datos independientemente del tipo de red (es decir, independientemente de las capas inferiores). Estos dos protocolos son TCP, UDP.

Capa 4 - Aplicación, la capa de aplicación debe incluir los detalles de las capas de sesión y presentación OSI. La capa de aplicación maneja aspectos de representación, codificación y control de diálogo. Algunos de los Protocolos que contiene esta capa son HTTP, SMTP, TELNET, FTP, DNS, AAA.

2.1.1 Protocolo Ethernet (Capa Acceso a la Red)

El comité de estándares para Redes Metropolitanas y locales del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) publicó los estándares para las Redes de Área Local (Local Area Network, LAN). Estos estándares comienzan con el número 802. El estándar para Ethernet es el 802.3. Ethernet opera en la capa inferior del modelo TCP/IP, denominada capa de Acceso a la red.

Ethernet implica señales, streams de bits que se transportan en los medios, componentes físicos que transmiten las señales a los medios y distintas topologías. Ethernet tiene un papel clave en la comunicación que se produce entre los dispositivos.

La encapsulación de datos posibilita:

- Delimitación de trama.
- Direccionamiento.
- Detección de errores.

El proceso de encapsulación viabiliza el direccionamiento. Cada encabezado Ethernet agregado a la trama contiene la dirección física (dirección MAC) que permite que la trama se envíe a un nodo de destino. Una función adicional de la encapsulación de datos es la detección de errores. Cada trama de Ethernet contiene un tráiler con una Comprobación Cíclica de Redundancia (CRC) de los contenidos de la trama. Una vez que se recibe una trama, el nodo receptor crea una CRC para compararla con la de la trama. Si estos dos cálculos de CRC coinciden, puede asumirse que la trama se recibió sin errores.

La mayor parte del tráfico en Internet se origina y termina en conexiones de Ethernet. Ethernet ha evolucionado para satisfacer la creciente demanda de LAN de alta velocidad. Cuando se introdujo el medio de fibra óptica, Ethernet se adaptó a esta nueva tecnología para aprovechar el mayor ancho de banda y el menor índice de error que ofrece la fibra.

El éxito de Ethernet se debe a los siguientes factores:

- Simplicidad y facilidad de mantenimiento
- Capacidad para incorporar nuevas tecnologías
- Confiabilidad
- Bajo costo de instalación y de actualización.

En las redes actuales, la Ethernet utiliza cables de cobre UTP (Unshielded Twisted Pair, par trenzado no apantallado) y fibra óptica para interconectar dispositivos de red a través de dispositivos intermediarios como hubs y switches. Dada la diversidad de tipos de medios que Ethernet admite, la estructura de la trama de Ethernet permanece constante a través de todas sus implementaciones físicas. Es por esta razón que puede evolucionar hasta cumplir con los requisitos de red actuales.

La estructura de la trama de Ethernet agrega encabezados y tráilers a la PDU de Capa Internet del modelo TCP/IP para encapsular el mensaje que se envía. Tanto el encabezado como el tráiler de Ethernet tienen varias secciones de información que el protocolo Ethernet utiliza. Cada sección de la trama se denomina campo. Hay dos estilos de tramas de Ethernet: el IEEE 802.3 (original) y el IEEE 802.3

revisado (Ethernet) (Ver anexo 2). Las diferencias entre los estilos de tramas son mínimas. La diferencia más significativa entre el IEEE 802.3 (original) y el IEEE 802.3 revisado es el agregado de un Delimitador de Inicio de Trama (SFD) y un pequeño cambio en el campo Tipo que incluye la longitud de la trama.

El estándar Ethernet original definió el tamaño mínimo de trama en 64 bytes y el tamaño máximo de trama en 1518 bytes. Esto incluye todos los bytes del campo Dirección MAC de destino a través del campo Secuencia de Verificación de Trama (Frame Check Sequence, FCS). Los campos Preámbulo y Delimitador de Inicio de Trama no se incluyen en la descripción del tamaño de una trama. El estándar IEEE 802.3ac, publicado en 1998, amplió el tamaño de trama máximo permitido a 1522 bytes.

Las direcciones de la Capa de Internet (Capa 2), como por ejemplo, las direcciones IPv4, brindan el direccionamiento general y local que se comprende tanto en el origen como en el destino. Para llegar a su último destino, un paquete transporta la dirección de destino de Capa 2 desde su origen. Cuando el dispositivo de origen reenvía el mensaje a una red Ethernet, se adjunta la información del encabezado dentro de la dirección MAC. El dispositivo de origen envía los datos a través de la red.

2.1.2 Protocolo IP (Capa Internet)

La Capa de Internet o Capa 2 del Modelo TCP/IP, provee servicios para intercambiar secciones de datos individuales a través de la red entre dispositivos finales identificados. Para realizar este transporte de un extremo a otro extremo, esta capa emplea procesos como: el direccionamiento, encapsulamiento, enrutamiento y desencapsulamiento. Los protocolos especifican la estructura y el procesamiento de los paquetes utilizados para llevar los datos desde un host hasta otro. Operar ignorando los datos de aplicación llevados en cada paquete permite a la capa de Internet llevar paquetes para múltiples tipos de comunicaciones entre hosts múltiples.

- Protocolos de la capa de Internet

Los servicios de capa de Internet implementados por el conjunto de protocolos TCP/IP son el Protocolo de Internet (IP). La versión 4 de IP (IPv4) es la versión de IP más ampliamente utilizada. Es el único protocolo

de esta capa que se utiliza para llevar datos de usuario a través de Internet y se aborda en este trabajo investigativo, los demás protocolos no son estudiados a profundidad. El IPv6 opera junto con IPv4 y este último puede ser reemplazado por el primero en el futuro.

El Protocolo de Internet (Internet Protocol, IP) fue diseñado como un protocolo con bajo costo. Provee sólo las funciones necesarias para enviar un paquete desde un origen a un destino a través de un sistema interconectado de redes. El protocolo no fue diseñado para rastrear ni administrar el flujo de paquetes. Estas funciones son realizadas por otros protocolos en otras capas.

❖ Características básicas de IPv4

✓ Sin conexión

Los paquetes IP se envían sin notificar al host final que están llegando. No requiere un intercambio inicial de información de control para establecer una conexión de extremo a extremo antes de que los paquetes sean enviados, ni requiere campos adicionales en el encabezado de la PDU para mantener esta conexión. Este proceso reduce en gran medida la sobrecarga del IP. La entrega del paquete sin conexión puede hacer que los paquetes lleguen al destino fuera de secuencia. Si los paquetes que no funcionan o están perdidos crean problemas para la aplicación que usa los datos, luego los servicios de las capas superiores tienen que resolver estas cuestiones.

✓ Servicio de mejor intento (no confiable)

IP a menudo se lo considera un protocolo no confiable. No confiable en este contexto no significa que el IP funciona adecuadamente algunas veces y no funciona bien en otras oportunidades. Tampoco significa que no es adecuado como protocolo de comunicaciones de datos. No confiable significa simplemente que IP no tiene la capacidad de administrar ni recuperar paquetes no entregados o corruptos. Como los protocolos en otras capas pueden administrar la confiabilidad, se le permite a IP funcionar con mucha eficiencia en la capa de Internet.

✓ Independiente de los medios

Es responsabilidad de la capa de Acceso al Medio del Modelo TCP/IP tomar un paquete IP y prepararlo para transmitirlo por el medio de comunicación. Esto significa que el transporte de paquetes IP no está limitado a un medio en particular, cualquier paquete IP individual puede ser comunicado eléctricamente por cable, como señales ópticas por fibra, o sin cables como las señales de radio.

IPv4 encapsula o empaqueta el datagrama o segmento de la capa de Transporte para que la red pueda entregarlo a su host de destino. La encapsulación de IPv4 permanece en su lugar desde el momento en que el paquete deja la capa de Internet del host de origen hasta que llega a la misma capa en el host de destino.

El proceso de encapsular datos por capas permite que los servicios en las diferentes capas se desarrollen y escalen sin afectar otras capas. Esto significa que los segmentos de la capa de Transporte pueden ser empaquetados fácilmente por los protocolos de la capa de Internet existentes, como IPv4 e IPv6, o por cualquier protocolo nuevo que pueda desarrollarse en el futuro.

2.1.3 Protocolo TCP (Capa de Transporte)

Cuando hablamos de TCP nos estamos refiriendo al acrónimo de Tansmission Control Protocol por sus siglas en inglés, que en español significan Protocolo de Control de Transmisión. Es el protocolo de transporte que administra las conversaciones individuales entre servidores web y clientes web. TCP divide en pequeñas partes nombradas segmentos, los mensajes http para enviarlos al cliente de destino. Además se encarga de controlar el tamaño y los intervalos a los que se intercambian los mensajes entre el servidor y el cliente. Es un protocolo orientado a la conexión, descrito en la RFC 793. TCP incurre en el uso adicional de recursos para agregar funciones. Las funciones adicionales especificadas por TCP están en el mismo orden de entrega, son de entrega confiable y de control de flujo. Los segmentos de TCP son de 20 bytes de carga en el encabezado, que encapsulan los datos de la capa de aplicación, mientras que los segmentos de UDP tienen solamente 8 bytes de carga. (Ver anexo 1)

❖ Protocolo TCP: Conversaciones confiables

En TCP, cada encabezado de segmento contiene un número de secuencia. Este número de secuencia permite que las funciones de la capa de Transporte del host de destino reensamblen los segmentos en el mismo orden en el que fueron transmitidos, lo que asegura que los datos lleguen a la aplicación de destino de la misma forma que fueron enviados. Mediante sesiones orientadas a la conexión se lleva a cabo la confiabilidad de la comunicación TCP. Antes de que un host que utiliza TCP envíe datos a otro host, la capa de Transporte inicia un proceso para crear una conexión con el destino, lo que permite entre los hosts un rastreo de sesión. A través de este proceso se asegura que la comunicación sea del conocimiento de cada hosts involucrado y se prepare para la misma. Además, se demanda para una conversación TCP completa, una sesión entre los hosts en ambas direcciones.

Cuando la sesión queda establecida, el destino envía acuses de recibo al origen por los segmentos que recibe. Estos acuses de recibo forman la base de la confiabilidad dentro de la sesión TCP. Cuando el origen recibe un acuse de recibo, reconoce que los datos se han entregado con éxito y puede dejar de rastrearlos. Si el origen no recibe el acuse de recibo dentro de un tiempo predeterminado, retransmite esos datos al destino. Parte de la carga adicional que genera el uso de TCP es el tráfico de red generado por los acuses de recibo y las retransmisiones. El establecimiento de las sesiones genera cargas en forma de segmentos adicionales intercambiados. También existen cargas adicionales en los hosts individuales, generadas por la necesidad de mantener un seguimiento de los segmentos que esperan acuse de recibo y por el proceso de retransmisión. Al finalizar la comunicación, las sesiones se cierran y termina la conexión. Cada campo con su función específica dentro del segmento TCP son los encargados de lograr esta confiabilidad. (Ver anexo 2)

El protocolo TCP es utilizado en aplicaciones de transferencia de archivos, correo electrónico y exploradores web.

❖ Procesos del servidor TCP

El administrador del sistema configura cada proceso de aplicación que se ejecuta en el servidor para utilizar un número de puerto, de forma predeterminada o manual. Un servidor individual no puede tener

dos servicios asignados al mismo número de puerto dentro de los mismos servicios de la capa de Transporte. Un host que ejecuta una aplicación de servidor Web y una de transferencia de archivos no puede configurar ambas para utilizar el mismo puerto (por ejemplo, el puerto 8080). Cuando una aplicación de servidor activa se asigna a un puerto específico, este puerto se considera "abierto" para el servidor. Esto significa que la capa de Transporte acepta y procesa segmentos direccionados a ese puerto. Toda solicitud entrante de un cliente direccionada al socket correcto es aceptada y los datos se envían a la aplicación del servidor. Pueden existir varios puertos simultáneos abiertos en un servidor, uno para cada aplicación de servidor activa. Es común que un servidor provea más de un servicio, como un servidor Web y un servidor FTP, al mismo tiempo.

Para lograr una mejor seguridad en los servidores seria bueno la restricción del acceso al servidor a sólo aquellos puertos asociados con los servicios y aplicaciones accesibles a solicitantes autorizados.

- Establecimiento y finalización de la conexión TCP

En una conexión se establecen dos sesiones. Para llevar a cabo una conexión los hosts realizan un intercambio de señales de tres vías.

Primeramente en el enlace de tres vías, se verifica que el dispositivo de destino esté presente en la red. En segundo lugar, se comprueba que el dispositivo de destino tenga un servicio activo y esté aceptando las peticiones en el número de puerto de destino que el cliente que lo inicia intenta usar para la sesión. Y por último, se informa al dispositivo de destino que el cliente de origen intenta establecer una sesión de comunicación en ese número de puerto.

En las conexiones TCP, el host que brinde el servicio como cliente inicia la sesión al servidor. Los siguientes tres pasos definen el establecimiento de una conexión TCP:

1. El cliente que inicia la conexión envía un segmento que contiene un valor de secuencia inicial, que actúa como solicitud para el servidor para comenzar una sesión de comunicación.
2. El servidor responde con un segmento que contiene un valor de reconocimiento igual al valor de secuencia recibido más 1, además de su propio valor de secuencia de sincronización. El valor es uno

mayor que el número de secuencia porque el ACK es siempre el próximo Byte u Octeto esperado. Este valor de reconocimiento permite al cliente unir la respuesta al segmento original que fue enviado al servidor.

3. El cliente que inicia la conexión responde con un valor de reconocimiento igual al valor de secuencia que recibió más uno. Esto completa el proceso de establecimiento de la conexión.

Es importante observar los distintos valores que intercambian los dos hosts, para poder entender el proceso de enlace de tres vías. El encabezado del segmento TCP incluye seis campos de 1 bit que contienen información de control utilizada para gestionar los procesos de TCP. Estos campos se nombran a continuación:

URG: Urgente campo de señalizador significativo

ACK: Campo significativo de acuse de recibo

PSH: Función de empuje

RST: Reconfiguración de la conexión

SYN: Sincronizar números de secuencia

FIN: No hay más datos desde el emisor

A estos campos se los denomina señaladores porque el valor de uno de estos campos es sólo de un bit, entonces tiene sólo dos valores: 1 ó 0. Si el valor del bit se establece en 1, indica la información de control que contiene el segmento.

❖ Protocolo TCP de enlace de tres vías

El funcionamiento detallado del protocolo TCP de enlace de tres vías quedaría de la siguiente forma:

Paso 1

Un cliente TCP comienza el enlace de tres vías enviando un segmento con el señalizador de control SYN (Sincronizar números de secuencia) establecido, indicando un valor inicial en el campo de número de secuencia del encabezado. Este valor inicial para el número de secuencia, conocido como número de secuencia inicial (ISN), se elige de manera aleatoria y se utiliza para comenzar a rastrear el flujo de datos desde el cliente al servidor para esta sesión. El ISN en el encabezado de cada segmento se incrementa en uno por cada byte de datos enviados desde el cliente hacia el servidor mientras continúa la conversación de datos.

Paso 2

El servidor TCP necesita reconocer la recepción del segmento SYN del cliente para establecer la sesión de cliente a servidor. Para hacerlo, el servidor envía un segmento al cliente con el señalizador ACK establecido indicando que el número de acuse de recibo es significativo. Con este señalizador establecido en el segmento, el cliente interpreta esto como acuse de recibo de que el servidor ha recibido el SYN del cliente TCP.

El valor del número de campo del acuse de recibo es igual al número de secuencia inicial del cliente más 1. Esto establece una sesión desde el cliente al servidor. El señalizador ACK permanecerá establecido para mantener el equilibrio de la sesión. Es de interés recordar que la conversación entre el cliente y el servidor está compuesta en realidad por dos sesiones de una vía: una del cliente al servidor y la otra del servidor al cliente. En este segundo paso del enlace de tres vías, el servidor debe iniciar la respuesta desde el hasta el cliente. Para comenzar esta sesión, el servidor utiliza el señalizador SYN de la misma manera en que lo hizo el cliente. Establece el señalizador de control SYN en el encabezado para establecer una sesión del servidor al cliente. El señalizador SYN indica que el valor inicial del campo de número de secuencia se encuentra en el encabezado. Este valor se utilizará para rastrear el flujo de datos en esta sesión del servidor al cliente.

Paso 3

Por último, el cliente TCP responde con un segmento que contiene un ACK que actúa como respuesta al SYN de TCP enviado por el servidor. No existen datos de usuario en este segmento. El valor del campo número de acuse de recibo contiene uno más que el número de secuencia inicial recibido del servidor. Una vez establecidas ambas sesiones entre el cliente y el servidor, todos los segmentos adicionales que se intercambien en la comunicación tendrán establecido el señalizador ACK.

De la siguiente forma podría añadírsele seguridad a la red de datos:

- Denegando el establecimiento de sesiones TCP
- Sólo permitiendo sesiones para ser establecidas por servicios específicos
- Sólo permitiendo el tráfico como parte de sesiones ya establecidas

Existe la opción de que esta seguridad se implemente sólo para las sesiones seleccionadas o para todas las sesiones.

❖ Reensamblaje de segmentos TCP

Cuando los servicios envían datos utilizando TCP, los segmentos pueden llegar a destinos desordenados. Para que el receptor comprenda el mensaje original, los datos en estos segmentos se reensamblan en el orden original. Para lograr esto, se asignan números de secuencia en el encabezado de cada paquete.

Durante la configuración de la sesión, se establece un número de secuencia inicial (ISN). Este número de secuencia inicial representa el valor de inicio para los bytes de esta sesión que se transmitirán a la aplicación receptora. A medida que se transmiten los datos durante la sesión, el número de secuencia se incrementa en el número de bytes que se han transmitido. Este rastreo de bytes de datos permite que cada segmento se identifique y se envíe acuse de recibo de manera exclusiva. Se pueden identificar segmentos perdidos.

El proceso TCP receptor coloca los datos del segmento en un búfer de recepción. Los segmentos se colocan en el orden de número de secuencia adecuado y se pasa a la capa de Aplicación cuando son reensamblados. Todos los segmentos que llegan con números de secuencia no contiguos se mantienen para su procesamiento posterior. Luego, se procesan los segmentos cuando llegan con los bytes perdidos. (Ver anexo 3)

- Acuse de recibo de TCP con uso de ventanas

Confirmación de recepción de segmentos

Una de las funciones de TCP es asegurar que cada segmento llegue a su destino. Los servicios TCP en el host de destino envían a la aplicación de origen un acuse de recibo de los datos recibidos.

El número de secuencia y el número de acuse de recibo del encabezado del segmento se utilizan para confirmar la recepción de los bytes de datos contenidos en los segmentos. El número de secuencia es el número relativo de bytes que ha sido transmitido en esta sesión más 1 (que es el número del primer byte de datos en el segmento actual). TCP utiliza el número de reconocimiento en segmentos que se vuelven a enviar al origen para indicar el próximo byte de esta sesión que espera el receptor. Esto se llama acuse de recibo de expectativa.

Se le informa al origen que el destino ha recibido todos los bytes de datos, pero sin incluir, el byte especificado por el número de acuse de recibo. Se espera que el host emisor envíe un segmento que utiliza un número de secuencia igual al número de acuse de recibo. Es importante recordar que cada conexión se representa en realidad por dos sesiones de una vía. Los números de secuencia y de acuse de recibo se intercambian en ambas direcciones.

La cantidad de datos que un origen puede transmitir antes de que un acuse de recibo deba ser recibido se denomina tamaño de la ventana. El tamaño de la ventana es un campo en el encabezado TCP que permite la administración de datos perdidos y el control del flujo.

❖ Retransmisión de TCP

Manejo de la pérdida de segmentos

Por óptimo que sea el diseño de una red, siempre se producen pérdidas ocasionales de datos. Por lo tanto, TCP cuenta con métodos para gestionar dichas pérdidas de segmentos. Entre los mismos existe un mecanismo para retransmitir segmentos con datos no reconocidos.

Un servicio de host de destino que utiliza TCP, por lo general sólo reconoce datos para secuencias de bytes contiguas. Si uno o más segmentos se pierden, sólo se acusa recibo de los datos de los segmentos que completan el grupo.

Cuando TCP en el host de origen no recibe un acuse de recibo pasado un tiempo predeterminado, vuelve al último número de acuse de recibo que recibió y retransmite los datos a partir de éste. El proceso de retransmisión no es especificado por RFC, sino que depende de la implementación de TCP en particular.

Para una implementación de TCP típica, un host puede transmitir un segmento, colocar una copia del segmento en una cola de retransmisión e iniciar un temporizador. Cuando se recibe el acuse de recibo de los datos, se elimina el segmento de la cola. Si no se recibe el acuse de recibo antes de que el temporizador venza, el segmento es retransmitido.

Los hosts actuales también suelen emplear una función opcional llamada acuses de recibo selectivos. Si ambos hosts admiten el Acuse de recibo selectivo, es posible que el destino reconozca los bytes de segmentos discontinuos y el host sólo necesite retransmitir los datos perdidos.

❖ Control de congestión de TCP

Control del flujo

TCP también provee mecanismos para el control del flujo. El control del flujo contribuye con la confiabilidad de la transmisión TCP ajustando la tasa efectiva de flujo de datos entre los dos servicios de

la sesión. Cuando el origen advierte que se recibió la cantidad de datos especificados en los segmentos, puede continuar enviando más datos para esta sesión.

El campo Tamaño de la ventana en el encabezado TCP especifica la cantidad de datos que puede transmitirse antes de que se reciba el acuse de recibo. El tamaño de la ventana inicial se determina durante el comienzo de la sesión a través del enlace de tres vías.

El mecanismo de retroalimentación de TCP ajusta la tasa de transmisión de datos efectiva al flujo máximo que la red y el dispositivo de destino pueden soportar sin sufrir pérdidas. TCP intenta gestionar la tasa de transmisión de manera que todos los datos se reciban y se reduzcan las retransmisiones.

- Reducción del tamaño de la ventana

Otra forma de controlar el flujo de datos es utilizar tamaños dinámicos de ventana. Cuando los recursos de la red son limitados, TCP puede reducir el tamaño de la ventana para lograr que los segmentos recibidos sean reconocidos con mayor frecuencia. Esto disminuye de manera efectiva la tasa de transmisión, ya que el origen espera que los datos sean recibidos con más frecuencia.

El host receptor TCP envía el valor del tamaño de la ventana al TCP emisor para indicar el número de bytes que está preparado para recibir como parte de la sesión. Si el destino necesita disminuir la tasa de comunicación debido a limitaciones de memoria del búfer, puede enviar un valor de tamaño de la ventana menor al origen como parte de un acuse de recibo.

Después de períodos de transmisión sin pérdidas de datos o recursos limitados, el receptor comienza a aumentar el tamaño de la ventana. Esto reduce la sobrecarga de la red, ya que se requiere enviar menos acuses de recibo. El tamaño de la ventana continua aumentando hasta que haya pérdida de datos, lo que produce una disminución del tamaño de la ventana.

Estas disminuciones y aumentos dinámicos del tamaño de la ventana representan un proceso continuo en TCP, que determina el tamaño de la ventana óptimo para cada sesión TCP. En redes altamente eficientes,

los tamaños de la ventana pueden ser muy grandes porque no se pierden datos. En redes donde se está estresando la infraestructura subyacente, el tamaño de la ventana probablemente permanece pequeño.

2.1.4 Protocolo UDP (Capa de Transporte)

UDP es un protocolo simple que provee las funciones básicas de la capa de Transporte. Genera mucho menos sobrecarga que TCP, ya que no es orientado a la conexión y no cuenta con los sofisticados mecanismos de retransmisión, secuenciación y control del flujo. Esto no significa que las aplicaciones que utilizan UDP no sean confiables. Sólo quiere decir que estas funciones no son contempladas por el protocolo de la capa de Transporte y deben implementarse aparte, si fuera necesario.

Pese a que es relativamente baja la cantidad total de tráfico UDP que puede encontrarse en una red típica, entre los protocolos principales de la capa de Aplicación que utilizan UDP se incluyen:

- Sistema de Denominación de Dominio (Domain Name System, DNS)
- Protocolo Simple de Administración de Red (Simple Network Management Protocol, SNMP)
- Protocolo de Configuración Dinámica de Host (DHCP)
- Protocolo de Información de Enrutamiento (Routing Information Protocol, RIP)
- Protocolo Trivial de Transferencia de Archivos (Trivial File Transfer Protocol, TFTP)
- Juegos en línea.

Algunas aplicaciones como los juegos en línea o VoIP pueden tolerar algunas pérdidas de datos. Si estas aplicaciones utilizaran TCP, experimentarían largas demoras, ya que TCP detecta la pérdida de datos y los retransmite. Estas demoras serían más perjudiciales para la aplicación que las pequeñas pérdidas de datos. Algunas aplicaciones, como DNS, simplemente reintentan enviar la solicitud si no obtienen respuesta y, por lo tanto, no necesitan TCP para garantizar la entrega del mensaje. La baja sobrecarga de UDP lo hace deseable para dichas aplicaciones.

❖ Reensamblaje de datagramas de UDP

Ya que UDP opera sin conexión, las sesiones no se establecen antes de que se lleve a cabo la comunicación, como sucede con TCP. Se dice que UDP es basado en transacciones. En otras palabras, cuando una aplicación posee datos para enviar, simplemente los envía. Muchas aplicaciones que utilizan UDP envían pequeñas cantidades de datos que pueden ocupar un segmento. Sin embargo, algunas aplicaciones enviarán cantidades mayores de datos que deben dividirse en varios segmentos. La PDU de UDP se conoce como datagrama, pese a que los términos segmento y datagrama a veces se utilizan de manera indistinta para describir una PDU de la capa de Transporte.

Cuando se envían múltiples datagramas a un destino, los mismos pueden tomar rutas distintas y llegar en el orden incorrecto. UDP no mantiene un seguimiento de los números de secuencia de la manera en que lo hace TCP. UDP no puede reordenar los datagramas en el orden de la transmisión. Por lo tanto, UDP simplemente reensambla los datos en el orden en que se recibieron y los envía a la aplicación. Si la secuencia de los datos es importante para la aplicación, la misma deberá identificar la secuencia adecuada de datos y determinar cómo procesarlos.

❖ Procesos y solicitudes del servidor UDP

Al igual que las aplicaciones basadas en TCP, a las aplicaciones de servidor basadas en UDP se les asigna números de puertos bien conocidos o registrados. Cuando se ejecutan estas aplicaciones o procesos, aceptan los datos que coincidan con el número de puerto asignado. Cuando UDP recibe un datagrama destinado a uno de esos puertos, envía los datos de aplicación a la aplicación adecuada en base a su número de puerto.

❖ Procesos del cliente UDP

Como en TCP, la comunicación cliente/servidor se inicia por una aplicación cliente que solicita datos de un proceso del servidor. El proceso de cliente UDP selecciona al azar un número de puerto del rango dinámico de números de puerto y lo utiliza como puerto de origen para la conversación. El puerto de destino por lo general será el número de puerto bien conocido o registrado asignado al proceso del servidor. Los números de puerto de origen seleccionados al azar colaboran con la seguridad. Si existe un

patrón predecible para la selección del puerto de destino, un intruso puede simular el acceso a un cliente de manera más sencilla intentando conectarse al número de puerto que tenga mayor posibilidad de estar abierto. Ya que no se crean sesiones con UDP, tan pronto como los datos están listos para ser enviados y los puertos estén identificados, UDP puede formar el datagrama y enviarlo a la capa de Red para el direccionamiento y envío a la red. Es importante recordar que una vez que el cliente ha elegido los puertos de origen y destino, estos mismos puertos se utilizarán en el encabezado de todos los datagramas que se utilicen en la transacción. Para la devolución de datos del servidor al cliente, se invierten los números de puerto de origen y destino en el encabezado del datagrama.

- Comparación

UDP y TCP son protocolos comunes de la capa de Transporte. Los datagramas UDP y los segmentos TCP tienen encabezados prefijados a los datos que incluyen un número de puerto origen y un número de puerto destino. Estos números de puertos permiten que los datos sean direccionados a la aplicación correcta que se ejecuta en la computadora de destino. TCP no envía datos a la red hasta que advierte que el destino está preparado para recibirlos. Luego TCP administra el flujo de datos y reenvía todos los segmentos de datos de los que recibió acuse a medida que se reciben en el destino.

TCP utiliza mecanismos de enlace, temporizadores y acuses de recibo y uso dinámico de ventanas para llevar a cabo estas funciones confiables. Sin embargo, esta confiabilidad representa cierta sobrecarga en la red en términos de encabezados de segmentos más grandes y mayor tráfico de red entre el origen y el destino que administra el transporte de datos. Si los datos de aplicación necesitan ser entregados a la red de manera rápida o si el ancho de banda de la red no admite la sobrecarga de mensajes de control que se intercambian entre los sistemas de origen y destino, UDP será el protocolo de la capa de Transporte preferido por el desarrollador. Esto es así porque UDP no rastrea ni reconoce la recepción de datagramas en el destino, sólo envía los datagramas recibidos a la capa de Aplicación a medida que llegan, y no reenvía datagramas perdidos. Sin embargo, esto no significa necesariamente que la comunicación no es confiable; puede haber mecanismos en los protocolos y servicios de la capa de Aplicación que procesan datagramas perdidos o demorados si la aplicación cuenta con esos requerimientos.

El desarrollador de la aplicación toma una decisión en cuanto al protocolo de la capa de Transporte en base a los requerimientos del usuario. Sin embargo, el desarrollador tiene en cuenta que las otras capas

cumplen un rol importante en las comunicaciones de redes de datos y tendrán influencia en el rendimiento.

- Fiabilidad

Se analiza como es posible enviar información fiable a través del protocolo UDP, que como ya se conoce es un protocolo no fiable. Si a través del protocolo TCP que es un protocolo fiable se pueden perder segmentos de los que se transportan, ¿Cómo logran llegar los datos del protocolo UDP a su destino?

El protocolo UDP para lograr una transferencia fiable devuelve una confirmación (*acknowledgement, ACK*) cada vez que recibe un mensaje, para que el emisor confirme que ha llegado correctamente. Si el emisor no recibe esta confirmación pasado un tiempo determinado, el emisor reenvía el mensaje.

A continuación se visualiza el modo más sencillo de proporcionar una comunicación fiable, aunque vale destacar que este método es ineficiente. El emisor envía un dato, pone en marcha su temporizador y espera su confirmación ACK. Si recibe su ACK antes de agotar el temporizador, envía el siguiente dato. Si se agota el temporizador antes de recibir el ACK, reenvía el mensaje. Los siguientes esquemas representan este comportamiento:

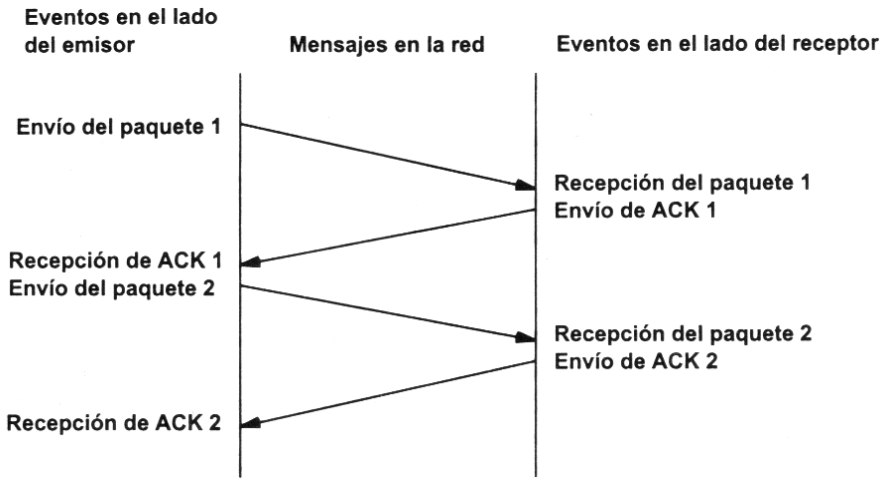


Fig. 2 Comunicación fiable UDP

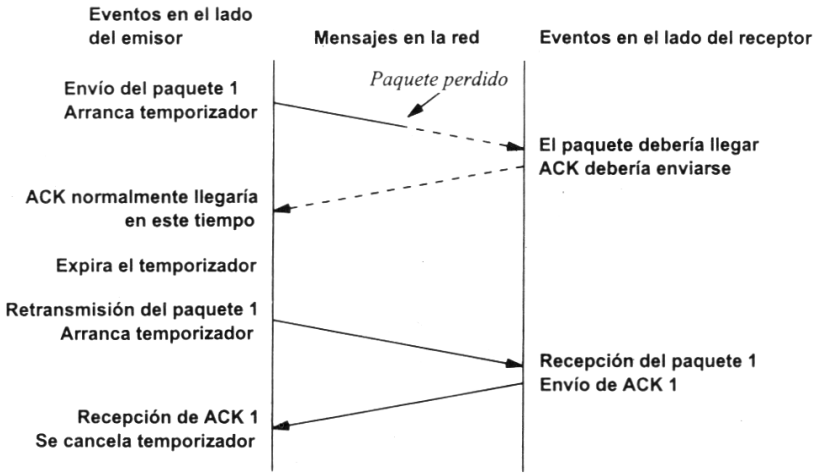


Fig. 2.1 Comunicación fiable UDP

2.2 Descripción del procedimiento

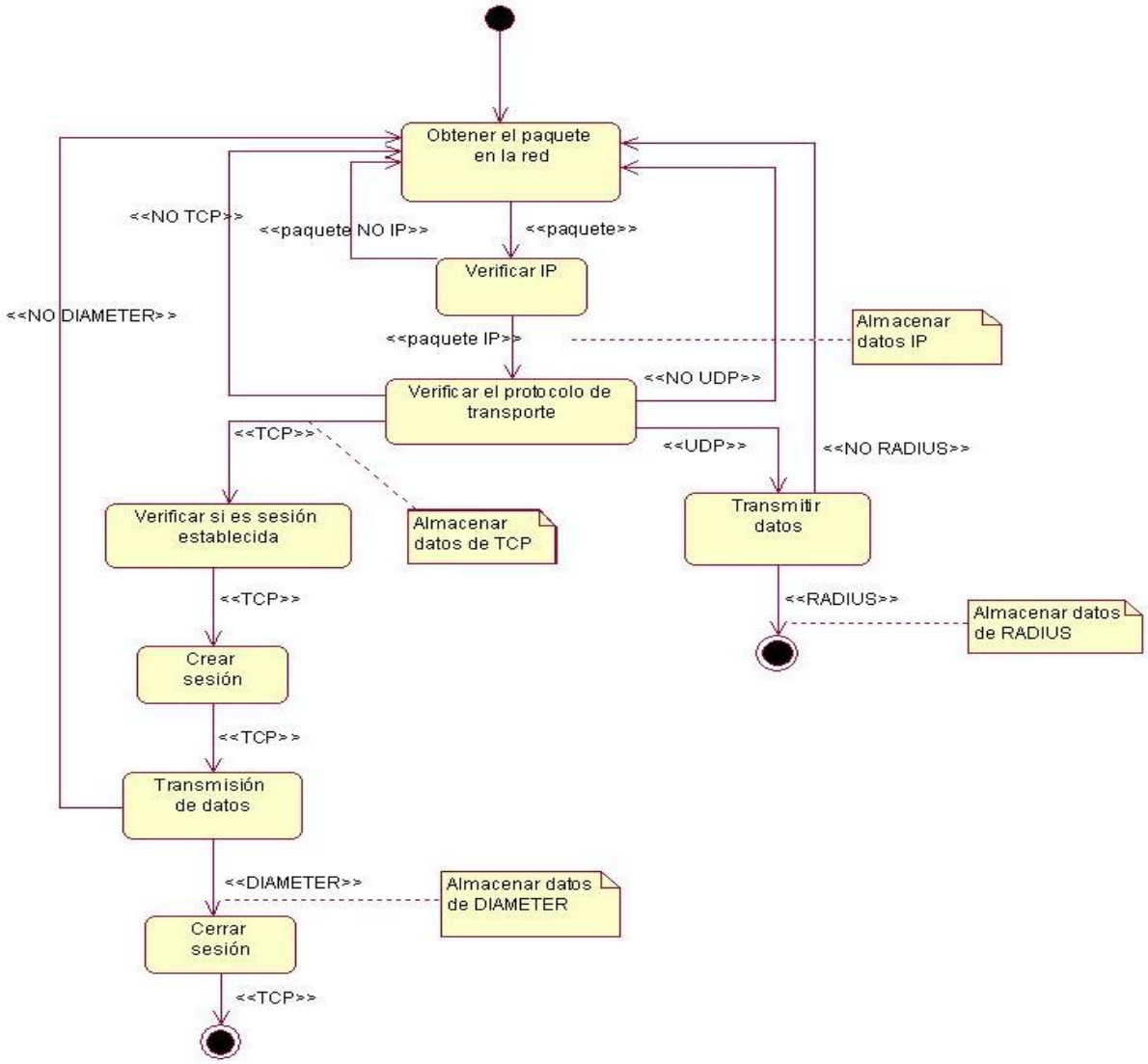


Fig. 3 Procedimiento

2.2.1 Obtención de un paquete en la red:

Cada paquete es enviado de forma independiente desde un origen hasta un destino a través de la capa de Acceso a la Red del Modelo TCP/IP. En cada ubicación de conmutación se define que ruta utilizar para

enviar el paquete hasta su destino final. En caso de que una ruta ya utilizada con anterioridad ya no se encuentre disponible, la función de enrutamiento puede seleccionar en forma dinámica la próxima ruta o vía disponible. Si se transmite el mensaje completo y único en caso de una falla se transmite nuevamente el mensaje, la ventaja principal de los paquetes es que si ocurre alguna falla solo se retransmite ese paquete que presentó la falla, en muchos casos el dispositivo de destino no tiene en cuenta que se ha producido un re enrutamiento o una falla. Los dispositivos existentes en una red tienen en cuenta de un paquete solamente los datos de la dirección destino final y del próximo dispositivo en la ruta hacia ese destino.

❖ Formato de la trama Ethernet:

IEEE 802.3						
7	1	6	6	2	46 a 1500	4
Preámbulo	Delimitador de inicio de trama	Dirección de destino	Dirección de origen	Longitud/Tipo	Encabezado y datos 802.2	Secuencia de verificación de trama

Fig. 4 Trama Ethernet Original

❖ Campos de la trama Ethernet

Preámbulo y Delimitador de Inicio de Trama (SFD)

Los campos de la trama Ethernet Preámbulo (7 bytes) y SFD (1 byte) se utilizan para la sincronización entre los dispositivos de envío y de recepción. Estos ocho primeros bytes de la trama se utilizan para captar la atención de los nodos receptores. Básicamente, los primeros bytes le indican al receptor que se prepare para recibir una trama nueva.

Dirección MAC de destino

El campo Dirección MAC de destino (6 bytes) es el identificador del receptor deseado. La Capa 2 utiliza esta dirección para ayudar a los dispositivos a determinar si la trama viene dirigida a ellos. La dirección de la trama se compara con la dirección MAC del dispositivo. Si coinciden, el dispositivo acepta la trama.

Dirección MAC de origen

El campo Dirección MAC de origen (6 bytes) identifica la NIC o interfaz que origina la trama. Los switches también utilizan esta dirección para ampliar sus tablas de búsqueda.

Datos y Relleno

Los campos Datos y Relleno (de 46 a 1500 bytes) contienen los datos encapsulados de una capa superior, que es una PDU de Capa 3 genérica o, con mayor frecuencia, un paquete IPv4. Todas las tramas deben tener al menos 64 bytes de longitud. Si se encapsula un paquete pequeño, el Pad se utiliza para aumentar el tamaño de la trama hasta alcanzar este tamaño mínimo.

Secuencia de Verificación de Trama (FCS)

FCS (Frame Check Sequence), campo de 32 bits (4 bytes) que contiene un valor de verificación CRC (Control de Redundancia Cíclica). El emisor calcula este CRC usando todo el contenido de la trama y el receptor lo recalcula y lo compara con el recibido a fin de verificar la integridad de la trama.

Campo Longitud/Tipo

El campo Longitud/Tipo (2 bytes) define la longitud exacta del campo Datos de la trama. Esto se utiliza posteriormente como parte de la FCS para garantizar que el mensaje se reciba adecuadamente. En este campo debe ingresarse una longitud o un tipo. Sin embargo, sólo uno u otro puede utilizarse en una determinada implementación. Si el objetivo del campo es designar un tipo, el campo Tipo describe qué

protocolo se implementa. El campo denominado Longitud/Tipo sólo aparecía como Longitud en las versiones anteriores del IEEE y sólo como Tipo en la versión DIX. Estos dos usos del campo se combinaron oficialmente en una versión posterior del IEEE, ya que ambos usos eran comunes. El campo Tipo de la Ethernet II se incorporó a la actual definición de trama del 802.3. La Ethernet II es el formato de trama de Ethernet que se utiliza en redes TCP/IP. Cuando un nodo recibe una trama, debe analizar el campo Longitud/Tipo para determinar qué protocolo de capa superior está presente. Si el valor de los dos octetos es equivalente a 0x0600 hexadecimal o 1536 decimal o mayor que éstos, los contenidos del campo Datos se codifican según el protocolo indicado. En caso de ser 0800 el paquete que se analiza es de la sección IP.

2.2.2 Verificación del paquete IP:

Ya obtenido el paquete se procede a verificar si es un paquete IP, para esto se hace necesario analizar la trama Ethernet, específicamente el campo Longitud/Tipo y si este es igual a 0800 estamos en presencia de un paquete IP.

- Encabezado del paquete IPv4

Un protocolo IPv4, define muchos campos diferentes en el encabezado del paquete. Estos campos contienen valores binarios que los servicios IPv4 de 32 bits, toman como referencia a medida que envían paquetes a través de la red.

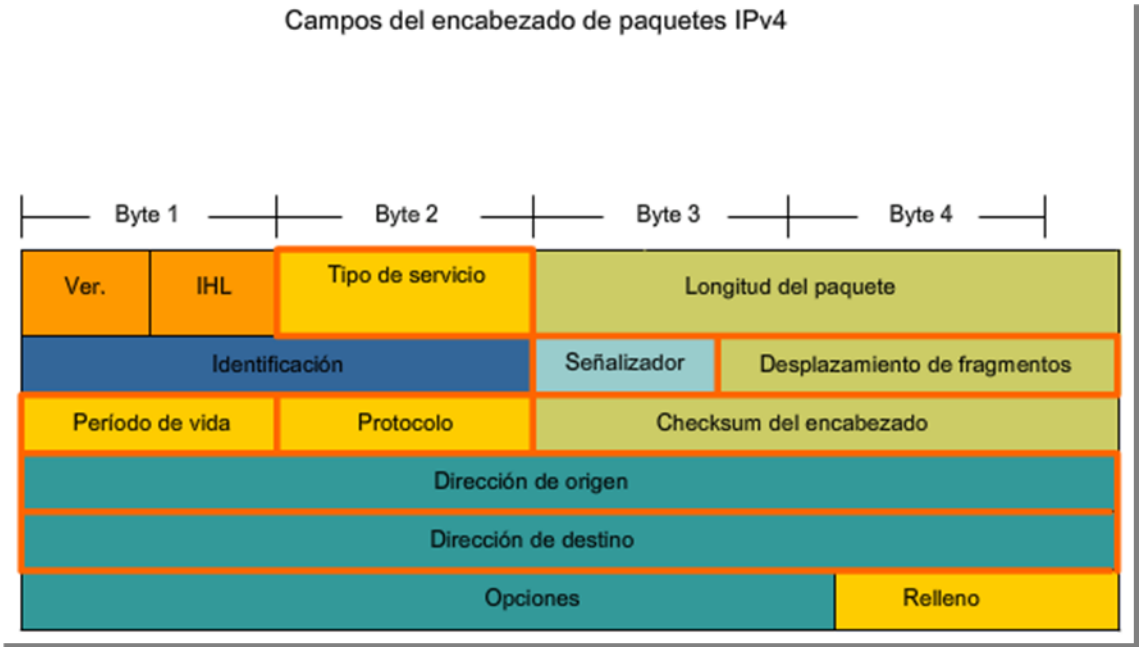


Fig. 5 Encabezado IPv4

❖ Descripción de los campos del paquete IPv4

Versión: 4 bits

Siempre vale lo mismo (0100). Este campo describe el formato de la cabecera utilizada. En la figura se describe la versión 4.

Tamaño Cabecera (IHL): 4 bits

IHL (Internet Header Length), longitud de la cabecera. Su valor mínimo es de 5 para una cabecera correcta, y el máximo de 15.

Tipo de Servicio: 8 bits

Indica una serie de parámetros sobre la calidad de servicio deseada durante el tránsito por una red. Algunas redes ofrecen prioridades de servicios, considerando determinado tipo de paquetes "más

importantes" que otros (en particular estas redes solo admiten los paquetes con prioridad alta en momentos de sobrecarga). Estos 8 bits se agrupan de la siguiente manera. Los 5 bits de menos peso son independientes e indican características del servicio:

Bit 0: sin uso, debe permanecer en 0.
Bit 1: 1 costo mínimo, 0 costo normal.
Bit 2: 1 máxima fiabilidad, 0 fiabilidad normal.
Bit 3: 1 máximo rendimiento, 0 rendimiento normal.
Bit 4: 1 mínimo retardo, 0 retardo normal.

Los 3 bits restantes están relacionados con la precedencia de los mensajes, un indicador ajunto que indica el nivel de urgencia basado en el sistema militar de precedencia de la CCEB, una organización de comunicaciones electrónicas militares formada por 5 naciones. La urgencia que estos estados representan aumenta a medida que el número formado por estos 3 bits lo hace, y responden a los siguientes nombres.

000: De rutina.
001: Prioritario.
010: Inmediato.
011: Relámpago.
100: Invalidación relámpago.
101: Procesando llamada crítica y de emergencia.
110: Control de trabajo de Internet.
111: Control de red.

Longitud Total: 16 bits

Es el tamaño total, en octetos, del datagrama, incluyendo el tamaño de la cabecera y el de los datos. El tamaño máximo de los datagramas usados normalmente es de 576 octetos (64 de cabeceras y 512 de datos). Una máquina no debería enviar datagramas mayores a no ser que tenga la certeza de que van a ser aceptados por la máquina destino.

En caso de fragmentación este campo contendrá el tamaño del fragmento, no el del datagrama original.

Identificador: 16 bits

Identificador único del datagrama. Se utilizará, en caso de que el datagrama deba ser fragmentado, para poder distinguir los fragmentos de un datagrama de los de otro. El originador del datagrama debe asegurar un valor único para la pareja origen-destino y el tipo de protocolo durante el tiempo que el datagrama pueda estar activo en la red.

Desplazamiento de fragmentos: 16 bits

- Indicadores: 3 bits

  Actualmente utilizado sólo para especificar valores relativos a la fragmentación de paquetes:

  bit **0**: Reservado; debe ser 0
  bit **1**: **0** = Divisible, **1** = No Divisible (DF)
  bit **2**: **0** = Último Fragmento, **1** = Fragmento Intermedio (le siguen más fragmentos) (MF)
  La indicación de que un paquete es indivisible debe ser tenida en cuenta bajo cualquier circunstancia. Si el paquete necesitara ser fragmentado, no se enviará.
- Posición de Fragmento: 13 bits

En paquetes fragmentados indica la posición, en unidades de 64 bits, que ocupa el paquete actual dentro del datagrama original. El primer paquete de una serie de fragmentos contendrá en este campo el valor 0.

Tiempo de Vida (Time To Live, TTL): 8 bits

Indica el máximo número de direccionadores que un paquete puede atravesar. Cada vez que algún nodo procesa este paquete disminuye su valor en, como mínimo, un direccionador. Cuando llegue a ser 0, el paquete no será reenviado.

Protocolo: 8 bits

Indica el protocolo de siguiente nivel utilizado en la parte de datos del datagrama.

Suma de Control de Cabecera: 16 bits

Suma de Control de cabecera. Se recalcula cada vez que algún nodo cambia alguno de sus campos (por ejemplo, el Tiempo de Vida). El método de cálculo (intencionadamente simple) consiste en sumar el complemento a 1 de cada palabra de 16 bits de la cabecera y hacer el complemento a 1 del valor resultante.

Dirección IP de origen: 32 bits

El campo de Dirección IP origen representa la dirección de host de capa de red de origen del paquete.

Dirección IP de destino: 32 bits

El campo de Dirección IP destino representa la dirección de host de capa de red de destino del paquete.

Opciones: Variable

Aunque no es obligatoria la utilización de este campo, cualquier nodo debe ser capaz de interpretarlo.

Relleno: Variable

Utilizado para asegurar que el tamaño, en bits, de la cabecera es un múltiplo de 32. El valor usado es el 0.

❖ Paquete IP típico

- Ver. = 4; version IP.
- IHL = 5; tamaño del encabezado en palabras de 32 bits (4 bytes). Este encabezado tiene 5*4 = 20 bytes, el tamaño mínimo válido.
- Longitud total = 472; tamaño del paquete (encabezado y datos) de 472 bytes.
- Identificación = 111; identificador original del paquete (requerido si se fragmenta posteriormente).
- Señalizador = 0; significa que el paquete puede ser fragmentado si se requiere.

- Desplazamiento de fragmentos = 0; significa que este paquete no está actualmente fragmentado (no existe desplazamiento).
- Período de vida = 123; es el tiempo de procesamiento en segundos de la Capa 3 antes de descartar el paquete (disminuye en al menos 1, cada vez que el dispositivo procesa el encabezado del paquete).
- Protocolo = 6; significa que los datos llevados por este paquete son un segmento TCP.

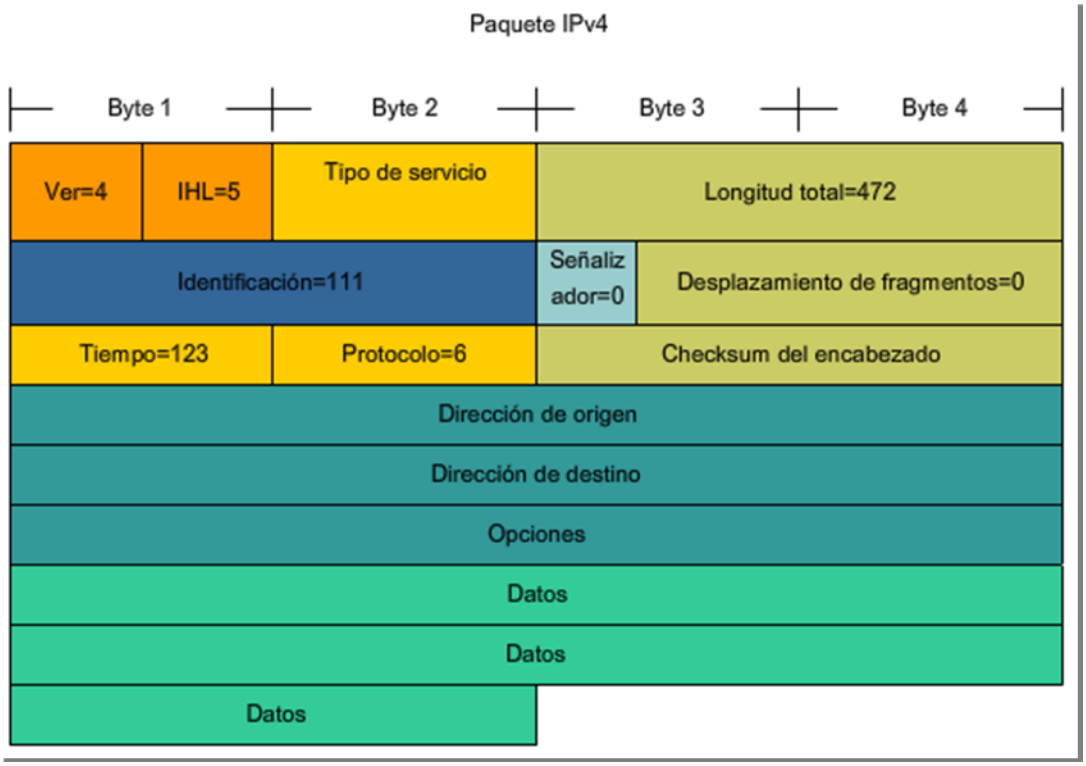


Fig. 6 Paquete IP

❖ Almacenamiento:

Ya realizado el análisis y verificación del paquete IP procedemos al almacenamiento en los campos IPorigen e IPdestino de la tabla TCP_IP de la base de datos llamada radius los datos que viajan en los campos Dirección IP de origen y Dirección IP de destino respectivamente.

2.2.3 Verificación del protocolo de transporte:

Una vez verificado que el paquete en cuestión es de la sección IP se procede al análisis del encabezado IP. Los campos de este encabezado traen información importante acerca del paquete, tales como la versión IP que utiliza, la dirección origen y destino de la transmisión, el tamaño total del paquete, el protocolo adecuado de la capa superior entre otros datos de interés. El campo protocolo es el que indica que protocolo de transporte se emplea para la transmisión en cuestión. Si es 1, corresponde al protocolo ICMP, 2 al IGMP, 6 al TCP y 17 al UDP. Cuando el campo protocolo es TCP, es decir, es igual a 6, es necesario verificar que tipo de paquete se está transmitiendo si es de sincronismo, de reconocimiento o de datos.

- Formato del segmento TCP

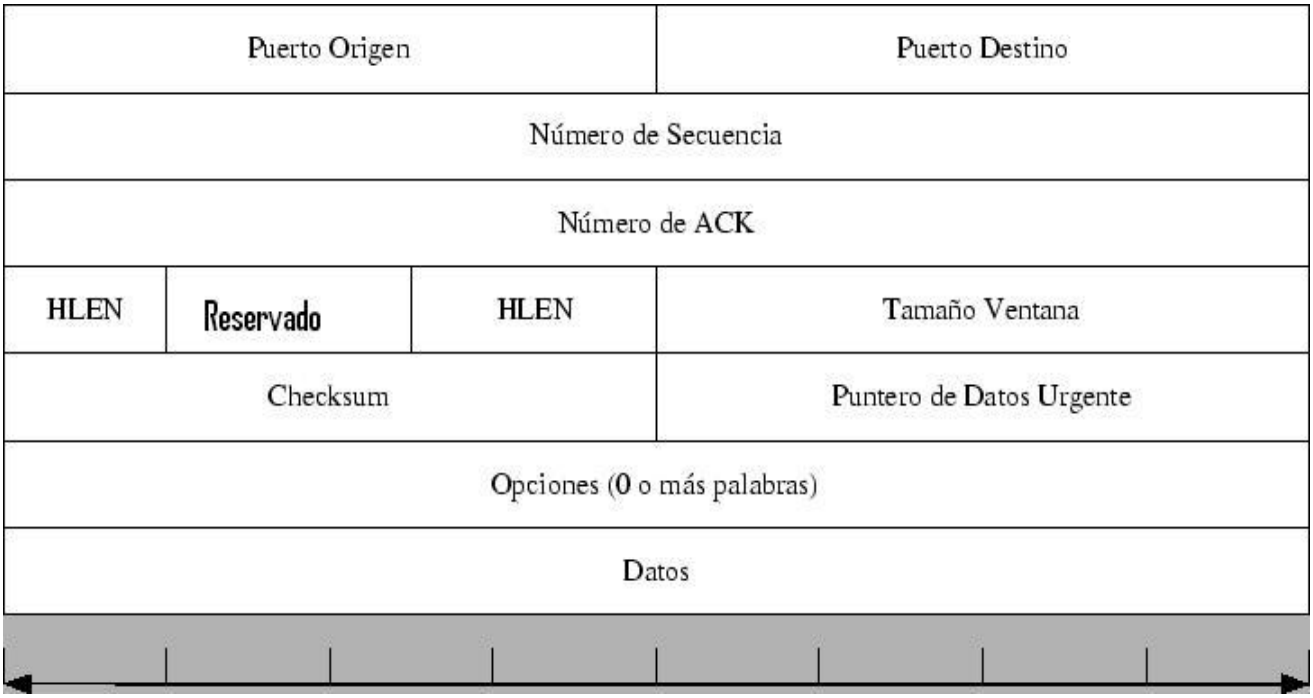


Fig. 7 Formato del segmento TCP

❖ Campos del segmento TCP

Puerto Origen y Destino

Contiene 2 bytes, cada uno identifican los puertos que se van a utilizar en cada host para comunicarse con las aplicaciones que intercambian datos.

Número de Secuencia

Contiene 4 bytes, indican el número de secuencia que corresponde, en la conexión, al primer byte que se envía en el campo datos de ese segmento.

Número de ACK

Contiene 4 bytes que apuntan al número de secuencia del primer byte del próximo segmento que se espera recibir del otro lado.

Longitud de Encabezado

Contiene 4 bits que especifican el largo del encabezado, en palabras de 32 bits. Este valor no incluye el campo datos, y el campo opciones hace que esta longitud pueda variar.

Bits de Codificación

Contiene 6 bits que se presentan a continuación de 6 bits no utilizados. Corresponden a bits flag, cuyo nombre y significado es el siguiente: URG (urgent, sirve para indicar que el segmento contiene datos urgentes, y el campo puntero de datos urgentes contiene la dirección donde terminan éstos), ACK (acknowledgement, indica que en este segmento el campo Número de ACK tiene el significado habitual , de lo contrario carece de significado; en la práctica, el bit ACK esta en 1 siempre, excepto en el primer segmento enviado por el host que inicia la conexión), PSH (push, indica que el segmento contiene datos PUSHED, es decir, que deben ser enviados rápidamente a la aplicación correspondiente sin esperar a

acumular varios segmentos), RST (reset, usado para indicar que se debe abortar una conexión porque se ha detectado un error de cualquier tipo), SYN (synchronize, este bit indica que se está estableciendo la conexión y está puesto en 1 sólo en el primer mensaje enviado por cada uno de los dos hosts en el inicio de la conexión) y FIN (finish, indica que no se tienen más datos que enviar y que se quiere cerrar la conexión; se usa ya que para que una conexión se cierre de manera normal cada host ha de enviar un segmento con el bit FIN puesto en 1)

Tamaño de Ventana

Contiene 2 bytes que indican la cantidad de bytes que se está dispuesto a aceptar del otro lado en cada momento. Mediante este parámetro el receptor establece un control de flujo sobre el flujo de datos que puede enviar el emisor.

Checksum

Contiene 2 bytes y sirve para detectar errores en el segmento recibido. Estos podrían ser debidos a errores de transmisión no detectados, a fallos en los equipos o a problemas en el software.

Puntero de Datos Urgentes.

Contiene 2 bytes, indican el final de un flujo de datos de tipo urgente, ya que el segmento podría contener datos no urgentes. TCP no marca el principio de los datos urgentes, es responsabilidad de la aplicación averiguarlo.

Opciones

Guarda cero o más bytes que habilitan un mecanismo por el cual es posible incluir extensiones al protocolo. Entre las más interesantes se encuentran las siguientes: tamaño máximo de segmento, uso de repetición selectiva (en vez de retroceso n), uso de NAK (acuse de recibo negativo en caso de no recepción de un segmento), uso de ventana mayor de 64 KB (Kilobytes) mediante el empleo de un factor de escala.

❖ Almacenamiento:

Posterior a la verificación y análisis y del Protocolo TCP, se procede a almacenar los campos de interés del segmento TCP. Se almacenan en los campos Puerto origen y Puerto destino de la tabla TCP_IP de la base de datos radius, los puertos que se van a utilizar en cada host para comunicarse con las aplicaciones que intercambian datos.

❖ Conexión TCP:

Para llevar a cabo una conexión los hosts realizan un intercambio de señales de tres vías. Primeramente en el enlace de tres vías, se verifica que el dispositivo de destino esté presente en la red. En segundo lugar, se comprueba que el dispositivo de destino tenga un servicio activo y esté aceptando las peticiones en el número de puerto de destino que el cliente que lo inicia intente usar para la sesión. Y por último, se informa al dispositivo de destino que el cliente de origen intenta establecer una sesión de comunicación en ese número de puerto. En las conexiones TCP, el host que brinde el servicio como cliente inicia la sesión al servidor. Los siguientes tres pasos definen el establecimiento de una conexión TCP:

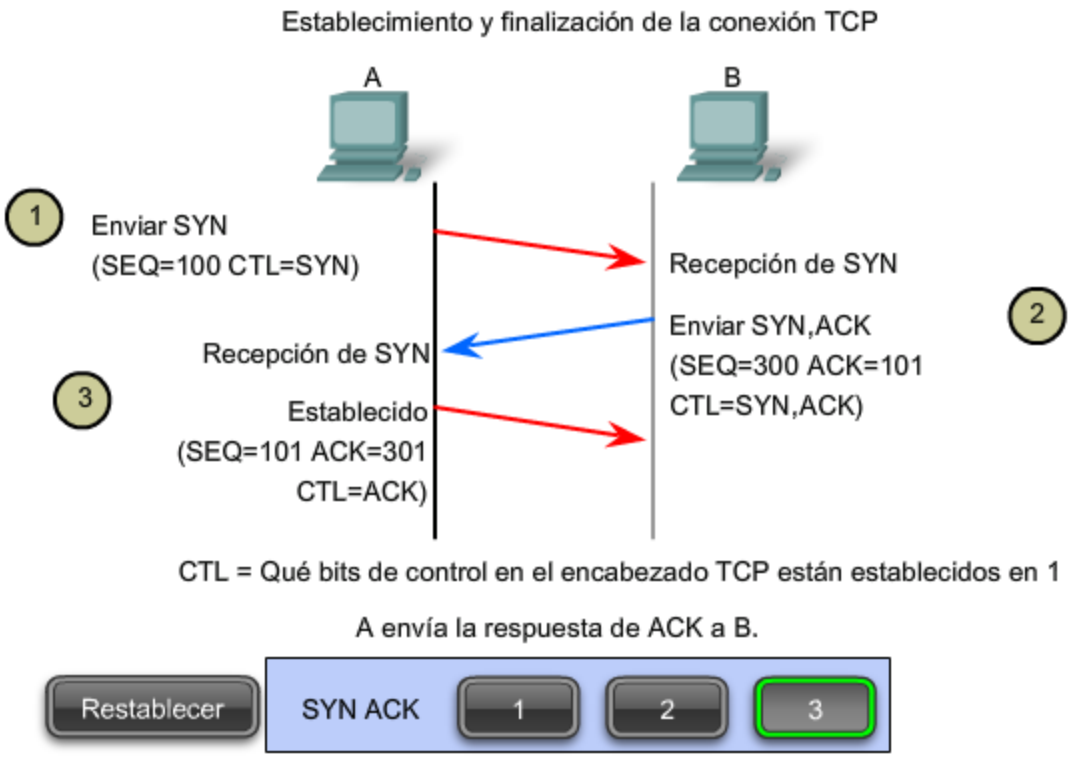


Fig. 8 Establecimiento de una conexión TCP

1. El cliente que inicia la conexión envía un segmento que contiene un valor de secuencia inicial, que actúa como solicitud para el servidor para comenzar una sesión de comunicación.
2. El servidor responde con un segmento que contiene un valor de reconocimiento igual al valor de secuencia recibido más 1, además de su propio valor de secuencia de sincronización. El valor es uno mayor que el número de secuencia porque el ACK es siempre el próximo Byte u Octeto esperado. Este valor de reconocimiento permite al cliente unir la respuesta al segmento original que fue enviado al servidor.
3. El cliente que inicia la conexión responde con un valor de reconocimiento igual al valor de secuencia que recibió más uno. Esto completa el proceso de establecimiento de la conexión.

Es importante observar los distintos valores que intercambian los dos hosts, para poder entender el proceso de enlace de tres vías. El encabezado del segmento TCP incluye seis campos de 1 bit que

contienen información de control utilizada para gestionar los procesos de TCP. Estos campos se nombran a continuación:

URG: Urgente campo de señalizador significativo.

ACK: Campo significativo de acuse de recibo.

PSH: Función de empuje.

RST: Reconfiguración de la conexión.

SYN: Sincronizar números de secuencia.

FIN: No hay más datos desde el emisor.

A estos campos se les denomina señaladores porque el valor de uno de estos campos es sólo de un bit, entonces tiene sólo dos valores: 1 ó 0. Si el valor del bit se establece en 1, indica la información de control que contiene el segmento. El campo puerto de destino define el puerto que emplea el host para comunicarse con las aplicaciones que intercambian los datos, si el campo número de puerto de destino del segmento TCP es 3868, entonces el protocolo que se utiliza para la autenticación de usuarios es el DIAMETER. Este protocolo ofrece un marco a nivel de tecnología de acceso, para los servicios que requieran soporte de AAA.

2.2.3.1 Verificación de la existencia de las sesiones TCP:

En caso de que se calcule el tamaño de la Longitud del paquete y este demuestre que solo se transmiten los datos del encabezado del paquete, es decir, que no están establecidas las sesiones, se procede a la creación de las mismas. Una sesión está formada por la combinación única de los siguientes 5 parámetros: ProtocoloIP, IpOrigen, IpDestino, PuertoOrigen, PuertoDestino**.**

El cálculo del tamaño de la Longitud del paquete se realiza con la siguiente fórmula:

Longitud_Datos = Longitud_Total – (Longitud_IP + Longitud_TCP)

Si la Longitud_Datos es un valor diferente de 0 significa que ya están establecidas las sesiones y se procede directamente a la trasmisión de los datos. En caso de que Longitud_Datos sea igual a 0 entonces se está en presencia de un mensaje de los que se emplean para el establecimiento de las sesiones y se procede a la creación de las mismas.

2.2.3.1.1 Establecimiento de las sesiones:

Si la longitud de los Datos presente en la fórmula anterior es 0 entonces se está en presencia de un paquete de control.

2.2.3.1.2 Transmisión de los datos:

DIAMETER

- Formato de segmento DIAMETER

A continuación se muestra un resumen del formato del encabezado DIAMETER. Los campos se transmiten en bytes por la red.

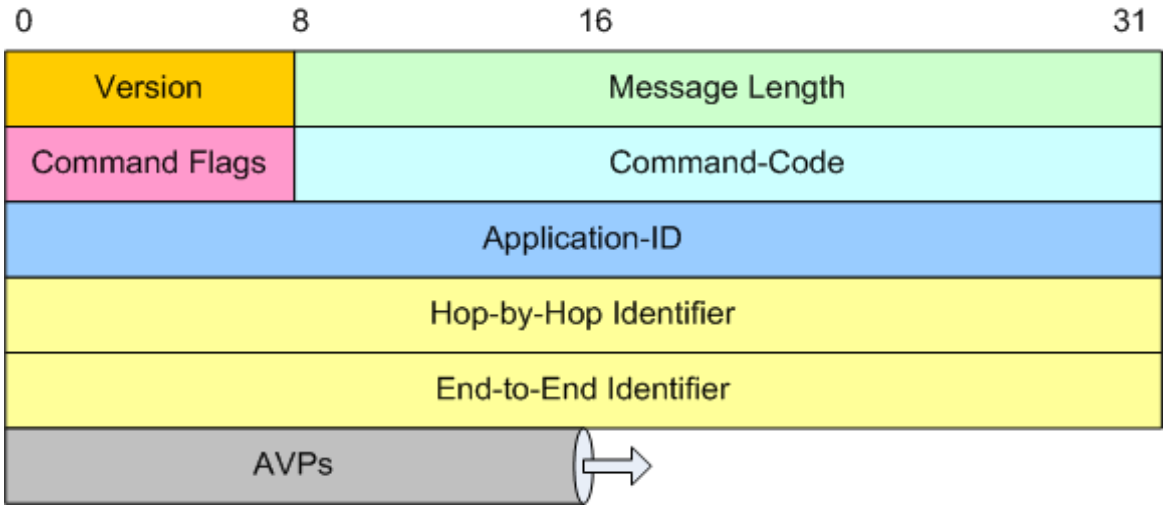


Fig. 9 Formato del segmento DIAMETER

Version

El campo versión debe estar en 1 para indicar la versión DIAMETER.

Message Length

El campo Message Length es de tres octetos e indica el tamaño del mensaje DIAMETER incluyendo los campos del encabezado.

Command Flags

El campo Command Flags es de ocho bits. Los siguientes bits son los que están asignados:

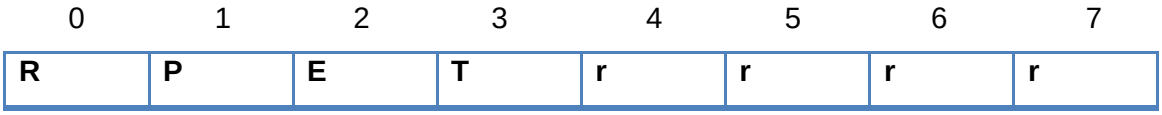


Fig. 10 Command Flags

Donde:

R(equest)

Si está activo el mensaje es una petición. Si no está activo entonces el mensaje es una respuesta.

P(roxiable)

Si se activa, el mensaje puede ser un servidor proxy, retransmitido o reorientado. Si no está activo, el mensaje debe ser procesado a nivel local.

E(rror)

Si se establece este bit, es porque el mensaje contiene un error de protocolo, y el mensaje no se ajusta a las ABNF descritas para este comando. Los mensajes con la «E» bit se conocen comúnmente mensajes de error. Este bit no debe ser activado en los mensajes de solicitud.

T(Potentially re-transmitted message)

Esta bandera se activa después procedimiento de enlace fallido, y además ayuda a la eliminación de los duplicados de las solicitudes. Se trata de definir cuándo reenviar las solicitudes aún no reconocidas como una indicación de una posible duplicación de enlace debido a un fallo. Este bit debe ser aclarado en el envío de una solicitud por primera vez, de lo contrario el remitente debe establecer este indicador. Los agentes DIAMETER sólo tienen que estar preocupados por el número de solicitudes que envían basados en una única solicitud recibida, las retransmisiones por otras entidades no necesitan ser rastreadas. Los agentes DIAMETER que reciben una solicitud con el T pabellón conjunto, deben mantener la T pabellón conjunto en la solicitud remitida. Esta bandera no debe activarse si un mensaje de error de respuesta (por ejemplo, un protocolo de error) se ha recibido por el mensaje anterior. Que sólo se puede ajustar en los casos en que no se ha recibido respuesta desde el servidor de una solicitud y la petición se envió de nuevo. Esta bandera no debe activarse en el mensaje de respuesta.

r(eserved)

La bandera de estos bits se reservan para uso futuro, y debe ser ajustado a cero, e ignorado por el receptor.

Command-Code

El campo Command-Code es de tres octetos, y se utiliza para comunicar el mensaje con el comando asociado.

Application-ID

El campo Application-ID es de cuatro octetos y se utiliza para identificar a los mensajes de aplicación. La aplicación puede ser una solicitud de autenticación, una aplicación de contabilidad o un vendedor de aplicación específica. El campo Application-ID de la solicitud en la cabecera debe ser el mismo que lo que está contenido en cualquier AVP's que figura en el mensaje.

Hop-by-Hop Identifier:

El Hop-by-Hop Identifier es un entero sin signo de 32 bits campo (orden de bytes de red) y ayuda en la adecuación de las solicitudes y respuestas. El remitente debe asegurarse de que el campo Hop-by-Hop Identifier en una solicitud es único en un determinado sentido en un momento dado, y puede garantizar que el número es único a través de reinicios. El remitente de un mensaje de respuesta debe garantizar que el campo Hop-by-Hop Identifier contiene el mismo valor que se encontró en la correspondiente solicitud. El Hop-by-Hop Identifier es normalmente un número cada vez mayor monótonamente, cuyo inicio se calculó con un valor generado al azar. Un mensaje de respuesta que se recibe con un campo Hop-by-Hop Identifier desconocido debe ser desechado.

End-to-End Identifier:

El campo End-to-End Identifier es un entero sin signo de 32 bits (orden de bytes de red) y se utiliza para la detección de mensajes duplicados. Al reiniciar la aplicación se puede establecer 12 bits de alto orden para contener la baja de 12 bits para la hora actual y 20 bits de bajo orden para un valor aleatorio. Los remitentes de mensajes de solicitud deben insertar un identificador único en cada mensaje. El identificador único a nivel local debe permanecer invariable por un período de al menos 4 minutos, incluso a través de reinicios. El iniciador de un mensaje de respuesta debe asegurar que el campo End-to-End Identifier contiene el mismo valor que se encontró en la correspondiente solicitud. Este campo no debe ser modificado por cualquier tipo de agentes DIAMETER. La combinación del Hosts - Origen y este campo se utiliza para la detección de duplicados. Los mensajes de respuesta duplicados que se ejecuten a nivel local deben ser descartados en silencio.

AVPs:

AVPs es un método de encapsulación de la información pertinente para el mensaje DIAMETER.

Cada comando par Solicitud/Respuesta se le asigna un código de comando, y el sub-tipo, es decir, la petición o respuesta se identifica a través de la «R» en el campo comando DIAMETER de la cabecera. Cada mensaje DIAMETER debe contener un código en su cabecera del campo Comando Código, que se utiliza para determinar la acción que se toma para un mensaje de mando. Los siguientes códigos se definen en el protocolo base DIAMETER:

- Comandos

Cada comando tiene asignado un código de comando, que se utiliza tanto para las solicitudes

Command-Name	Abbr.	Code
AA-Request	AAR	265
AA-Answer	AAA	265
Diameter-EAP-Request	DER	268
Diameter-EAP-Answer	DEA	268
Abort-Session-Request	ASR	274
Abort-Session-Answer	ASA	274
Accounting-Request	ACR	271
Accounting-Answer	ACA	271
Credit-Control-Request	CCR	272
Credit-Control-Answer	CCA	272
Capabilities-Exchange-Request	CER	257
Capabilities-Exchange-Answer	CEA	257
Device-Watchdog-Request	DWR	280
Device-Watchdog-Answer	DWA	280
Disconnect-Peer-Request	DPR	282
Disconnect-Peer-Answer	DPA	282

Re-Auth-Request	RAR	258
Re-Auth-Answer	RAA	258
Session-Termination-Request	STR	275
Session-Termination-Answer	STA	275
User-Authorization-Request	UAR	300
User-Authorization-Answer	UAA	300
Server-Assignment-Request	SAR	301
Server-Assignment-Answer	SAA	301
Location-Info-Request	LIR	302
Location-Info-Answer	LIA	302
Multimedia-Auth-Request	MAR	303
Multimedia-Auth-Answer	MAA	303
Registration-Termination-Request	RTR	304
Registration-Termination-Answer	RTA	304
Push-Profile-Request	PPR	305
Push-Profile-Answer	PPA	305
User-Data-Request	UDR	306
User-Data-Answer	UDA	306
Profile-Update-Request	PUR	307
Profile-Update-Answer	PUA	307
Subscribe-Notifications-Request	SNR	308
Subscribe-Notifications-Answer	SNA	308
Push-Notification-Request	PNR	309
Push-Notification-Answer	PNA	309
Bootstrapping-Info-Request	BIR	310
Bootstrapping-Info-Answer	BIA	310
Message-Process-Request	MPR	311
Message-Process-Answer	MPA	311

Fig. 11 Comandos y códigos del protocolo DIAMETER

❖ Atributos:

Attribute-Name	Code	Data Type
Acct-Interim-Interval	85	Unsigned32
Accounting-Realtime-Required	483	Enumerated
Acct-Multi-Session-Id	50	UTF8String
Accounting-Record-Number	485	Unsigned32
Accounting-Record-Type	480	Enumerated
Accounting-Session-Id	44	OctetString
Accounting-Sub-Session-Id	287	Unsigned64
Acct-Application-Id	259	Unsigned32
Auth-Application-Id	258	Unsigned32
Auth-Request-Type	274	Enumerated
Authorization-Lifetime	291	Unsigned32
Auth-Grace-Period	276	Unsigned32
Auth-Session-State	277	Enumerated
Re-Auth-Request-Type	285	Enumerated
Class	25	OctetString
Destination-Host	293	DiamIdent
Destination-Realm	283	DiamIdent
Disconnect-Cause	273	Enumerated
E2E-Sequence	300	Grouped
Error-Message	281	UTF8String
Error-Reporting-Host	294	DiamIdent
Event-Timestamp	55	Time
Experimental-Result	297	Grouped
Experimental-Result-	298	Unsigned32

Failed-AVP	279	Grouped
Firmware-Revision	267	Unsigned32
Host-IP-Address	257	Address
Inband-Security-Id	299	Unsigned32
Multi-Round-Time-Out	272	Unsigned32
Origin-Host	264	DiamIdent
Origin-Realm	296	DiamIdent
Origin-State-Id	278	Unsigned32
Product-Name	269	UTF8String
Proxy-Host	280	DiamIdent
Proxy-Info	284	Grouped
Proxy-State	33	OctetString
Redirect-Host	292	DiamURI
Redirect-Host-Usage	261	Enumerated
Redirect-Max-Cache-Time	262	Unsigned32
Result-Code	268	Unsigned32
Route-Record	282	DiamIdent
Session-Id	263	UTF8String
Session-Timeout	27	Unsigned32
Session-Binding	270	Unsigned32
Session-Server-Failover	271	Enumerated
Supported-Vendor-Id	265	Unsigned32
Termination-Cause	295	Enumerated
User-Name	1	UTF8String
Vendor-Id	266	Unsigned32
Vendor-Specific-Application-Id	260	Grouped

Fig. 12 Atributos del protocolo DIAMETER

Secuencia de mensajes DIAMETER:

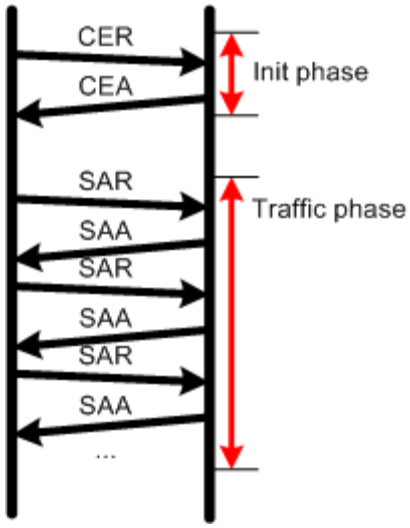


Fig. 13 Secuencia de mensajes DIAMETER

❖ Fase de Inicio

Con los comandos que se describen a continuación se logra el inicio y establecimiento de la conexión y se procede al envío de los datos.

CER: DIAMETER aplicación ha señalado que una conexión debe ser iniciada con el compañero.

CEA: Se recibe un acuse de recibo indicando que se ha establecido el transporte de la conexión, y el CER asociados ha llegado

❖ Fase de Tráfico:

SAR: DIAMETER realiza una solicitud de asignación al servidor.

SAA: DIAMETER recibe la respuesta de la solicitud de asignación realizada.

❖ Almacenamiento:

Posterior a la verificación y análisis del Protocolo DIAMETER se procede al almacenamiento en la tabla TCP_IP de la base de datos radius los campos Código del Comando y Banderas.

2.2.3.1.3 Terminación de las conexiones:

El señalizador de FIN (Finalizar) en el encabezado del segmento se debe establecer para cerrar la conexión. Para finalizar todas las sesiones TCP de una vía, se utiliza un enlace de dos vías, que consta de un segmento FIN y un segmento ACK. Por lo tanto, para terminar una conversación simple admitida por TCP, se requieren cuatro intercambios para finalizar ambas sesiones.

I. Cuando el cliente no tiene más datos para enviar al servidor, envía un segmento con el señalizador FIN establecido.
II. El servidor envía un ACK para acusar recibo de Fin y terminar la sesión del cliente al servidor.
III. El servidor envía un FIN al cliente para finalizar la sesión del servidor al cliente.
IV. El cliente responde con un ACK para dar acuse de recibo de FIN desde el servidor.

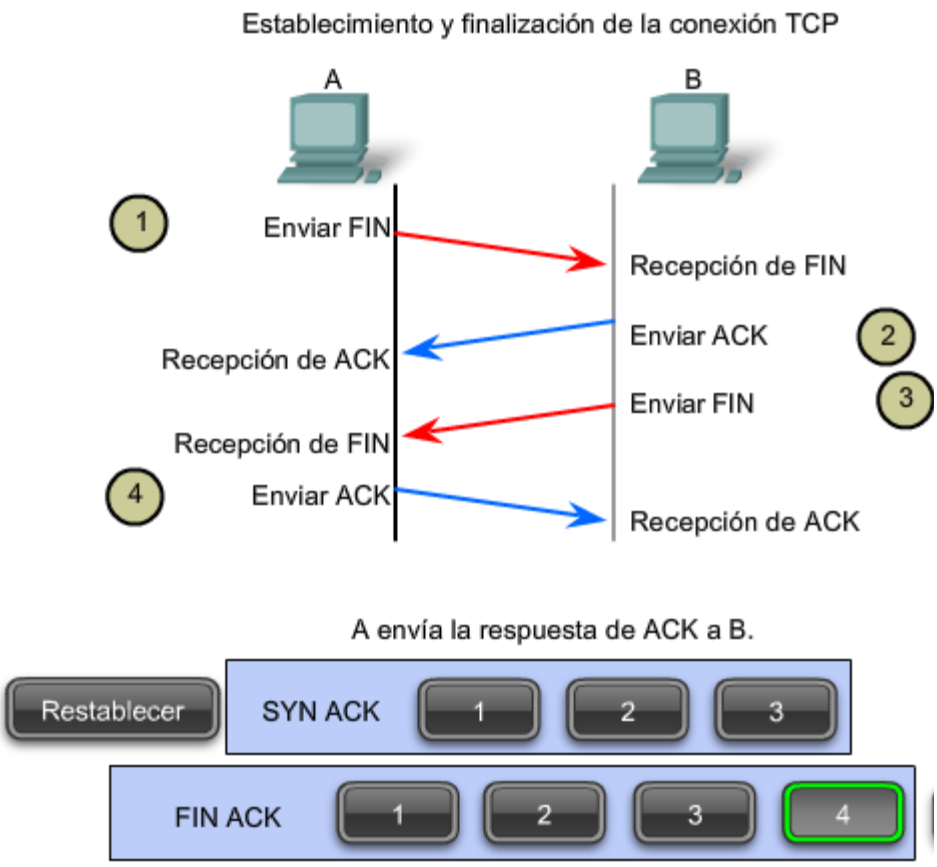


Fig. 14 Cierre de una sesión TCP

Es importante señalar que se usan los términos cliente y servidor como referencia pero cualquiera de los dos hosts que completan la sesión pueden iniciar la finalización del proceso.

Cuando no se tienen más datos para transferir desde la sesión del cliente, el señalizador FIN se establece en el encabezado de un segmento. Luego, el servidor finaliza la conexión y envía un segmento normal que contiene datos con el señalizador ACK establecido utilizando el número de acuse de recibo, confirmando así que se han recibido todos los bytes de datos. Al producirse el acuse de recibo de todos los segmentos la sesión se cierra.

Mediante el mismo proceso se cierra la sesión en la otra dirección. El receptor indica que no existen más datos para enviar estableciendo el señalizador FIN en el encabezado del segmento enviado al origen. Se

confirma, a través de un acuse de recibo de retorno que han sido recibidos todos los bytes de datos y, por ende, la sesión se ha cerrado.

También se puede mediante un enlace de tres vías terminar la conexión. Al momento de que el cliente no posee más datos para enviar, transmite un señalizador FIN al servidor. Si el servidor tampoco tiene más datos para enviar, puede responder con los señalizadores FIN y ACK, combinando dos pasos en uno. El cliente responde con un ACK.

2.2.3.2 Obtención de los datos:

Si el campo Protocolo del paquete IP es 17, entonces se está en presencia de un segmento de datos del protocolo UDP. El protocolo UDP proporciona una comunicación muy sencilla entre las aplicaciones de dos ordenadores. Como UDP opera sin conexión, las sesiones no se establecen antes de que se lleve a cabo la comunicación, como sucede con TCP. Se dice que UDP es basado en transacciones. En otras palabras, cuando una aplicación posee datos para enviar, simplemente los envía.

Cuando se envían múltiples datagramas a un destino, los mismos pueden tomar rutas distintas y llegar en el orden incorrecto. Por lo tanto, UDP simplemente reensambla los datos en el orden en que se recibieron y los envía a la aplicación. Si la secuencia de los datos es importante para la aplicación, la misma debe identificar la secuencia adecuada de datos y determinar cómo procesarlos.

❖ Formato del datagrama UDP

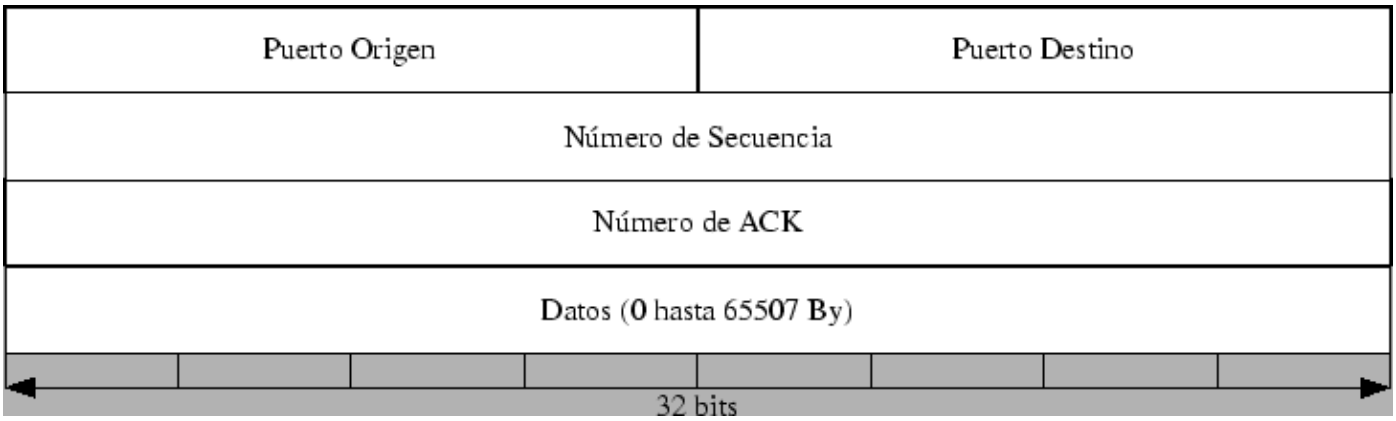


Fig. 15 Formato del datagrama UDP

❖ Campos del datagrama UDP

Longitud

Contiene 2 bytes que indican la longitud del mensaje, incluyendo los campos de encabezado.

Checksum

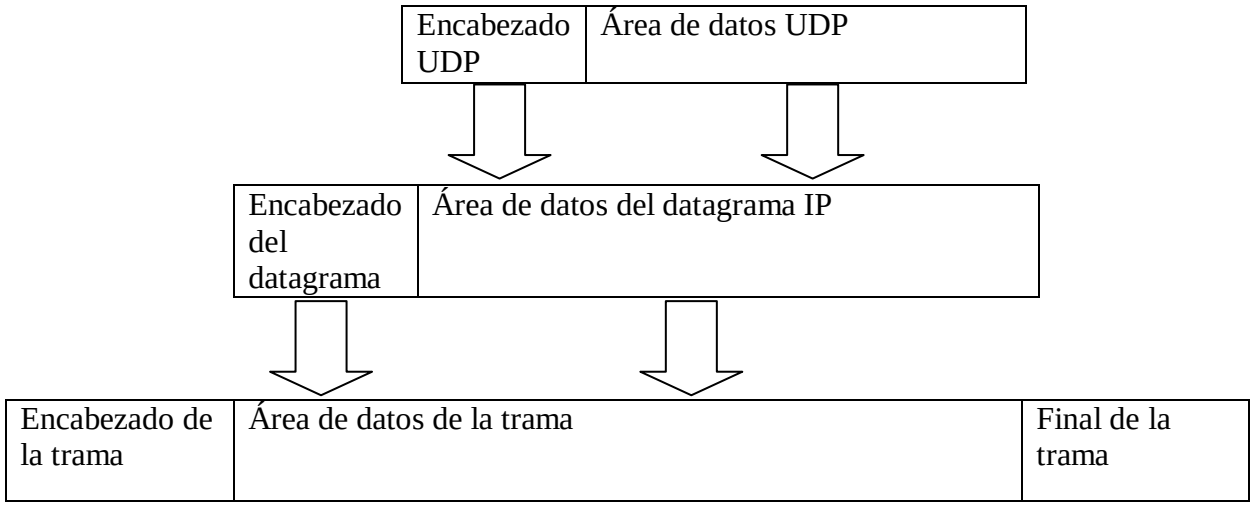
Contiene 2 bytes. Su uso es opcional en IPv4 y obligatorio en IPv6, ya que en ese caso se ha suprimido el checksum a nivel de red. Cuando se envía información en tiempo real su uso puede omitirse. Si la verificación del checksum en el receptor arroja un error, el mensaje es descartado sin notificarlo al nivel de aplicación ni al emisor.

Datos

Contiene los datos a transmitir.

Puerto Origen y Destino

Contiene 2 bytes cada uno, que especifican el puerto de la aplicación que genera y recibe el mensaje. A diferencia de TCP, el campo origen contiene normalmente cero, salvo que la aplicación solicite una respuesta. En el campo destino el puerto 1812 se utiliza para la autenticación y el 1813 para servicios de administración de cuentas.



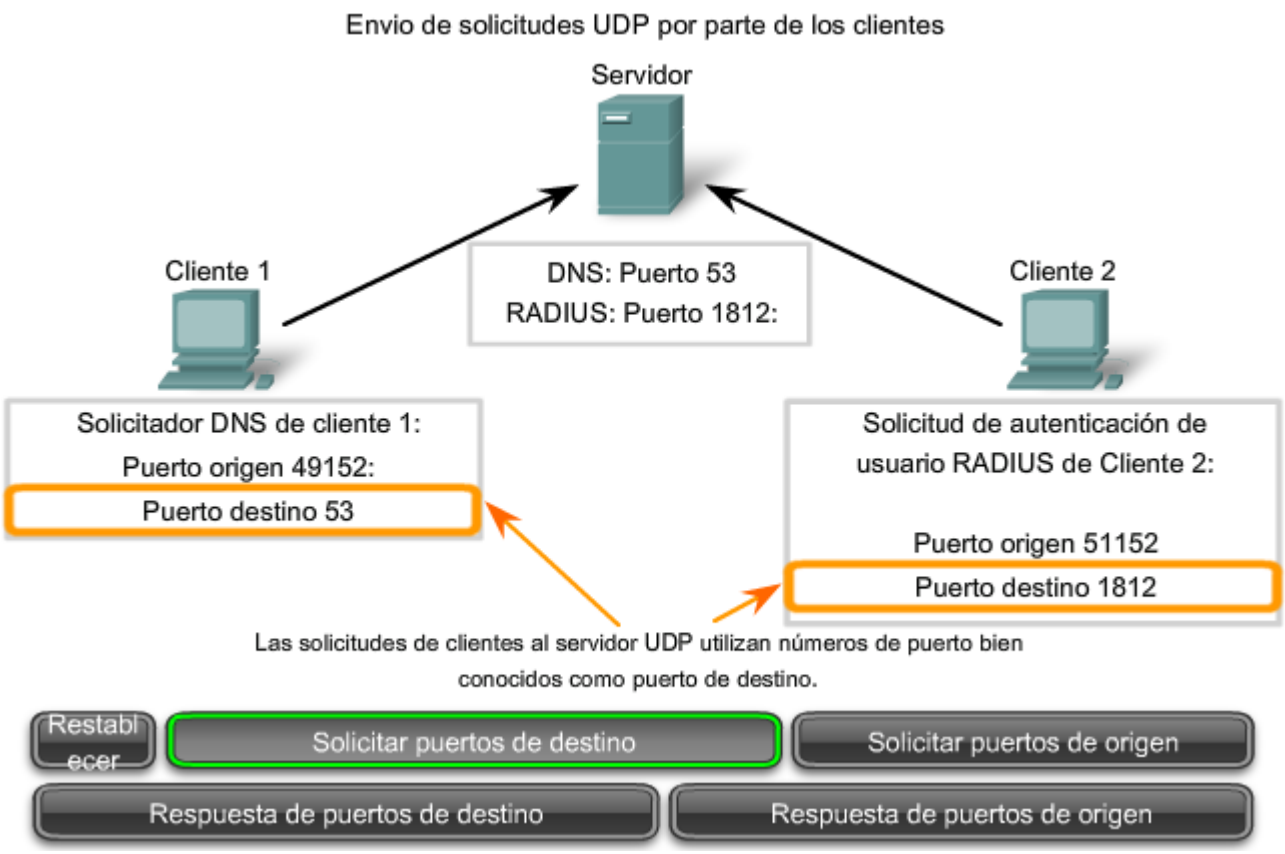


Fig. 16 Solicitudes UDP Clientes

UDP utiliza el protocolo IP para transportar sus mensajes. Como vemos, no añade ninguna mejora en la calidad de la transferencia; aunque sí incorpora los puertos origen y destino en su formato de mensaje. Las aplicaciones deben programarse teniendo en cuenta que la información puede no llegar de forma correcta.

Cuando UDP recibe un datagrama destinado a uno de esos puertos, envía los datos de aplicación a la aplicación adecuada en base a su número de puerto. Cuando el número del campo puerto es 1812 o 1813 contamos con un paquete enviado utilizando el protocolo RADIUS.

❖ Almacenamiento:

Una vez verificado y analizado el Protocolo UDP, se procede al almacenamiento de los campos de interés del datagrama UDP. Se almacenan en la tabla, los puertos origen y destino de la aplicación que genera y que recibe el mensaje.

RADIUS

Los datos se intercambian a través de paquetes RADIUS entre el cliente y el servidor. La siguiente información muestra la estructura del formato del paquete:

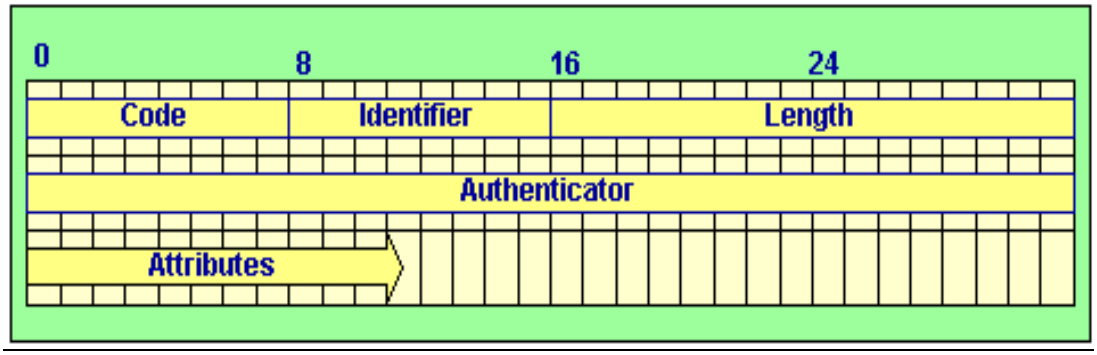


Fig. 17 Formato del paquete RADIUS

❖ Campos del paquete:

➢ Code (Código). El campo Code es un octeto, que identifica el tipo de paquete RADIUS. Cuando se recibe un paquete con un campo de código no válido, es descartado silenciosamente.

Los códigos se asignan de la siguiente manera:

Valor	Descripción
1	Access-Request
2	Access-Accept
3	Access-Reject
4	Accounting-Request
5	Accounting-Response
6	Accounting-Status (now Interim Accounting)
7	Password-Request
8	Password-Ack
9	Password-Reject
10	Accounting-Message
11	Access-Challenge
12	Status-Server (experimental)
13	Status-Client (experimental)
255	Reserved

Fig. 18 Valor y mensaje del campo Code del protocolo RADIUS

❖ Tipos de mensajes RADIUS:

Access-Request: Enviado por un cliente RADIUS para solicitar autenticación y autorización para conectarse a la red. Debe contener el usuario y contraseña (ya sea de usuario o CHAP); además del puerto NAS, si es necesario.

Access-Accept: Enviado por un servidor RADIUS en respuesta a un mensaje de Access-Request. Informa que la conexión está autenticada y autorizada y le envía la información de configuración para comenzar a usar el servicio.

Access-Reject: Enviado por un servidor RADIUS en respuesta a un mensaje de Access-Request. Este mensaje informa al cliente RADIUS que el intento de conexión ha sido rechazado. Un servidor RADIUS envía este mensaje ya sea porque las credenciales no son auténticas o por que el intento de conexión no está autorizado.

Access-Challenge: Envío de un servidor RADIUS en respuesta a un mensaje de Access-Request. Este mensaje es un desafío para el cliente RADIUS. Si este tipo de paquete es soportado, el servidor pide al cliente que vuelva a enviar un paquete Access-Request para hacer la autenticación. En caso de que no sea soportado, se toma como un Access-Reject.

Accounting-Request: Enviado por un cliente RADIUS para especificar información de cuenta para una conexión que fue aceptada.

Accounting-Response: Enviado por un servidor RADIUS en respuesta a un mensaje de Accounting-Request. Este mensaje reconoce el procesamiento y recepción exitosa de un mensaje de Accouting-Response.

- Identifier (Identificador). Es un octeto que permite al cliente RADIUS relacionar una respuesta RADIUS con la solicitud adecuada.

- Length. La longitud del paquete es de 2 octetos. Indica la longitud del paquete incluidos todos los campos.

- Authenticator (Verificador). Valor usado para autenticar la respuesta del servidor RADIUS. Es usado en el algoritmo de encubrimiento de contraseña.

- Attributes (Atributos). Aquí son almacenados un número arbitrario de atributos, que en cada paquete son pares: atributo-valor. Los únicos atributos obligatorios son el User-Name (usuario) y el User-Password (contraseña), además de estos existen otros como los que se muestran a continuación:

Atributos	Código
User-Name	1
User-Password	2
CHAP-Password	3
NAS-IP-Address	4
NAS-Port	5
Service-Type	6
Framed-Protocol	7
Framed-IP-Address	8
Framed-IP-Netmask	9
Framed-Routing	10
Filter-Id	11
Framed-MTU	12
Framed-Compression	13
Login-IP-Host	14
Login-Service	15
Login-TCP-Port	16
(unassigned)	17
Reply-Message	18
Callback-Number	19
Callback-Id	20
(unassigned)	21
Framed-Route	22
Framed-IPX-Network	23
State	24
Class	25
Vendor-Specific	26
Session-Timeout	27
Idle-Timeout	28
Termination-Action	29

Called-Station-Id	30
Calling-Station-Id	31
NAS-Identifier	32
Proxy-State	33
Login-LAT-Service	34
Login-LAT-Node	35
Login-LAT-Group	36
Framed-AppleTalk-Link	37
Framed-AppleTalk-Network	38
Framed-AppleTalk-Zone	39
(reserved for accounting)	40-59
CHAP-Challenge	60
NAS-Port-Type	61
Port-Limit	62
Login-LAT-Port	63

Fig. 19 Atributos y código del protocolo RADIUS

A continuación se expresa el significado de cada uno de los atributos antes presentados:

- User-name: Este atributo indica el nombre del usuario que se va a autenticar.
- User-Password: Atributo que indica la contraseña del usuario a ser autenticado. Sólo se utiliza en los paquetes de solicitud de acceso.
- CHAP-Password: Este atributo indica el valor de la respuesta proporcionada por los usuarios PPP-CHAP en respuesta al desafío. Sólo se utiliza en los paquetes de solicitud de acceso.
- NAS-IP-Address: Atributo que indica la dirección IP que identifica al NAS, la cual es pedida en la autenticación de usuario.
- NAS-Port: Este atributo indica el número de puerto físico del NAS, que es la autenticación del usuario. Sólo se utiliza en los paquetes de solicitud de acceso.

- Service-Type: Este atributo indica el tipo de servicio que el usuario ha pedido o el tipo de servicio que se la va a brindar.
- Framed-Protocol: Atributo que indica cuál protocolo de enmarcado se usa en el acceso. Tiene varios valores, PPP, SLIP.
- Framed-IP-Address: Este atributo indica la dirección a ser configurada por el usuario. Puede ser usado en paquetes Access-Accept.
- Framed-IP-Netmask: Atributo que indica la máscara de red IP que debe estar configurada para el usuario. Puede ser usado en paquetes Access-Accept.
- Framed-Routing: Este atributo indica el método de enrutamiento para el usuario. Sólo se utiliza en paquetes Access-Accept.
- Filter-Id: Atributo que indica el nombre de la lista de filtros para este usuario. En un paquete Access-Accept pueden ser enviados cero o más atributos de este tipo.
- Framed-MTU: Atributo que indica la unidad máxima de transmisión que puede ser configurada por el usuario, cuando esta no es negociada por algún otro medio, como PPP.
- Framed-Compression: Este atributo indica el protocolo de compresión que se utiliza en el enlace. Puede enviarse más de un atributo de protocolo de compresión.
- Login-IP-Host: Este atributo indica la dirección IP del servidor al que el usuario debe conectarse para obtener el servicio especificado en el atributo Login-Service.
- Login-Service: Atributo que indica el tipo de servicio solicitado por el usuario. Puede tener valores como Telnet y Rlogin, entre otros.
- Login-TCP-Port: Este atributo indica el puerto TCP con el que el usuario se conecta, cuando el atributo Login-Service también está presente. Este sólo se utiliza en paquetes Access-Accept.
- Reply-Message: Este atributo indica un mensaje que se le muestra al usuario. Cuando se utiliza en un paquete Access-Accept, es un mensaje de éxito. Cuando se utiliza en un paquete Access-Request, es un mensaje de fracaso.
- Callback-Number: Atributo que indica una cadena de marcación que se utiliza para la devolución de llamadas. Puede ser usado en paquetes Access-Accept.
- Callback-Id: Este atributo indica el nombre de un lugar a ser llamado, a ser interpretado por el NAS. Puede ser usado en paquetes Access-Accept.
- Framed-Route: Atributo que proporciona información de enrutamiento a ser configurada para el usuario en el NAS. Se utiliza en el paquete Access-Accept y pueden aparecer varias veces.

- Framed-IPX-Network: Este atributo indica el número de red IPX a ser configurado para el usuario. Se utiliza en paquetes Access-Accept.
- State: Atributo que se encuentra disponible para ser enviado por el servidor al cliente en un paquete Access-Challenge y debe ser enviado sin modificaciones desde el cliente al servidor en el nuevo Access-Request de respuesta a este reto, en su caso.
- Class: Este atributo se encuentra disponible para ser enviado por el servidor al cliente en un Access-Accept y deben ser enviados sin modificaciones por el cliente para el servidor como parte del paquete de Accounting-Request si la contabilidad es compatible.
- Vendor-Specific: Atributo que se encuentra disponible para permitir a los proveedores apoyar a sus propios atributos extendidos, no es adecuado para uso general. No debe afectar el funcionamiento del protocolo RADIUS.
- Sesion-Timeout: Atributo que fija el número máximo que dura la conexión. Es enviado por el servidor al cliente en un Access-Accept o Access-Challenge.
- Idle-Timeout: Este atributo fija el número máximo de segundos que puede durar una conexión ociosa. Es enviado por el servidor al cliente en un Access-Accept o Access-Challenge.
- Termination-Action: Atributo que indica qué medidas debería tomar el NAS especificado cuando el servicio se haya completado. Sólo se utiliza en paquetes Access-Accept.
- Called-Station-Id: Este atributo permite que el NAS envié en el paquete de solicitud de acceso el número de teléfono al cual el usuario llamó, usando identificación del número marcado (DNIS) o una tecnología similar.
- Calling-Station-Id: Atributo que permite que el NAS envíe en el paquete de solicitud de acceso el número de teléfono desde el que proviene la llamada, usando identificación automática del número (ANI) o una tecnología similar.
- NAS-Identifier: Este atributo contiene una cadena de identificación del NAS originando el Access-Request. Sólo se utiliza en paquetes Access-Request. Los atributos NAS-IP-Address o NAS-identifier deben estar presente en un paquete Access-Request.
- Proxy-State: Atributo que se encuentra disponible para ser enviado por un servidor proxy a otro servidor al enviar un Access-Request y debe ser retomada sin modificaciones en un Access-Accept, Access-Reject o Access-Challenge. Cuando el servidor proxy recibe la respuesta a su solicitud, se debe quitar de su propio Proxy-State (el último Proxy-State en el paquete) antes de transmitir la

respuesta al NAS. Si un atributo Proxy-State se agrega a un paquete cuando se transmitió, el atributo Proxy-State debe ser añadido después de cualquier atributo Proxy-State existente.

- Login-LAT-Service: Este atributo indica el sistema con el que el usuario se conecta por LAT. Puede ser usado en paquetes Access-Accept, pero solamente cuando LAT es especificado como Login-Service. Puede ser usado en un paquete Access-Request como una pista para el servidor, pero el servidor no está obligado a honrar la pista.
- Login-LAT-Node: Atributo que indica el Nodo con el que el usuario se conecta automáticamente por LAT. Puede ser usado en paquetes Access-Accept, pero sólo cuando LAT se especifica como Login-Service. Puede ser usado en un paquete Access-Request como una pista para el servidor, pero el servidor no está obligado a honrar la pista.
- Login-LAT-Group: Este atributo contiene una cadena de identificación de los códigos de grupo de LAT que este usuario está autorizado a utilizar. Puede ser usado en paquetes Access-Accept, pero sólo cuando se especifica LAT como Login-Service. Puede ser usado en un paquete Access-Request como una pista para el servidor, pero el servidor no está obligado a honrar la pista.
- Framed-AppleTalk-Link: Atributo que indica el número de red AppleTalk que se debe utilizar para el enlace serie del usuario, que es otro Enrutador AppleTalk. Sólo se utiliza en paquetes Access-Accept.
- Framed-AppleTalk-Network: Este atributo indica el número de red AppleTalk que se debe utilizar para el enlace serie de usuario, que es otro Enrutador AppleTalk. Sólo se utiliza en paquetes Access-Accept.
- Framed-AppleTalk-Zone: Este atributo indica la zona de AppleTalk predeterminada que se utiliza para este usuario. Sólo se emplea en paquetes Access-Accept. No se permiten múltiples casos de este atributo en el mismo paquete.
- CHAP-Challenge: Atributo que contiene el CHAP-Challenge enviado por el NAS al usuario. Sólo se utiliza en paquetes Access-Request.
- NAS-Port-Type: Atributo que indica el tipo de puerto físico del NAS, que es la autenticación del usuario. Se puede utilizar en lugar de o además del atributo NAS-Port (5). Sólo se emplea en paquetes Access-Request. El NAS-Port (5) o NAS-Port-Tipo o ambos deben estar presentes en un paquete Access-Request, si el NAS diferencia entre sus puertos.
- Port-Limit: Este atributo establece el número máximo de puertos que debe facilitarse al usuario por el NAS. Puede ser enviado por el servidor al cliente en un paquete Access-Accept.

- Login-LAT-Port: Atributo que indica el puerto con el que el usuario va a ser conectado por LAT. Puede ser usado en paquetes Access-Accept, pero sólo cuando LAT se especifica como Login-Service. Puede ser usado en paquetes Access-Request como una pista para el servidor, pero el servidor no está obligado a honrar la pista.

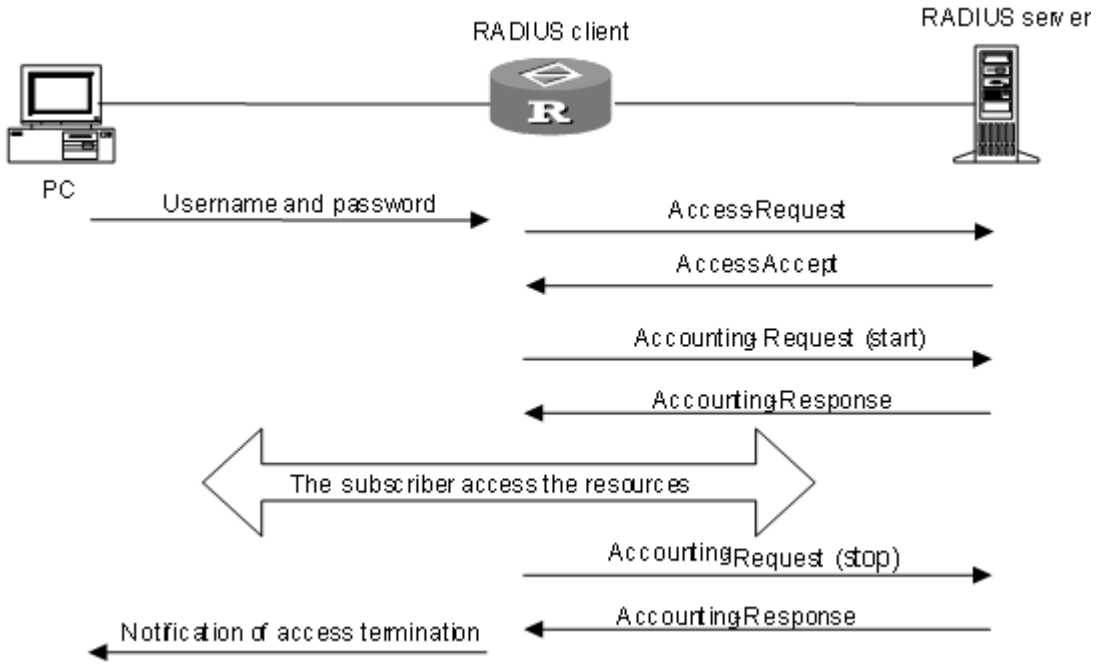


Fig. 20 Tráfico de mensajes del protocolo RADIUS

Sigue la siguiente secuencia:

1) El cliente envía su usuario/contraseña en un paquete que utiliza el puerto UDP 1812, esta información es encriptada con una llave secreta y enviada en un mensaje Access-Request, que tiene un 1 en su código, al servidor RADIUS, esta paso es nombrado Fase de Autenticación.

2) La relación usuario/contraseña es verificada en una base de datos, si es correcta, entonces el servidor envía un mensaje de aceptación, Access-Accept, que tiene un 2 en el código del

mensaje, con información extra (Por ejemplo: dirección IP, máscara de red, tiempo de sesión permitido, etc.) lo que es conocido como Fase de Autorización.

3) En caso de que la relación usuario/contraseña sea incorrecta, entonces el servidor envía un mensaje de rechazo, Access-Reject, que tiene un 3 en el código del mensaje, informándole al cliente que alguna de sus credenciales no son válidas.

4) Si se logró la autenticación, el cliente ahora envía un mensaje de Accounting-Request (Start) con la información correspondiente a su cuenta y para indicar que el usuario está reconocido dentro de la red, esta paso es nombrado Fase de Contabilidad.

5) El servidor RADIUS responde con un mensaje Accounting-Response, cuando la información de la cuenta es almacenada.

6) Cuando el usuario ha sido identificado, éste puede acceder a los servicios proporcionados. Finalmente, cuando desee desconectarse, enviará un mensaje de Accounting-Request (Stop) con la siguiente información:

- Delay Time: Tiempo que el cliente lleva tratando de enviar el mensaje.
- Input Octets: Número de octetos recibido por el usuario.
- Output Octets: Número de octetos enviados por el usuario.
- Session Time: Número de segundos que el usuario ha estado conectado.
- Input Packets: Cantidad de paquetes recibidos por el usuario.
- Output Packets: Cantidad de paquetes enviados por el usuario.
- Reason: Razón por la que el usuario se desconecta de la red.

7) El servidor RADIUS responde con un mensaje de Accounting-Response cuando la información de cuenta es almacenada.

❖ Almacenamiento:

Una vez verificado y analizado el Protocolo RADIUS se procede al almacenamiento de los datos existentes en los campos User_name, NAS_IP_address, Service_Type, Sesion_Timeout, Loging_Service del campo Attributes en los campos User_name, NAS_IP_address, Service_Type, Sesion_Timeout, Loging_Service de la tabla TCP_IP de la base de datos radius respectivamente.

Conclusiones

Se realizó el procedimiento de captura, procesamiento y almacenamiento en línea de los datos más importantes que se intercambian, para los protocolos DIAMETER y RADIUS.

Se profundizó en el estudio de dichos protocolos y su inserción dentro del modelo TCP/IP, además de analizar el estándar que regula su funcionamiento, recogido en los RFC 3588, 3589, 2865 y 2866.

CAPÍTULO 3: PRUEBAS DE LABORATORIO

Introducción

En el presente capítulo se realizan las pruebas de laboratorios pertinentes, para comprobar el correcto funcionamiento del procedimiento descrito en el capítulo anterior. Con la captura de los paquetes que transitan por la red podemos conocer y analizar el funcionamiento de los protocolos implicados, incluyendo RADIUS, también se pueden visualizar los datos que viajan en los distintos campos de dichos protocolos. Esto se realiza con el fin de llevar un completo monitoreo de los eventos y transacciones que ocurren dentro de la red y bajo el protocolo.

3.1 Descripción de las pruebas realizadas en el laboratorio a los protocolos AAA más utilizados.

Primero se realizó el estudio pertinente para un mejor entendimiento del uso de las redes, así como un profundo análisis de los protocolos y servicios más utilizados a nivel mundial en las distintas capas del Modelo TCP/IP. Una vez cumplido lo anteriormente mencionado se procedió a realizar las prácticas de laboratorio correspondientes a esta etapa, logrando así mantener correspondencia con el grupo de tareas planteadas al inicio del trabajo. Para lograr realizar las pruebas se necesitaron dos elementos fundamentales, el cliente y el servidor, se realizaron además varios pasos consecuentes para obtener unas pruebas eficientes.

1er Paso:
Primeramente se realizó la configuración del servidor RADIUS y el cliente ya estaba preparado para efectuar la conexión, la cual se inicia con la petición del mismo al servidor.

2do Paso:
Realizada la petición se procedió a la captura de los protocolos. Para realizar la captura de ellos se empleó el analizador de protocolos de red o sniffer Wireshark, con este se efectuó el análisis y visualización del funcionamiento de los protocolos en la red. Wireshark analiza los paquetes que son transmitidos por la red, además visualiza los datos de estos paquetes de forma específica. Este analizador muestra protocolos que están ejecutándose en ese momento en la red. Los datos de los protocolos

capturados que muestra la mencionada herramienta son: número del paquete, el tiempo de captura, dirección IP origen, dirección IP destino, protocolo y los datos que transportó el paquete.

Con la captura de todos los paquetes que ha encontrado el analizador, se ha acumulado gran cantidad de información que no es la que interesa analizar, haciendo de esta forma más difícil aparentemente nuestro trabajo. Una de las facilidades del Wireshark es que permite realizar filtros para de esta forma hacer capturas al protocolo que verdaderamente es de interés.

De esta manera se procede a realizar algunas capturas al protocolo RADIUS, colocando su nombre en el filtro. En la siguiente imagen se puede apreciar un filtro que se realizó para capturar solamente el protocolo RADIUS. Se observa también los campos anteriormente mencionados y en la información que transportó el paquete se muestran varios de los mensajes, con sus respectivos códigos, que emplea este protocolo para sus labores.

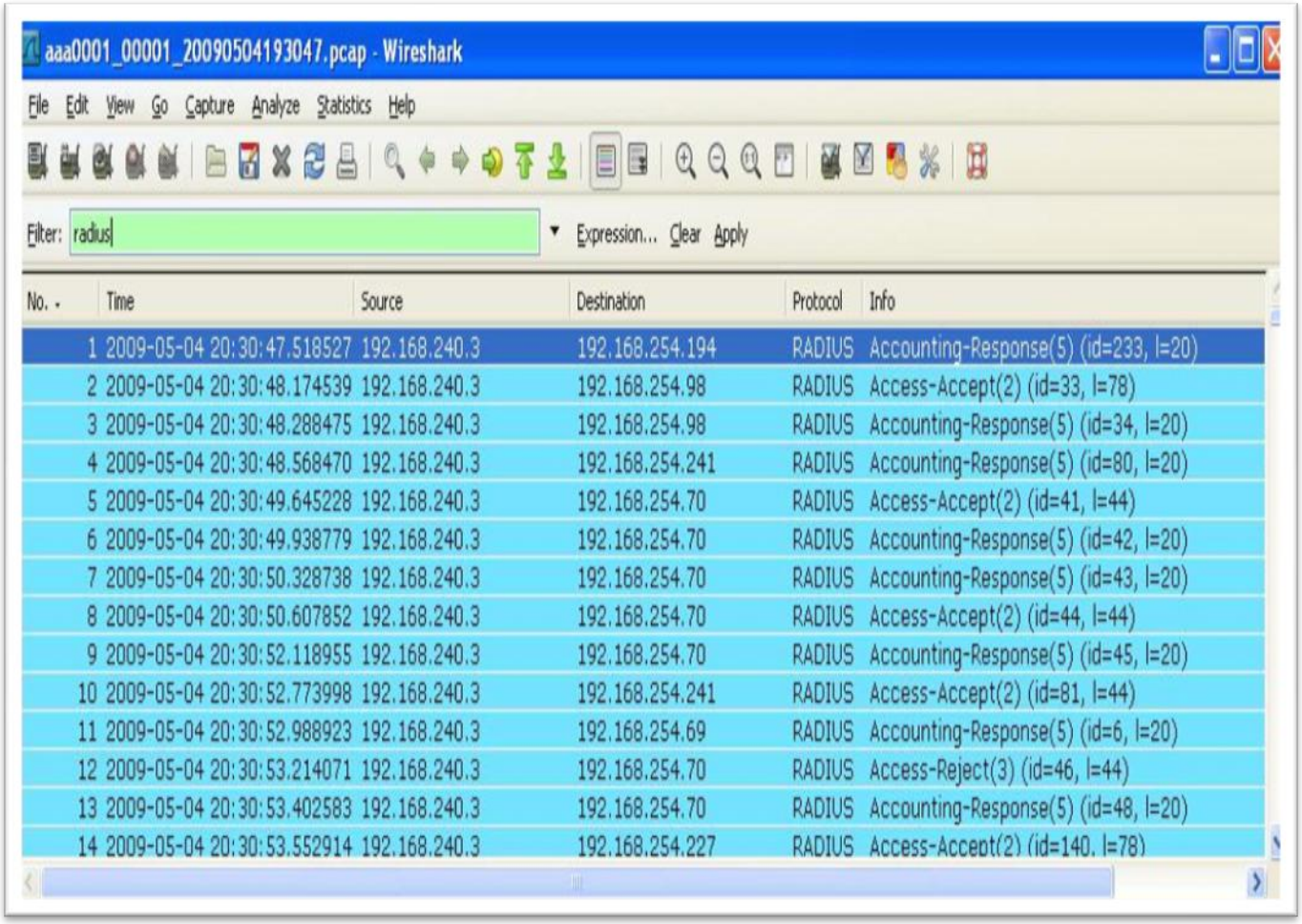


Fig. 21 Filtro del protocolo RADIUS

3er Paso:

A continuación se desglosan cada uno de los protocolos con los campos que utilizan. El primero en analizar es el protocolo Ethernet, el cual tiene entre sus campos más importantes la dirección MAC origen y destino, que indican quien envía la información y hacia quién, y el campo Longitud/Tipo, que define el tamaño de todo lo que contiene y el protocolo que le sigue en la pila, estos se muestran en la siguiente imagen.

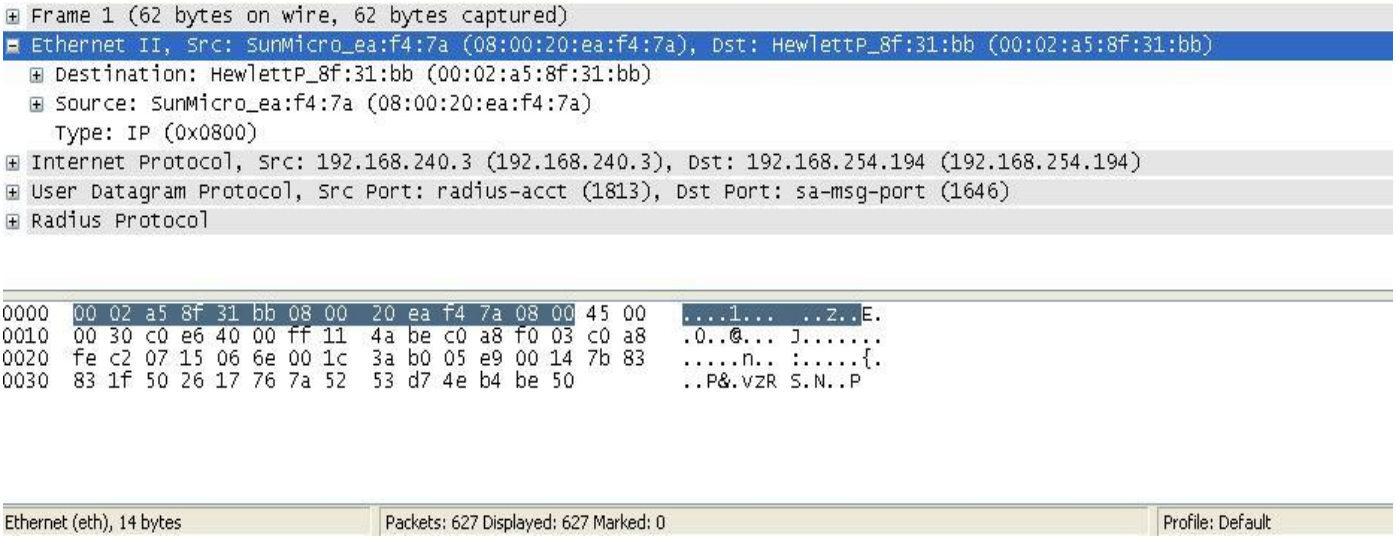


Fig. 22 Ethernet

El campo Type indica el tipo de paquete que se está transmitiendo, a continuación se muestra su descripción en bytes.

.

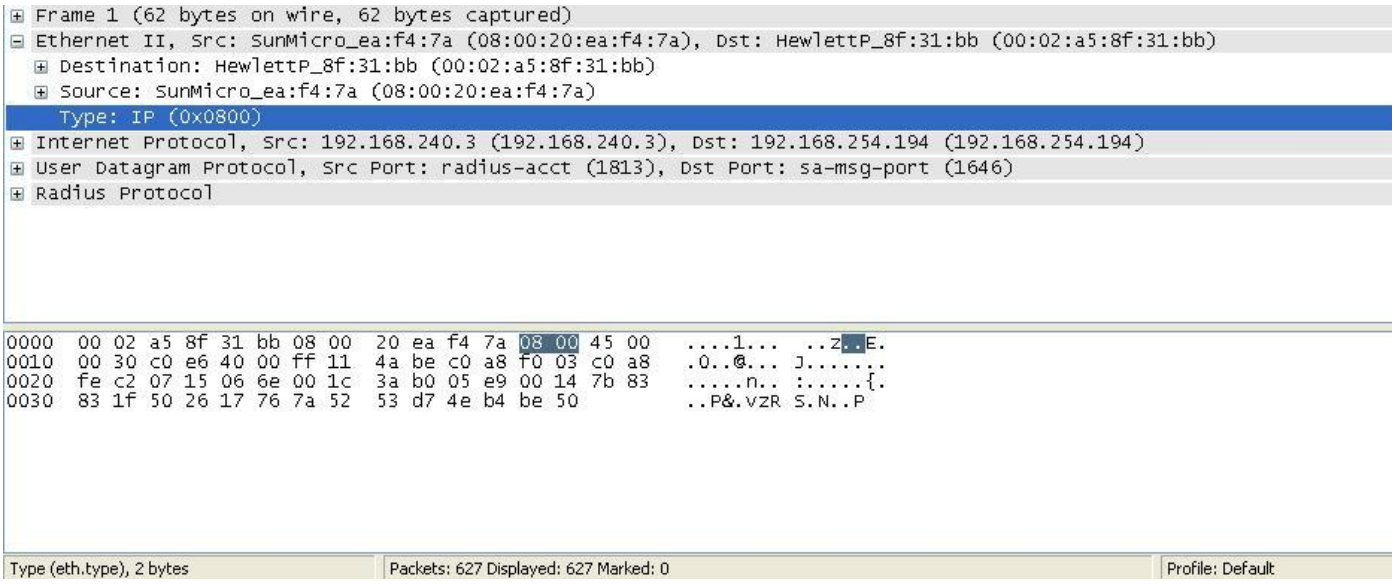


Fig. 23 Campo Type

Como bien indica el campo Type en la imagen, el protocolo que sigue es el IP encargado del direccionamiento y enrutamiento del paquete, que cuenta con campos claves como la versión utilizada, las direcciones IP origen y destino, el tiempo de vida y el campo protocolo que especifica cual es el que continúa en la capa de transporte. La imagen siguiente muestra lo antes explicado.

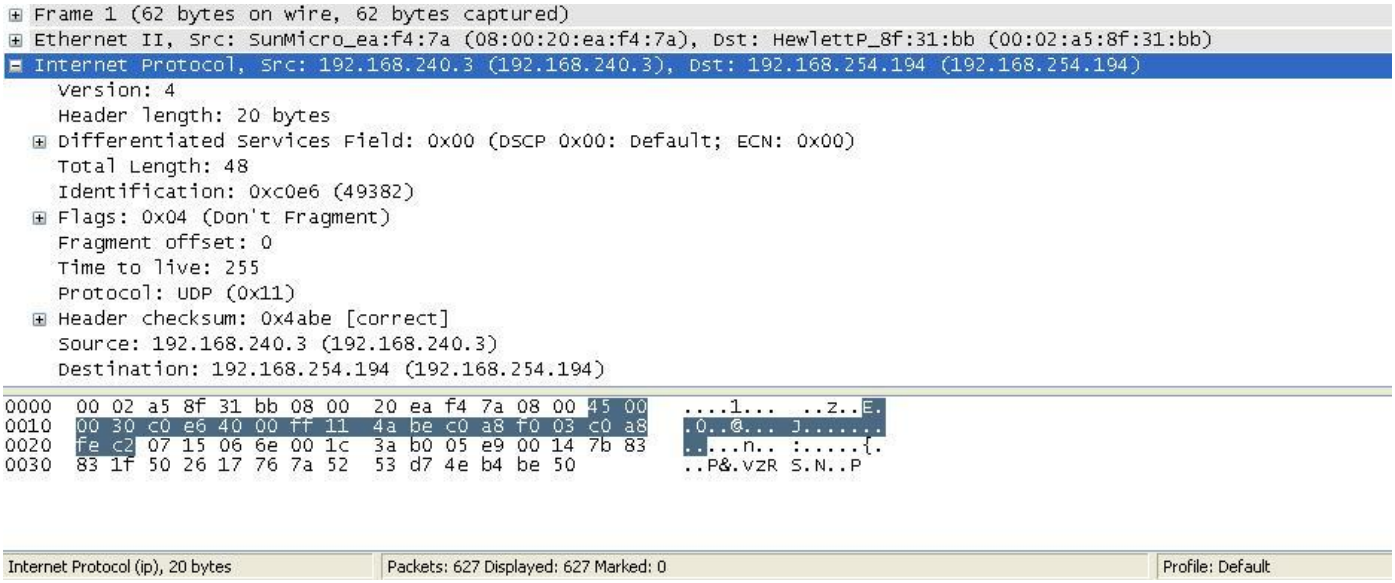


Fig. 24 IP

En la anterior imagen se evidencia la captura realizada al protocolo IP. Donde el campo más importante para el procedimiento descrito en el capítulo anterior, es el campo Protocolo que indica que protocolo de transporte se utiliza para la transmisión del paquete en cuestión.

Los campos de este protocolo con sus datos representados en bytes, de interés para el procedimiento descrito, se muestran a continuación.

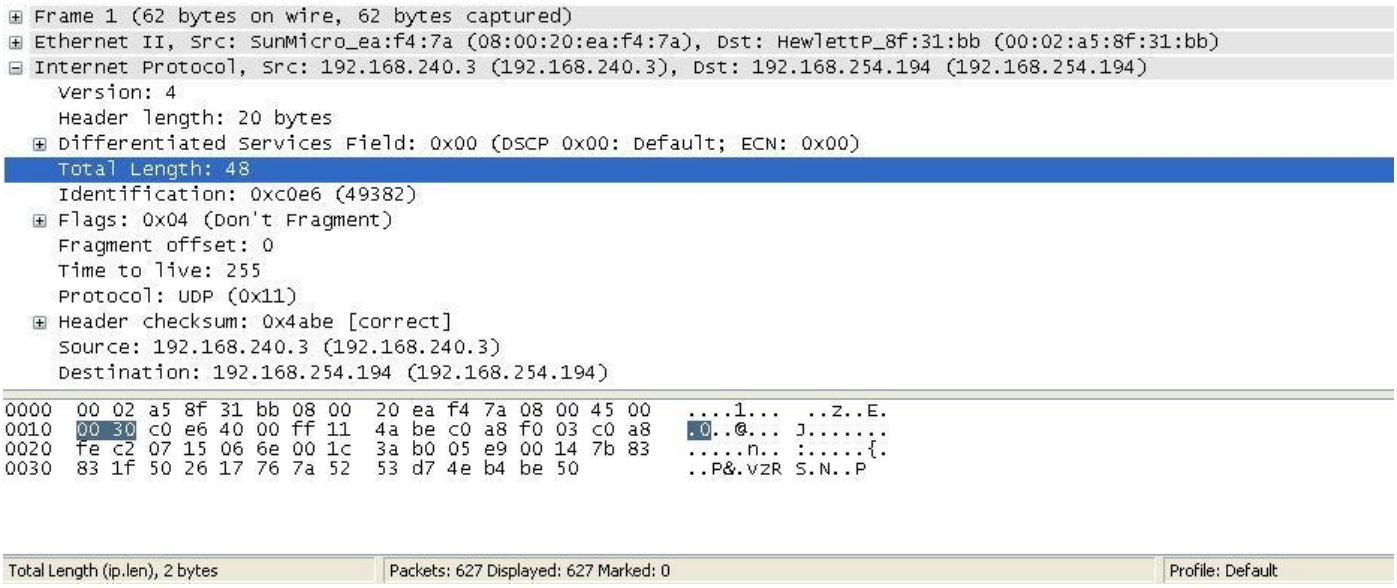


Fig. 25 Campo Total Length

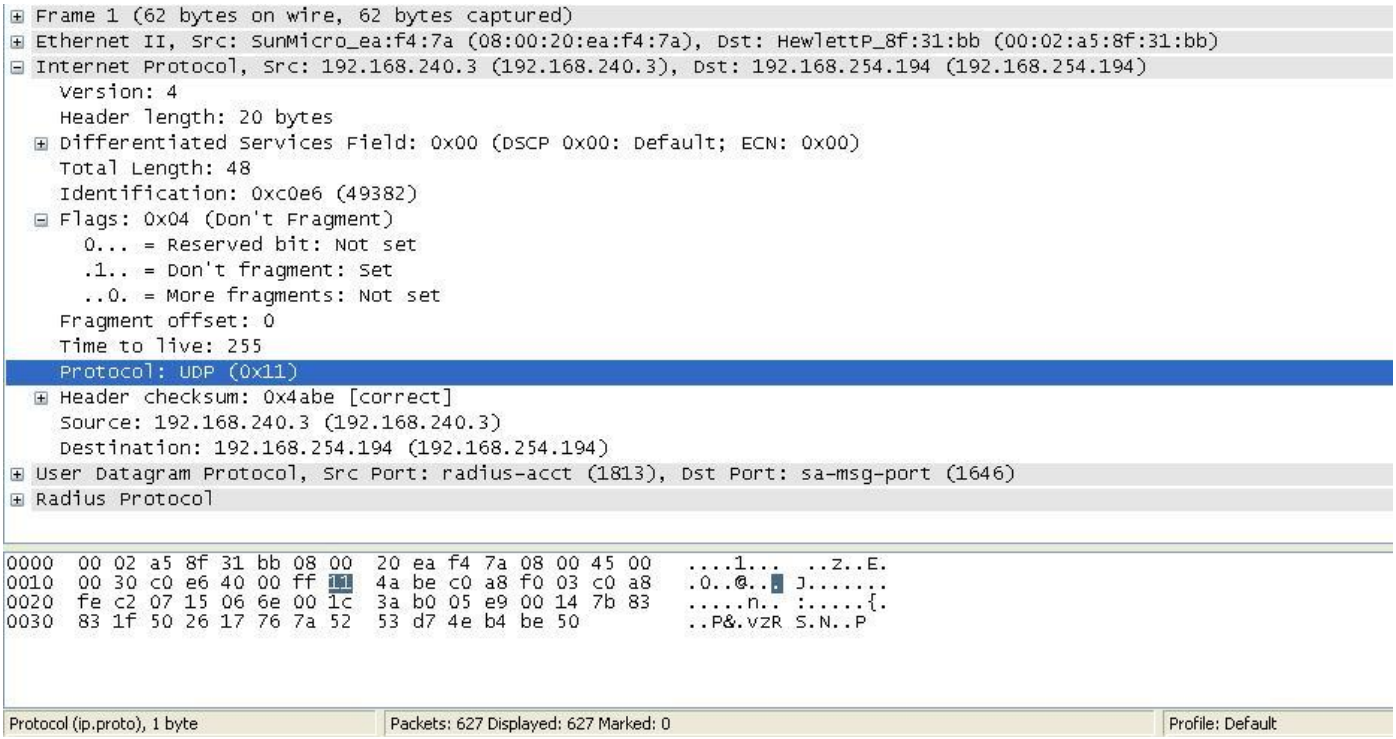


Fig. 26 Campo Protocol

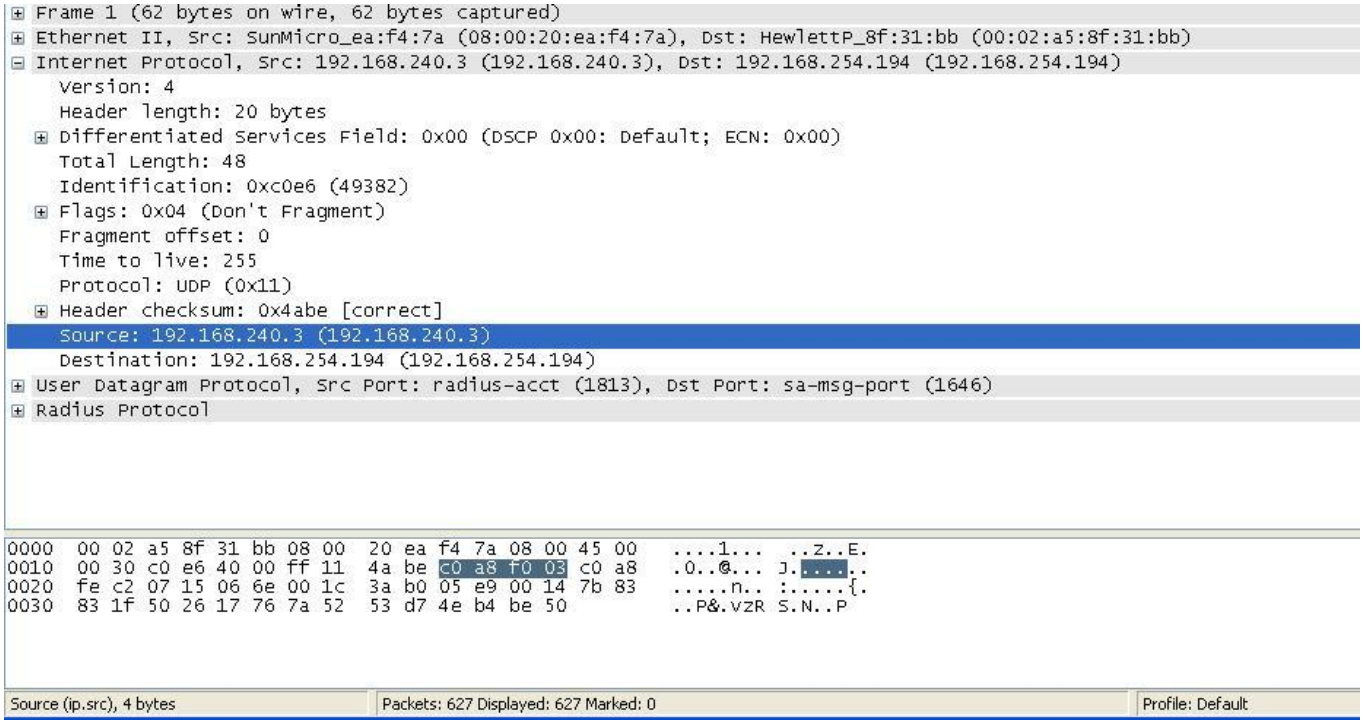


Fig. 27 Campo Source

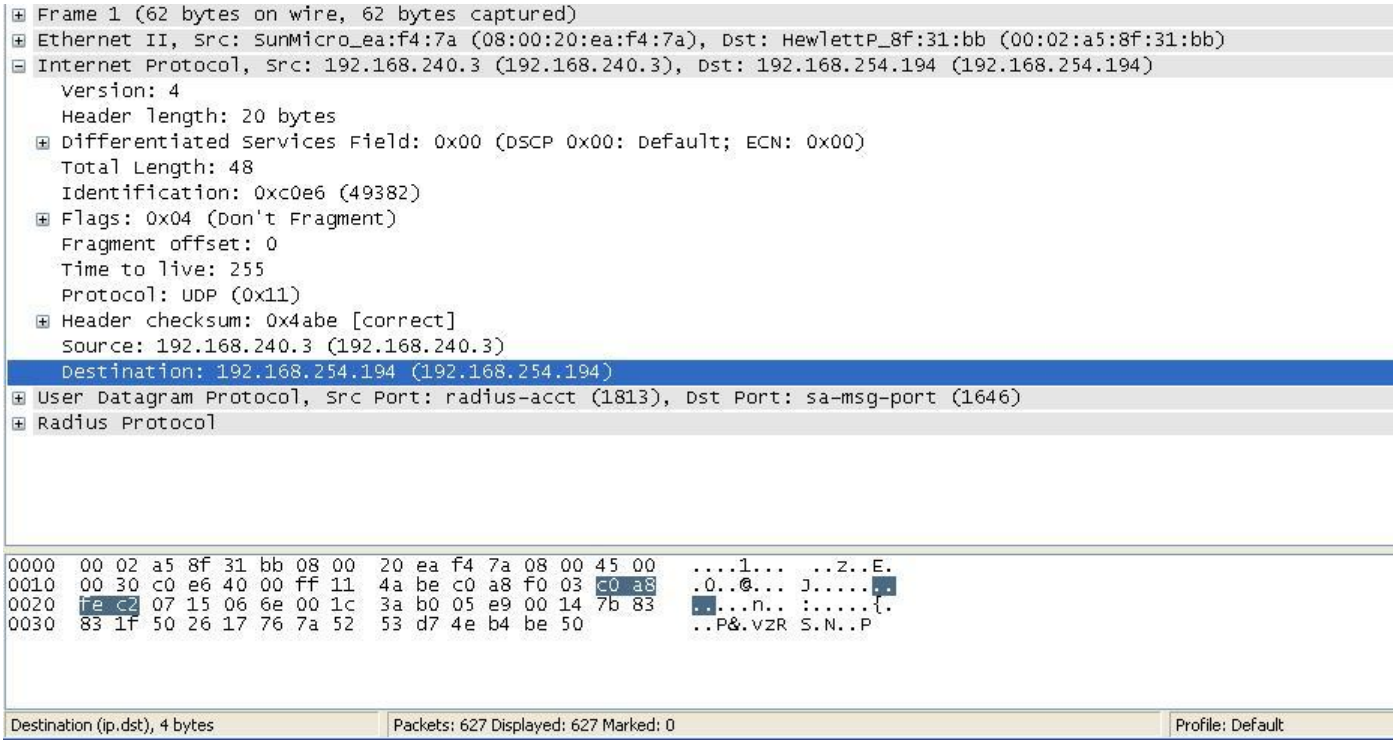


Fig. 28 Campo Destination

En dependencia del tipo de servicio AAA utilizado es el tipo de protocolo de transporte que se emplea, porque como se ha planteado anteriormente, el protocolo DIAMETER trabaja con TCP y el RADIUS usa UDP, por tanto, el protocolo UDP un poco más simple, que es el que utiliza RADIUS, tiene entre sus campos más importantes el puerto de origen y destino, la siguiente imagen presenta estos campos que él utiliza.

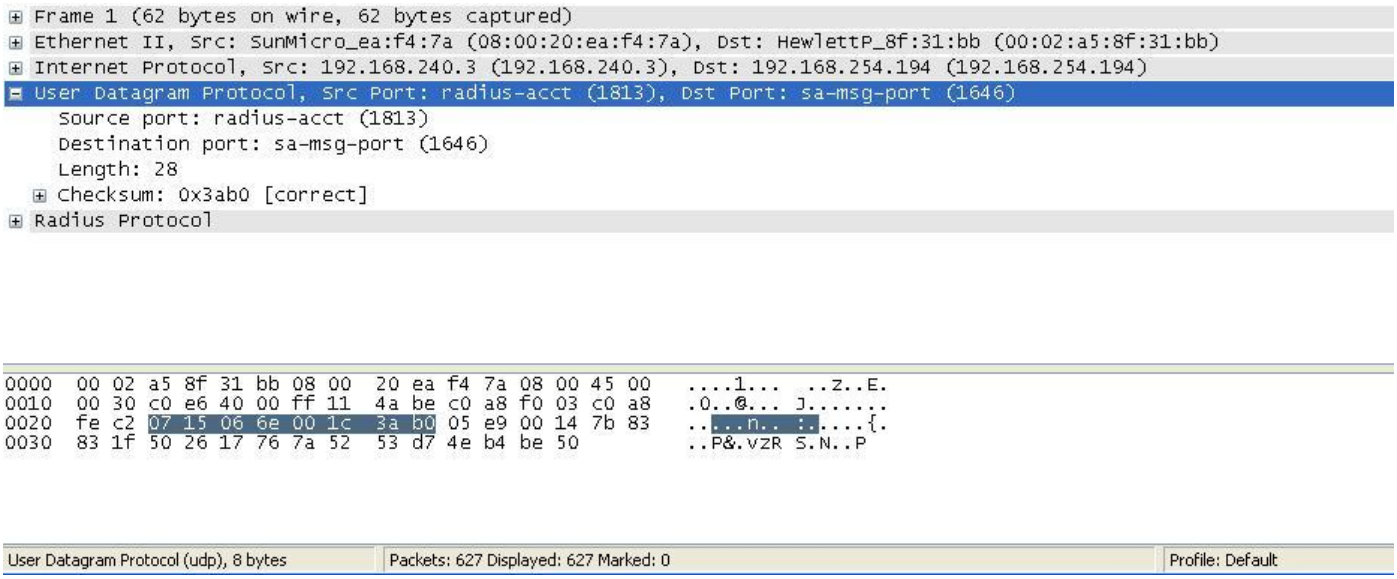


Fig. 29 Datagrama UDP

A continuación se muestran los campos de este protocolo, de interés para el procedimiento descrito con sus datos representados en bytes.

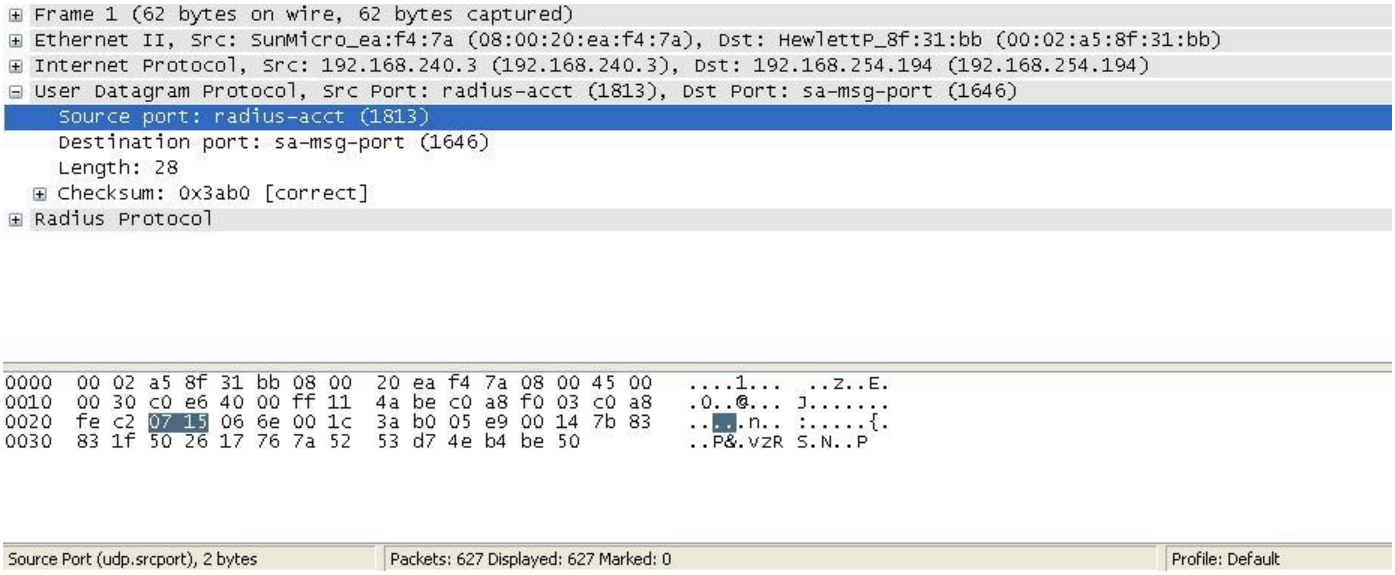


Fig. 30 Campo Source Port

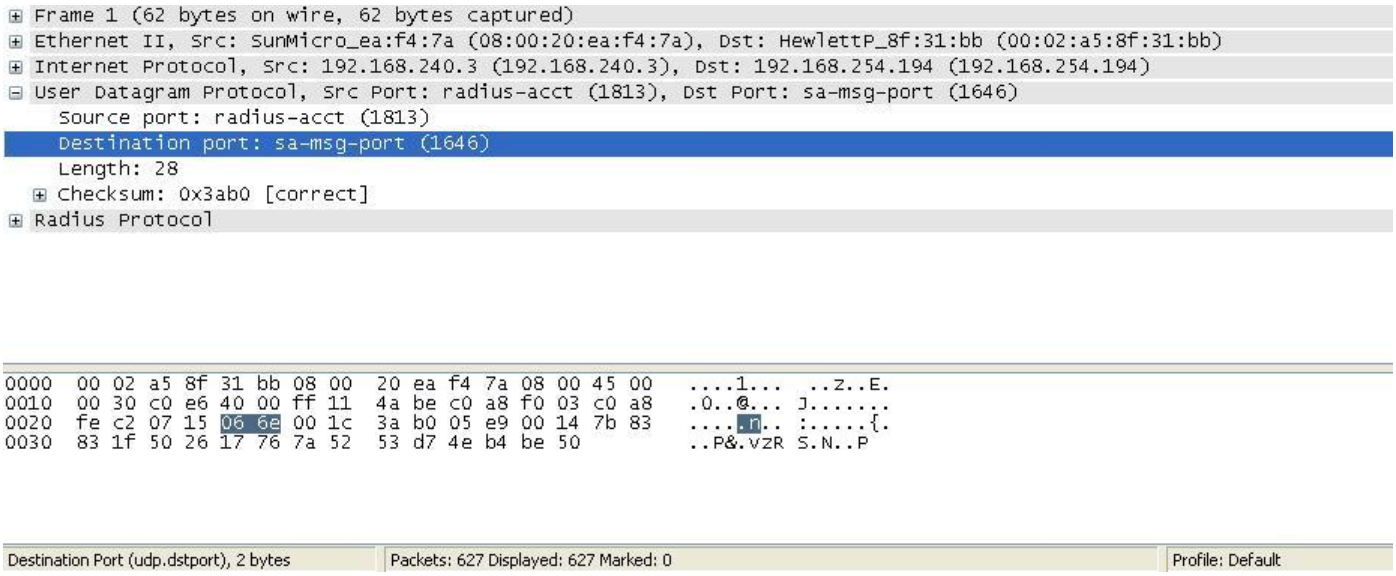


Fig. 31 Campo Destination Port

Como indica el campo puerto de origen en el datagrama UDP, el protocolo que se utiliza para el servicio AAA es el RADIUS, a continuación se muestra una imagen con los campos que tiene este protocolo.

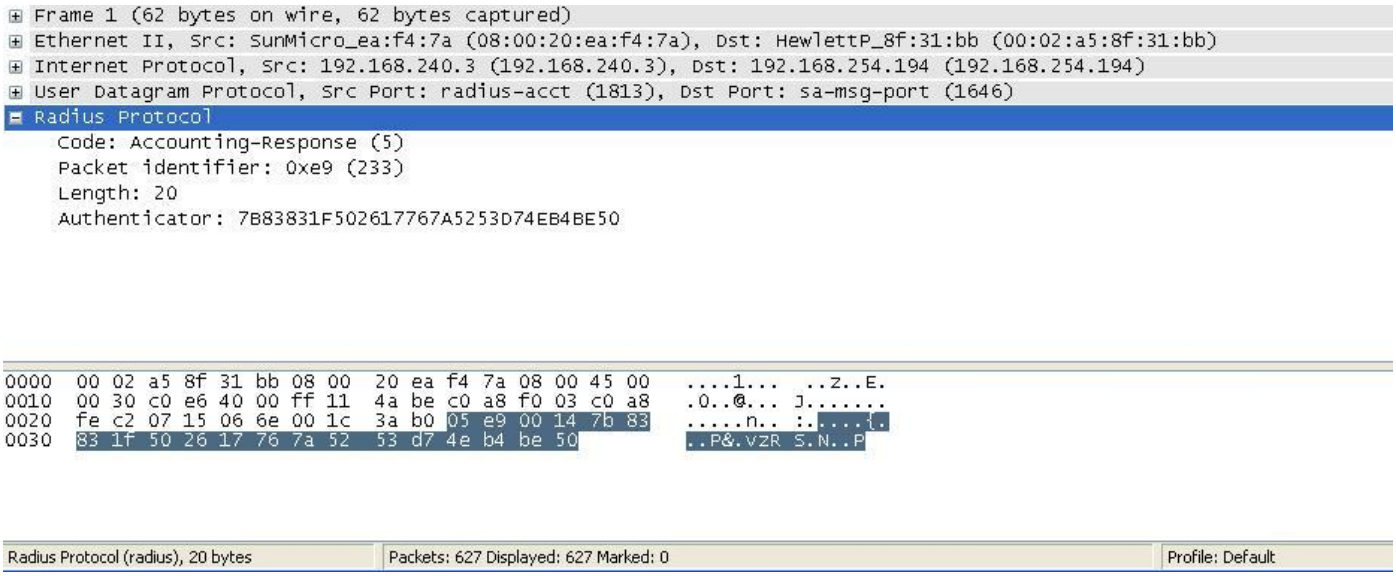


Fig. 32 Segmento RADIUS

Cada campo de este protocolo con sus datos representados en bytes, se muestran a continuación.

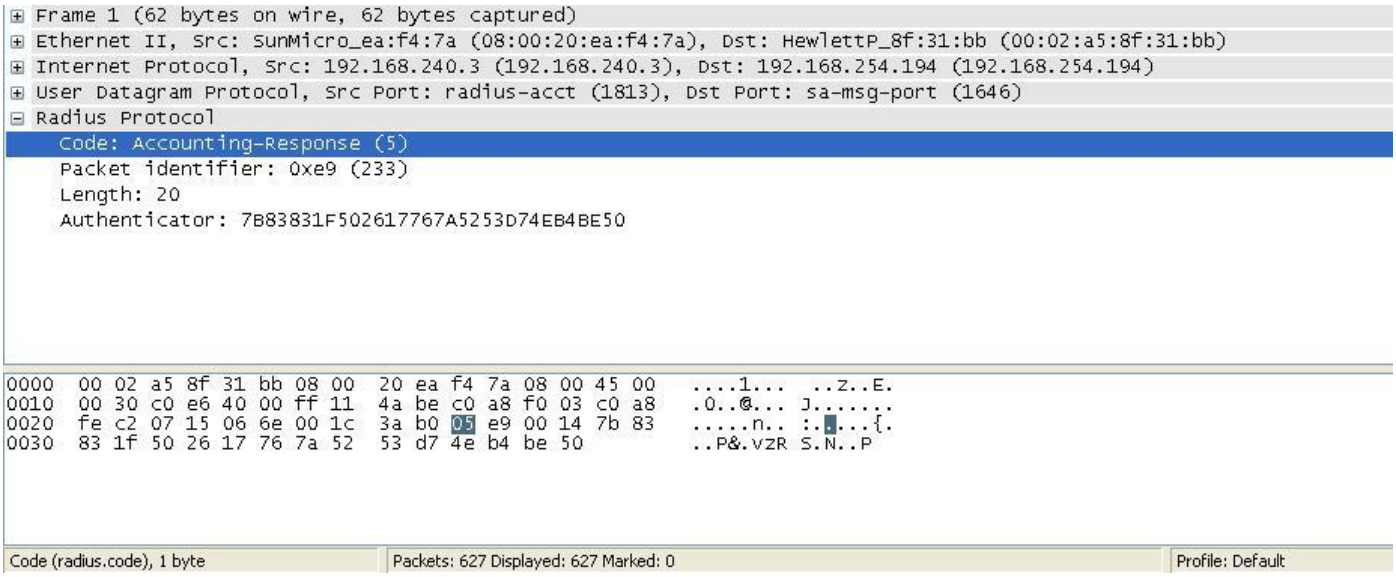


Fig. 33 Campo Code del mensaje Accounting-Response

La imagen anterior muestra uno de los mensajes que son intercambiados entre cliente y servidor RADIUS, en este caso el Accounting-Response con código 5, a continuación se presentan otros ejemplos de mensajes como el Access-Accept con código 2. En la imagen se observa la representación en bytes del protocolo RADIUS en el momento que el servidor acepta una solitud de acceso.

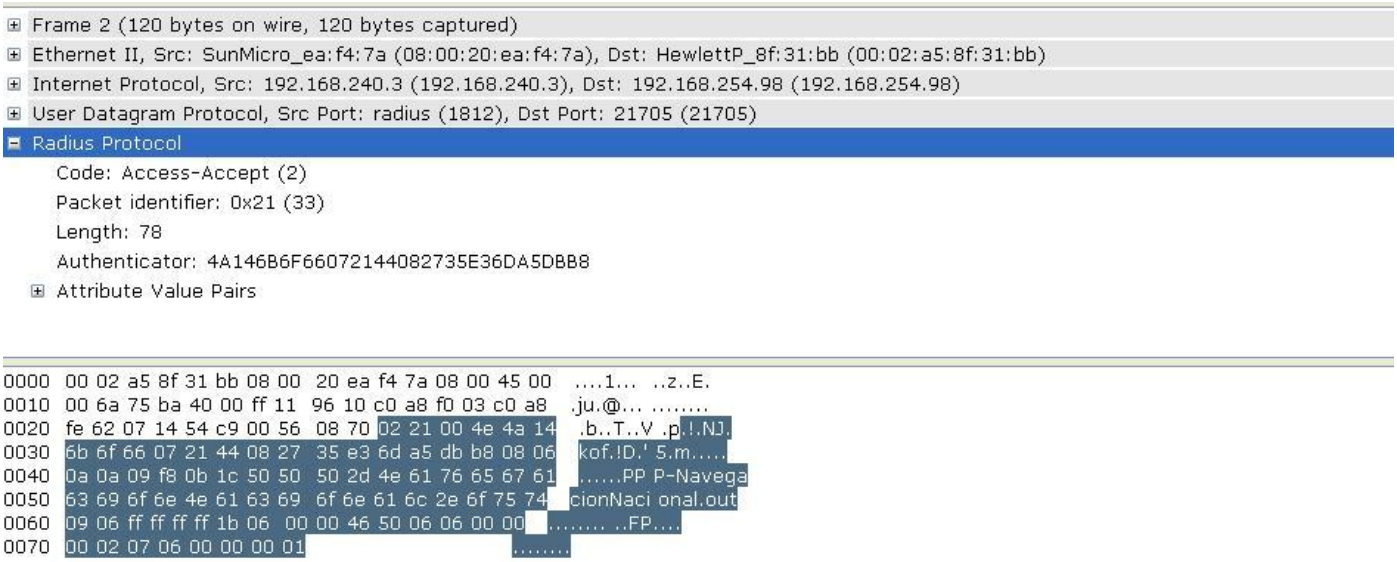


Fig. 34 Segmento RADIUS

En la imagen que sigue se muestra la representación en bytes del mensaje Access-Accept.

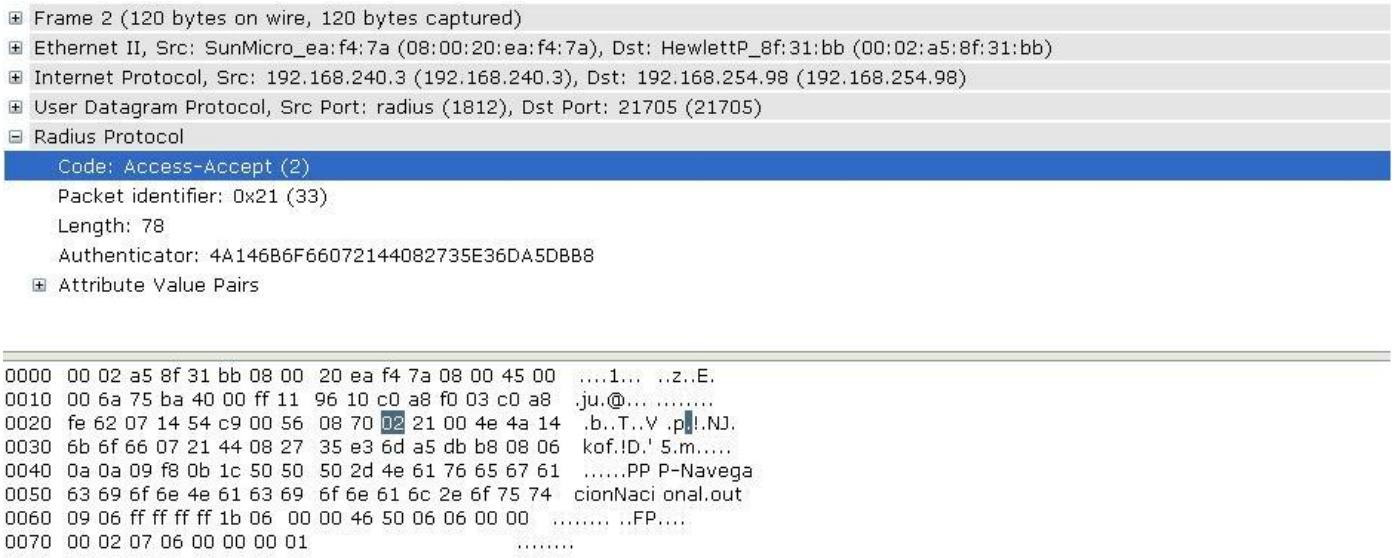


Fig. 35 Campo Code del mensaje Access-Accept

A continuación se visualiza el segmento del protocolo RADIUS en el caso que se envía un rechazo a una solicitud de acceso.

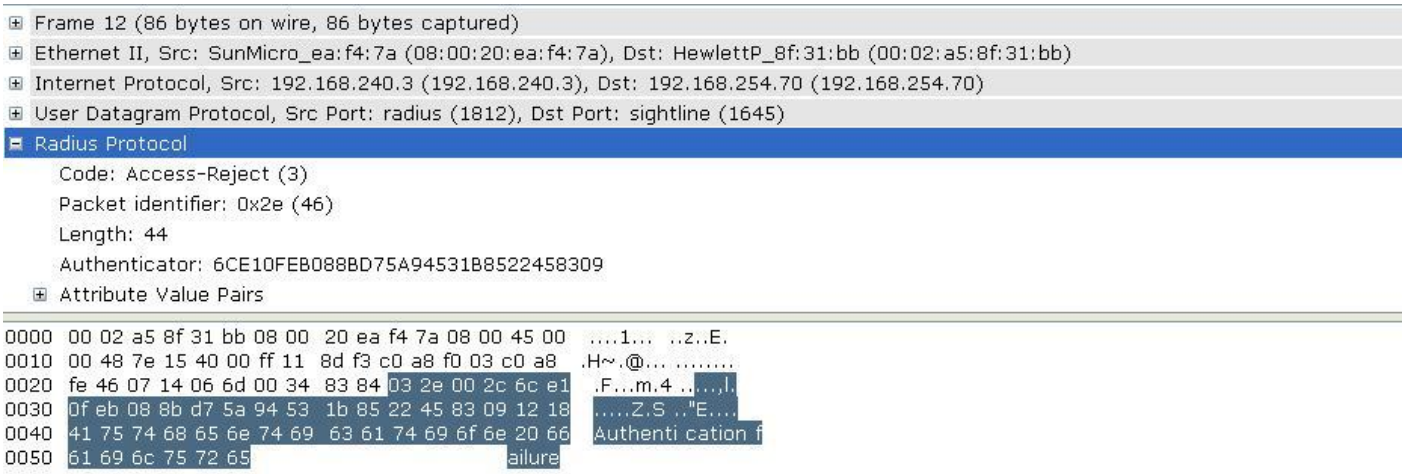


Fig. 36 Segmento RADIUS

En la siguiente imagen se muestra el campo Code en bytes para el caso de un Access-Reject.

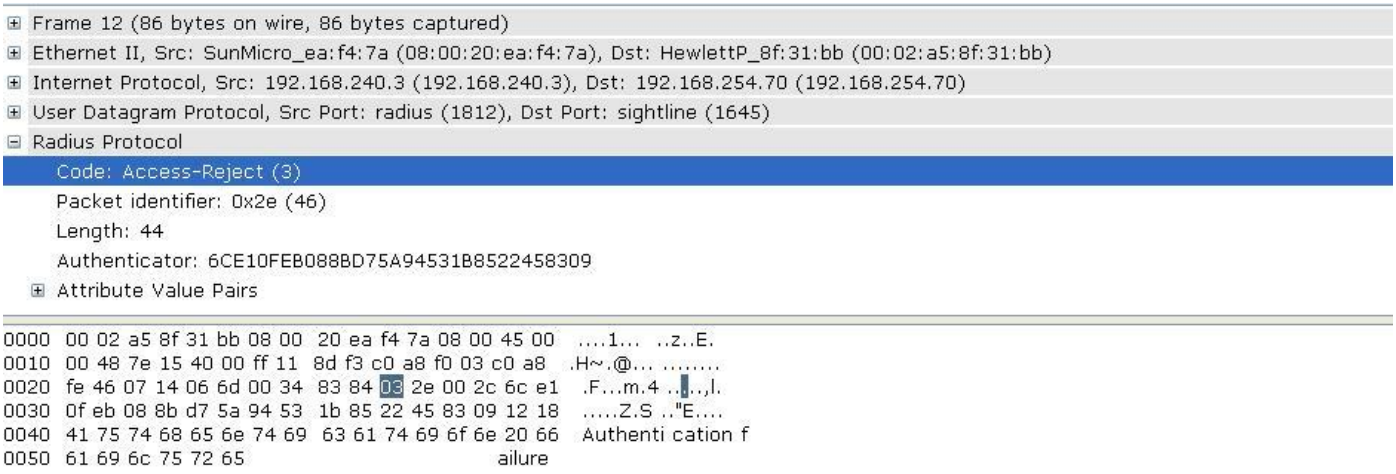
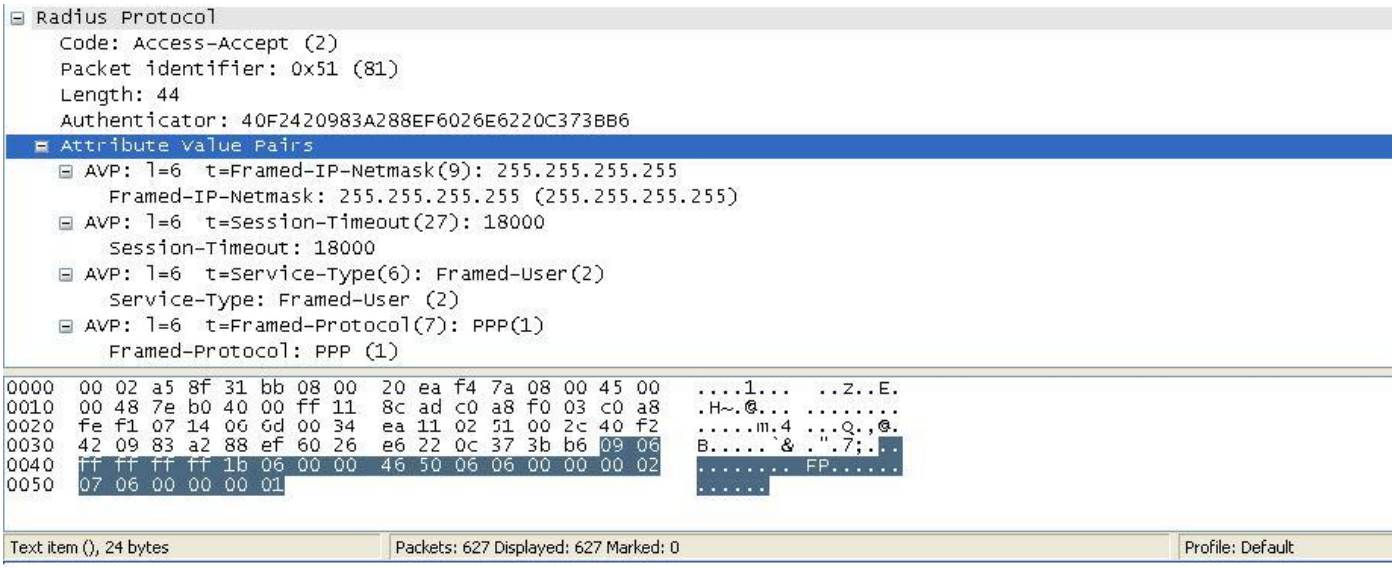
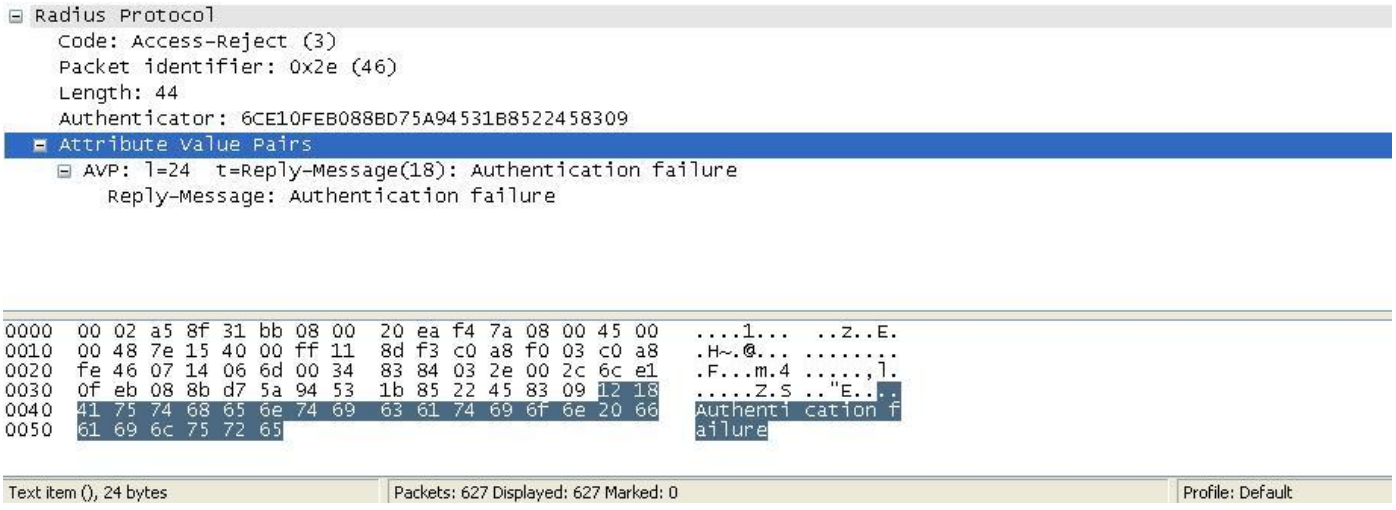


Fig. 37 Campo Code del mensaje Access-Reject

El campo attribute guarda varios elementos importantes para la autenticación como es el caso que se muestra en la siguiente imagen de un mensaje Access-Accept.



Igual sucede con la imagen que a continuación se pone de ejemplo de un mensaje de Access-Reject, en la cual se observan los atributos que utiliza cuando es rechazado un intento de autenticación.



CONCLUSIONES GENERALES

Se realizó el estudio de los modelos en capas OSI y TCP/IP, particularmente este último en el que está basada la investigación, permitió conocer detalladamente la representación del funcionamiento de una red.

El análisis plasmado de los protocolos propició los aspectos fundamentales para tener la idea más exacta posible de cómo se produce la comunicación entre ellos.

Se realizó el procedimiento, y se espera que sea de gran aporte al conocimiento de las personas que les interese el tema o simplemente quieran tener información acerca del mismo. Sobretodo, que sirva de mucha utilidad para el MININT, para el cual fue realizado y tribute a suplir las necesidades que demanda hoy este organismo.

RECOMENDACIONES

Se recomienda que este trabajo se utilice como herramienta para la ingeniería de software de internet.

Que el procedimiento diseñado se utilice para la implementación de aplicaciones de análisis de las trazas suministradas por los protocolos AAA, especialmente el RADIUS.

BIBLIOGRAFÍA

(1) aLaide.com - RFC - RFC 2865 - Remote Authentication Dial In User Service (RADIUS). Available from World Wide Web: <http://www.alaide.com/rfc.php?doc_id=RFC2865>.

(2) capitulo6.pdf (application/pdf Object). Available from World Wide Web: <http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/mayoral_p_e/capitulo6.pdf>.

(3) Comparativa TACACS+ y RADIUS | Debian & Comunicación. Available from World Wide Web: <http://brixtoncat.esdebian.org/27318/comparativa-tacacs-radius>.

(4) El modelo TCP/IP. Available from world wide web: <http://technet.microsoft.com/es-es/library/cc786900(WS.10).aspx>.

(5) Manual_Usuario_Servidor_Radius.pdf (application/pdf Object). Available from World Wide Web: <http://empresas.telefonica.es/documentacion/catalogo_servicios/Manual_Usuario_Servidor_Radius.pdf>.

(6) Protocolo RADIUS. Available from World Wide Web: <http://technet.microsoft.com/es-es/library/cc781821.aspx>.

(7) Protocolos TCP/IP. Available from World Wide Web: <http://www.saulo.net/pub/tcpip/a.htm>.

(8) RADIUS « Alipi Style. Available from World Wide Web: <http://yobtrams.wordpress.com/tag/radius/>.

(9) RFC 2865 Traducción al español. Available from World Wide Web: <http://www.normes-internet.com/normes.php?rfc=rfc2865&lang=es>.

(10) RFC 2866 Traducción al español. Available from World Wide Web: <http://www.normes-internet.com/normes.php?rfc=rfc2866&lang=es>.

(11) TACACS | Spanish | Dictionary & Translation by Babylon. Available from World Wide Web: <http://www.babylon.com/definition/TACACS/Spanish>.

(12) Telem@tica_AnoIV_No3.pdf (application/pdf Object). Available from World Wide Web: <http://telematica.cicese.mx/revistatel/archivos/Telem@tica_AnoIV_No3.pdf>.

(13) RADIUS.pdf (application/pdf Object). Available from World Wide Web: <http://santiagozky.files.wordpress.com/2007/11/radius.pdf>.

(14) capitulo4.pdf. Available from World Wide Web: <http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/morales_h_sp/capitulo4.pdf>.

(15) Guía de planeamiento: diseño de una infraestructura de RADIUS para la seguridad de LAN inalámbricas. Available from World Wide Web: <http://technet.microsoft.com/es-es/library/dd458735.aspx>.

(16) 16-2_7.pdf (Objeto application/pdf). Available from World Wide Web: <http://www.umng.edu.co/www/resources/16-2_7.pdf>.

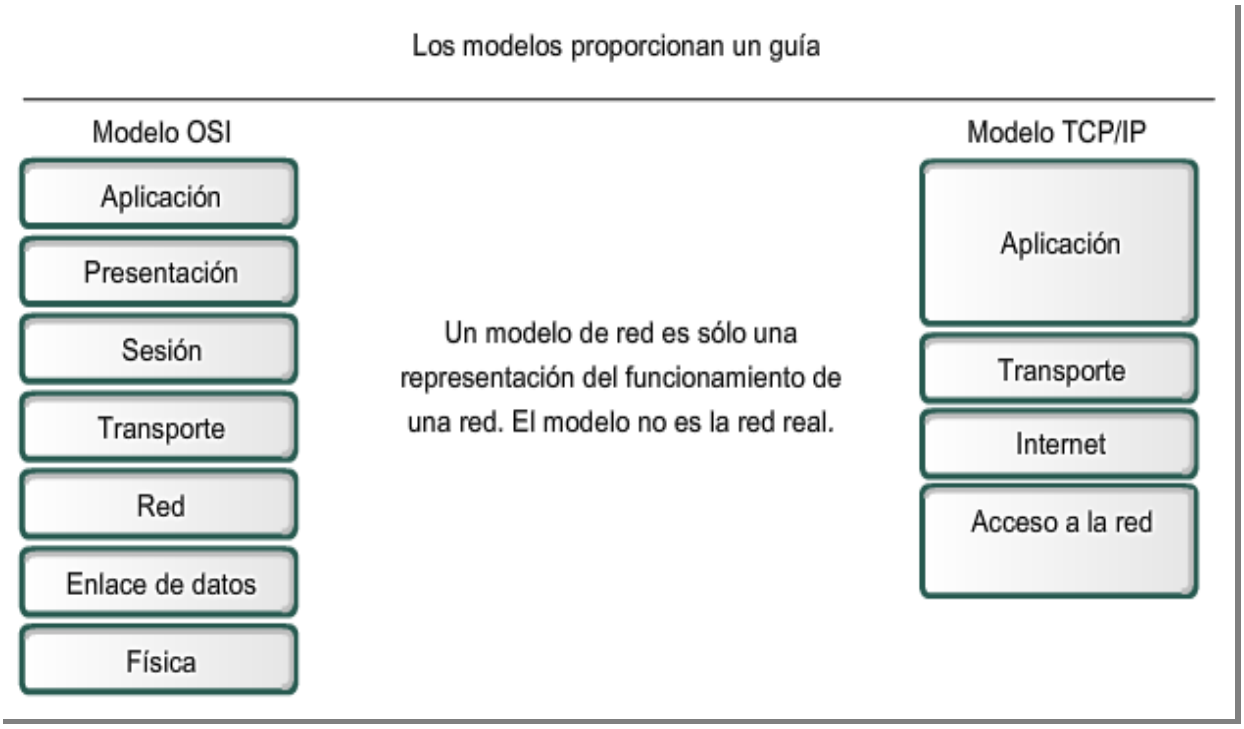
(17) capitulo5.pdf (Objeto application/pdf). Available from World Wide Web: <http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/matanzo_d_a/capitulo5.pdf>.

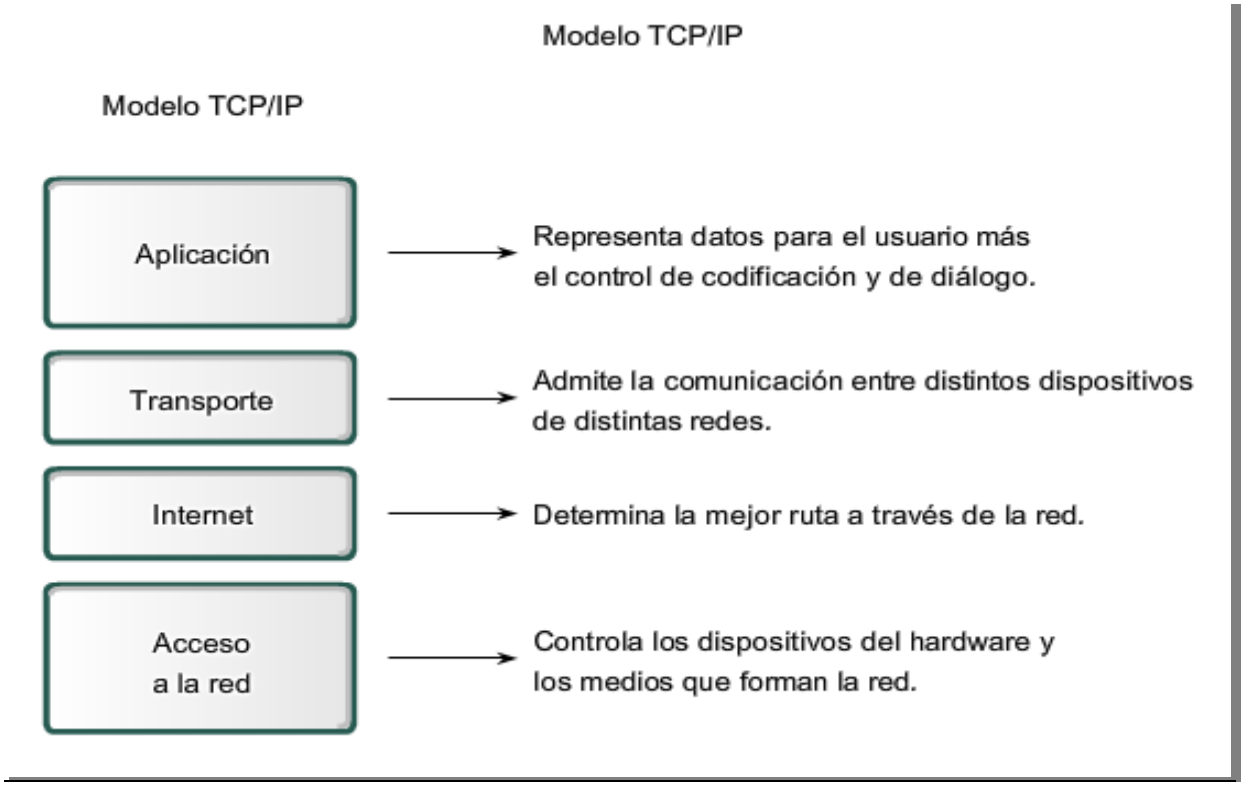
(18) Redes: Definición de Protocolos de Red. Available from World Wide Web: <http://fmc.axarnet.es/redes/tema_06.htm>.

(19) SEINIT Portal. Available from World Wide Web: <http://www.isoc.org/seinit/portal/index.php?option=com_content&task=view&id=49&Itemid=26>.

ANEXOS

Anexo 1



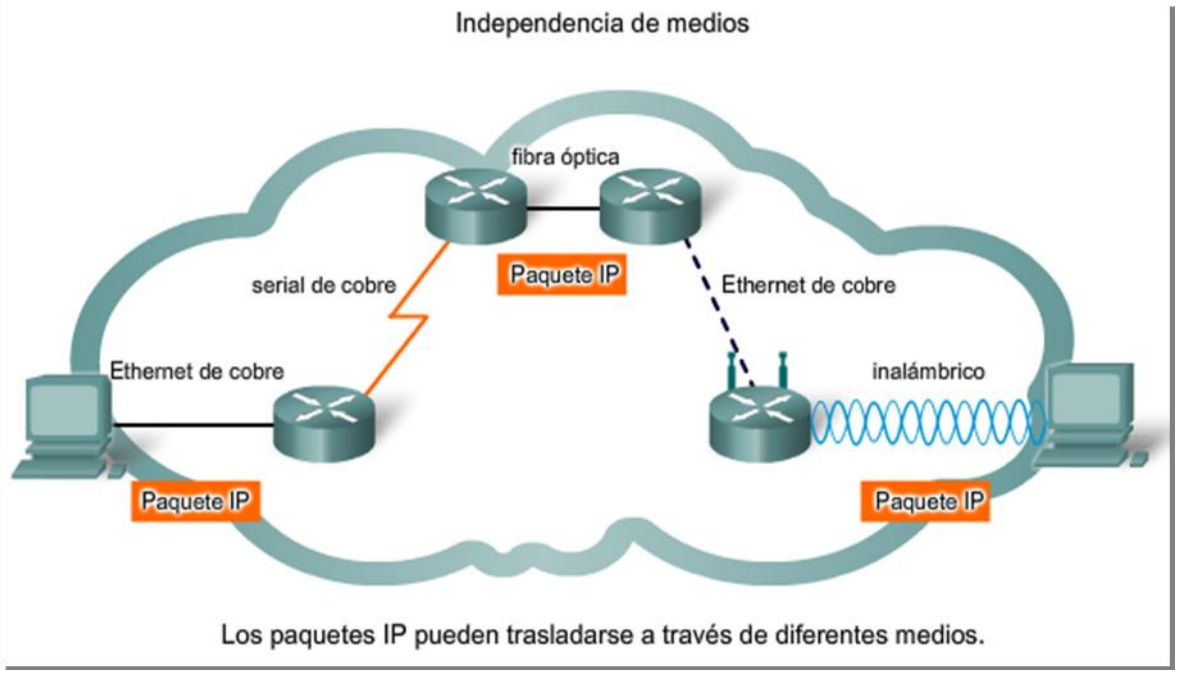

Modelo TCP/IP
Modelo TCP/IP
Aplicación
Representa datos para el usuario más el control de codificación y de diálogo.
Transporte
Admite la comunicación entre distintos dispositivos de distintas redes.
Internet
Determina la mejor ruta a través de la red.
Acceso a la red
Controla los dispositivos del hardware y los medios que forman la red.

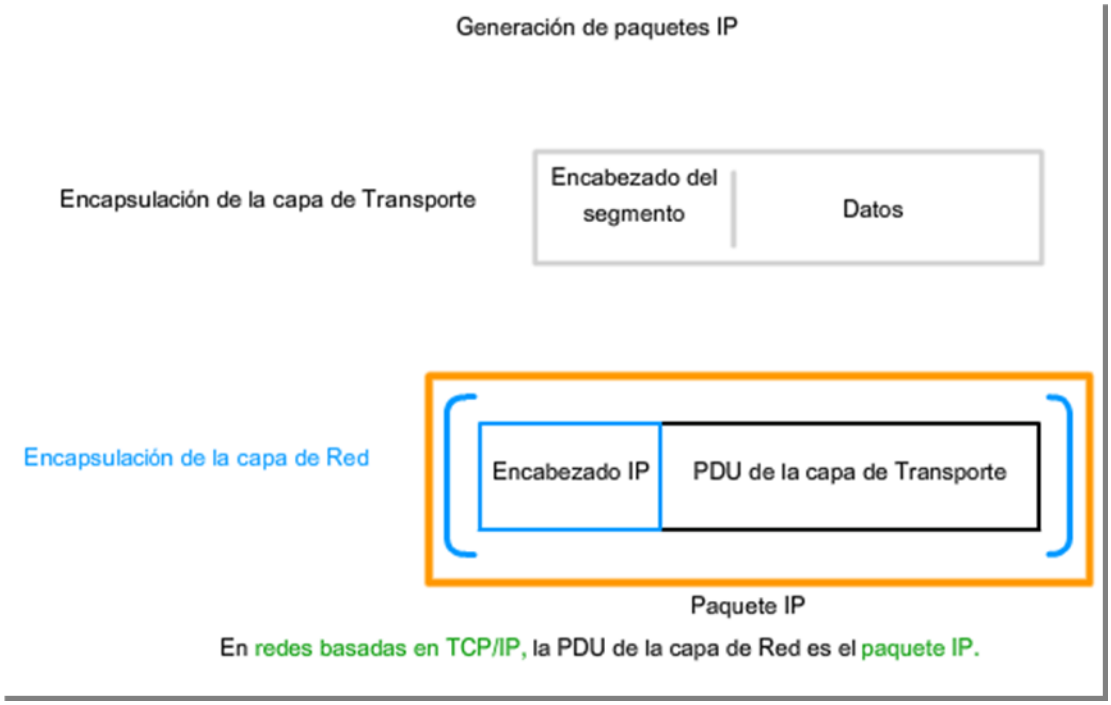
Anexo 2

IEEE 802.3						
7	1	6	6	2	46 a 1500	4
Preámbulo	Delimitador de inicio de trama	Dirección de destino	Dirección de origen	Longitud/Tipo	Encabezado y datos 802.2	Secuencia de verificación de trama

Ethernet					
8	6	6	2	46	4
Preámbulo	Dirección de destino	Dirección de origen	Tipo	Datos	Secuencia de verificación de trama

Anexo 3




Generación de paquetes IP
Encapsulación de la capa de Transporte
Encabezado del segmento
Datos
Encapsulación de la capa de Red
Encabezado IP
PDU de la capa de Transporte
Paquete IP
En redes basadas en TCP/IP, la PDU de la capa de Red es el paquete IP.

Anexo 4

Fig. Encabezado de protocolos de transporte

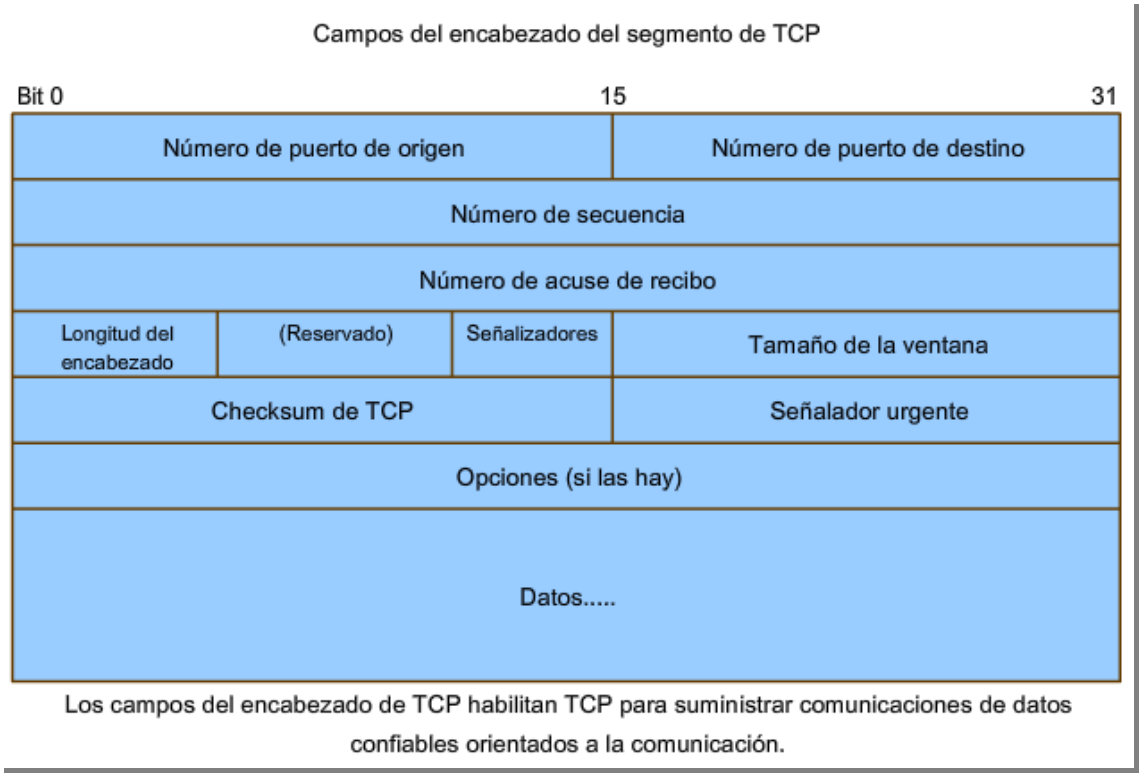


Fig. Campos de un segmento TCP

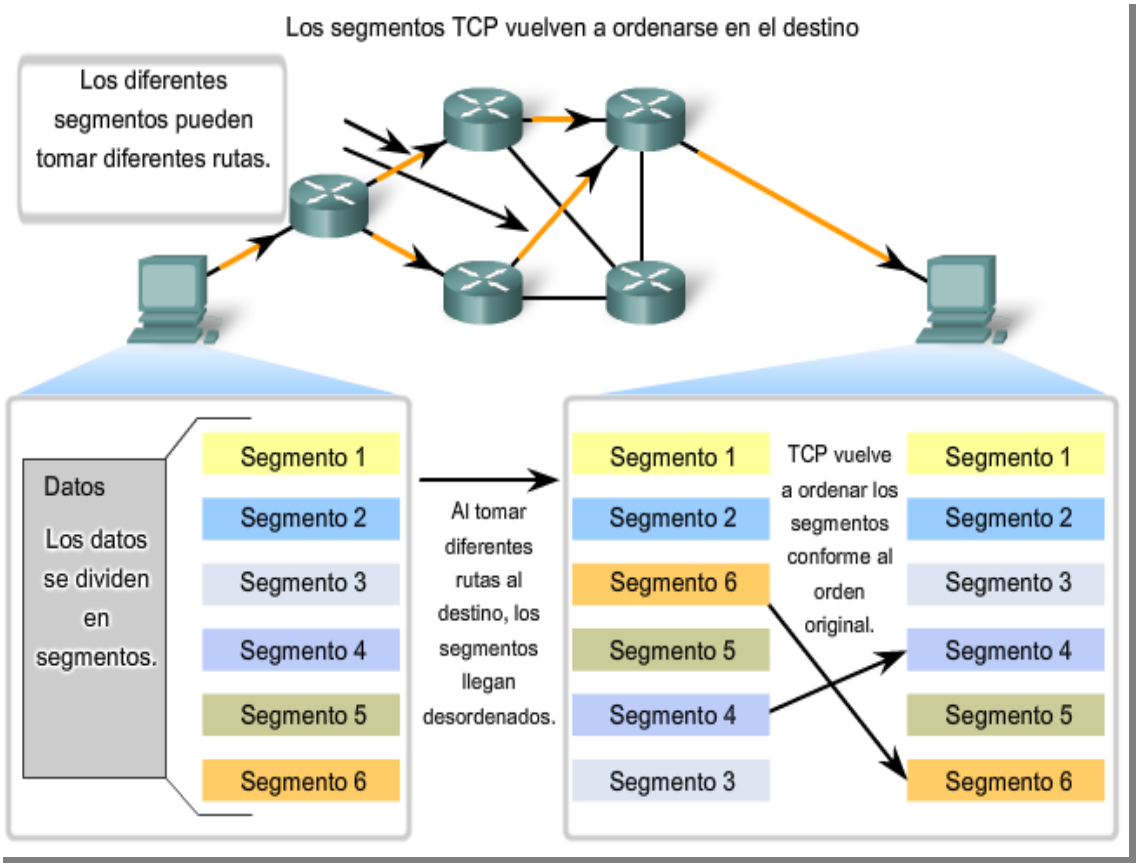
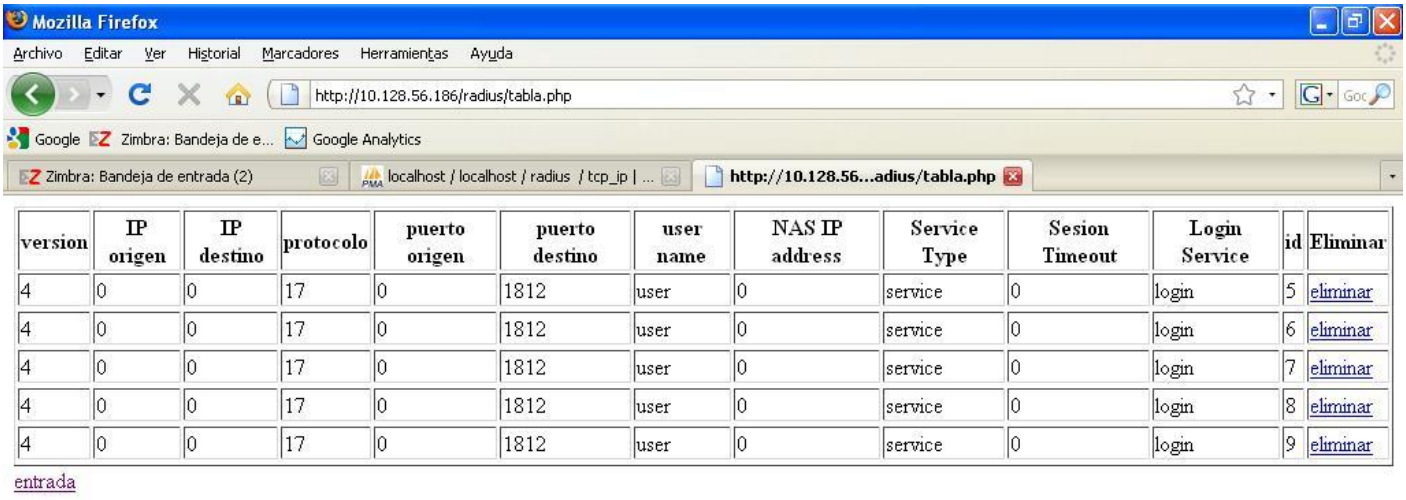


Fig. Transferencia de segmentos TCP



version	IP origen	IP destino	protocolo	puerto origen	puerto destino	user name	NAS IP address	Service Type	Sesion Timeout	Login Service	id	Eliminar
4	0	0	17	0	1812	user	0	service	0	login	5	eliminar
4	0	0	17	0	1812	user	0	service	0	login	6	eliminar
4	0	0	17	0	1812	user	0	service	0	login	7	eliminar
4	0	0	17	0	1812	user	0	service	0	login	8	eliminar
4	0	0	17	0	1812	user	0	service	0	login	9	eliminar

entrada

Tabla TCP_IP de la base de datos radius.

GLOSARIO

Acceso Remoto: Utilidad para que un usuario acceda desde su propio ordenador a otro que esté ubicado remotamente y pueda operar sobre él.

CHAP - Challenge Handshake Authentication Protocol: Protocolo de autenticación para servidores PPP donde la contraseña no sólo se exige al empezar la conexión sino también durante la conexión, mucho más seguro que el PAP. Una vez efectuado el enlace, el servidor envía un mensaje de desafío al solicitante de la conexión, el cual responde con un valor hash que será comparado por el servidor con sus cálculos del valor hash esperado. Si el valor coincide, la autenticación prospera, de lo contrario, finaliza. En cualquier momento el servidor puede solicitar un mensaje de desafío. Debido a que los identificadores cambian frecuentemente y por que la autenticación puede ser solicitada en cualquier momento.

Control de Acceso: Se utiliza para restringir el acceso a determinadas áreas de la PC, de la red, mainframes, Internet, ftp, web, etc... El permiso o la denegación de acceso puede realizarse en función de la dirección IP, el nombre de dominio, nombre de usuario y password, certificados del cliente, protocolos de seguridad de redes, etc...

EAP (Extensible Authentication Protocol) - Protocolo de Autenticación Extensible: Extensión del Protocolo punto a punto (PPP). Proporciona un mecanismo estándar para aceptar métodos de autenticación adicionales junto con PPP. Al utilizar EAP, se pueden agregar varios esquemas de autenticación, entre los que se incluyen tarjetas de identificación, contraseñas de un sólo uso, autenticación por clave pública mediante tarjetas inteligentes, certificados y otros. Junto con los métodos de autenticación EAP de alto nivel, es un componente tecnológico crítico para las conexiones seguras a través de una red privada virtual (VPN), puesto que ofrece mayor seguridad frente a ataques físicos o de diccionario y de investigación de contraseñas, que otros métodos de autenticación, como CHAP.

IEEE: Instituto de Ingenieros Eléctricos y Electrónicos.

LAN (Local Area Network)- Red de Área Local: Red informática que cubre un área relativamente pequeña (generalmente un edificio o grupo de edificios). La mayoría conecta puestos de trabajo (workstations) y PC's. Cada nodo (ordenador individual) tiene su propio CPU y programas pero también puede acceder a los datos y dispositivos de otros nodos así como comunicarse con éstos (e-mail)... Sus características son: Topología en anillo o lineal, Arquitectura punto a punto o cliente/servidor, Conexión por fibra óptica, cable coaxial o entrelazado, ondas de radio.

LDAP (Lightweight Directory Access Protocol) - Protocolo de Acceso Ligero a Directorio: Protocolo para el acceso a directorios jerárquicos de información. Basado en el estándar X.500, pero significativamente más simple por lo que también se le denomina x.500-lite, se diferencia de éste porque soporta TCP/IP, necesario para cualquier tipo de acceso a Internet. Aunque no está ampliamente extendido, debería poderse implementar en la práctica para la mayoría de las aplicaciones que se ejecutan virtualmente sobre plataformas informáticas para obtener información de directorios tales como direcciones de correo y llaves públicas. Ya que es un protocolo abierto, no afecta el tipo de servidor en el que se aloje el directorio.

LEAP - Lightweight Extensible Authentication Protocol: Protocolo del tipo EAP patentado por Cisco basado en nombre de usuario y contraseña que se envía sin protección. Esta metodología descuida la protección de las credenciales durante la fase de autenticación del usuario con el servidor.

PEAP - Protected Extensible Authentication Protocol: Protocolo del tipo EAP desarrollado conjuntamente por Microsoft, RSA Security y Cisco para la transmisión datos autenticados, incluso claves, sobre redes inalámbricas 802.11. Autentica clientes de red wi-fi empleando sólo certificados del lado servidor creando una túnel SSL/TLS encriptado entre el cliente y el servidor de autenticación. El túnel luego protege el resto de intercambios de autenticación de usuario.

PPP (Point-to-Point Protocol) - Protocolo punto a punto: Es un protocolo estandarizado en el documento RFC 1661. Por tanto, se trata de un protocolo asociado a la pila TCP/IP de uso en Internet.

Protocolo: Estándar establecido. En lo referente a conectividad de redes, el empleo de un protocolo se realiza para direccionar y asegurar la entrega de paquetes a través de la red.

Roaming: En redes inalámbricas se refiere a la capacidad de moverse desde un área cubierta por un Punto de Acceso a otra sin interrumpir el servicio o pérdida de conectividad.

Servidor de Autenticación: Servidores que gestionan las bases de datos de todos los usuarios de una red y sus respectivas contraseñas para acceder a determinados recursos. Permiten o deniegan el acceso en función de los derechos atribuidos.

SLIP (Serial Line Internet Protocol): Es un estándar de transmisión de datagramas IP para líneas serie, pero que ha quedado bastante obsoleto. Fue diseñado para trabajar a través de puerto serie y conexión de módem. Su especificación se encuentra en el documento RFC 1055.

Sniffers: Programa y/o dispositivo que monitoriza la circulación de datos a través de una red. Los sniffers pueden emplearse tanto con funciones legítimas de gestión de red como para el robo de información. Los sniffers no autorizados pueden ser extremadamente peligrosos para la seguridad de una red ya que virtualmente es casi imposible detectarlos y pueden ser emplazados en cualquier lugar, convirtiéndolos en un arma indispensable de muchos piratas informáticos. Algunas herramientas sniffers conocidas son: WepCrack, Airsnort o NetStumbler, entre otras.

TLS - Transport Layer Security: Protocolo del tipo EAP que garantiza la privacidad y la seguridad de datos entre aplicaciones cliente/servidor que se comunican vía Internet. Trabaja en dos niveles: El protocolo de registro TLS - situado en el nivel superior de un protocolo de transporte seguro como TCP asegura que la conexión es privada empleado encriptación simétrica de datos y asegura que la conexión es fiable.

WAN - Red de Área Amplia: Tipo de red compuesta por dos o más redes de área local (LANs) conectas entre si vía teléfono (generalmente digital).