

Examen 1 Tecnológico de Costa Rica
Escuela de Ingeniería en Computación
Redes (IC 7602)
Primer Semestre 2023
Esteban Ignacio Durán Vargas - 2020388144

Pregunta 1 (30 pts)

En los últimos tiempos el boxeo femenino se ha puesto de moda en Costa Rica, con una campeona que ha ganado 5 títulos mundiales y una cantidad de jóvenes promesas que destacan por su calidad, disciplina y gallardía, este auge del boxeo femenino no ha pasado desapercibido tanto a nivel nacional como internacional, luego de múltiples reuniones entre cámaras empresariales, deportistas, gobiernos locales, cámaras y asociaciones de turismo se ha decidido promover a Costa Rica como un destino boxístico, en especial la provincia de Guanacaste y de Puntarenas, el plan en términos generales es desarrollar un complejo turístico en la Isla de Chira, el mismo será conocido como Boxing Island, en este complejo se tendrán varios hoteles de lujo de capital privado y se contará con el Champions Dome, una arena de boxeo que tendrá capacidad para 20 mil personas superando al emblemático MGM Grand Garden Arena. Una vez que Boxing Island se encuentre en operación, será necesario realizar transmisiones en tiempo real de eventos deportivos en modalidad PPV, una alianza entre ISPs Internacionales, RACSA y Grupo ICE garantizaran conexiones a Internet de banda ancha, redundantes y con alta disponibilidad (un SLA de 99 con 9 nueves luego del punto decimal), para mejorar la experiencia se ha decidido hacer transmisiones desde dos ubicaciones, la primera es desde Boxing Island y la segunda desde AWS Cloud (esta última recibe la transmisión desde Boxing Island por lo cual expone un delay más alto), el siguiente diagrama muestra la solución: Debido al delay más alto que exhibirá AWS Cloud, se espera que los usuarios prefieren consumir la transmisión desde Boxing Island, es importante mencionar que el sistema de transmisión en tiempo real estará implementado sobre TCP utilizando el protocolo de capa de aplicación HTTPS (Hypertext Transfer Protocol Secure). Gracias a este proyecto se han abierto muchas oportunidades laborales en redes, con el beneficio que se podrá vivir en una isla paradisíaca, por esta razón usted ha decidido aplicar por un puesto de Network Engineer y en la entrevista técnica le hicieron las siguientes preguntas:

- Con el fin de mejorar el rendimiento de todo el sistema ocupamos decidir el tipo de Load Balancer que usaremos para balancear la carga que va hacia los streaming servers, actualmente estamos discutiendo utilizar un Load Balancer de capa 7 (aplicación) o un Load Balancer de capa 3 y 4 (red) , explique detalladamente ¿Qué tipo de Load Balancer recomendaría? (15 pts)

Analizando la situación brindada y considerando que utiliza protocolos de TCP y HTTPS; y que además se considera la velocidad como una prioridad para transmisiones en tiempo real, se decide que la mejor opción sería el Load Balancer de capa de red. La primera razón es que tiene mejor rendimiento que el de aplicación, esto debido a que el load balancer de capa 7 hace el balanceo basado en el contenido de los paquetes de la aplicación lo cuál requiere una inspección mayor que los de capa de red. Estos nada más son dirigidos con el contenido de sus headers y al implementar protocolos como TCP se puede balancear la carga al servidor correspondiente dependiendo la transmisión

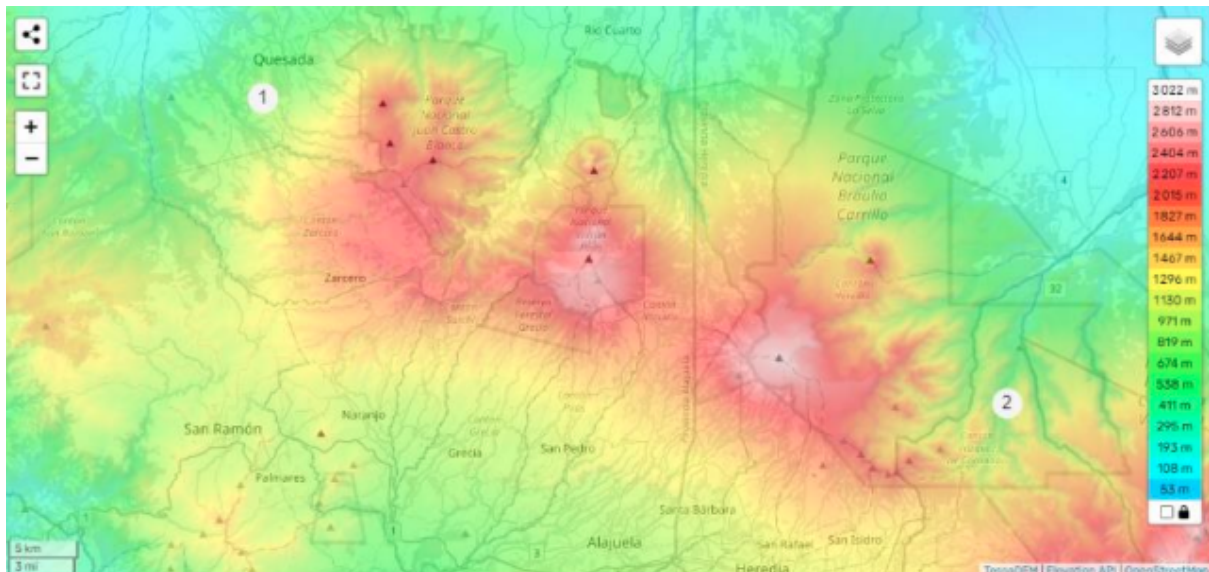
que se requiera ver, no específicamente el contenido que se esté enviando en el HTTPs, que no cambia en su mayoría por ser principalmente transmisión en flujo continuo del mismo tipo dentro de la aplicación. Es por esto que se le da prioridad a TCP, debido a que existen protocolos como RTSP dedicados a este trabajo y no teniendo que ver tanto con lo que es el contenido en sí (Tanenbaum, secciones 7.4.3).

- Existe la posibilidad de que se cambie el sistema de transmisión, en términos generales se cambiaría el protocolo de transporte por UDP y el protocolo de aplicación por uno privativo ampliamente utilizado por las aplicaciones de streaming más populares, ¿Considera que la escogencia de tipo de Load Balancer de la pregunta anterior sigue siendo adecuada? Justifique cuidadosamente su respuesta. (15 pts)

Esta escogencia no terminaría afectando la elección del Load Balancer de capa de red. El principal motivo es que UDP sigue siendo compatible con el Load Balancer de capa de red, al igual que lo era TCP, debido a que es un protocolo que estas capas manejan. Afectaría el servicio en el sentido de que se perderían más paquetes que con TCP, pero la distribución de carga sigue partiendo del mismo principio. Entonces, el cambio de un protocolo a otro no afecta ni disminuye el hecho de que revisar estos para balancear la carga de los servidores sigue teniendo mejor rendimiento que revisar el contenido de los paquetes de este nuevo protocolo privativo. Del mismo modo, se le da prioridad a UDP sobre el protocolo de aplicación porque importa más para protocolos de streaming como RTP en este caso (Tanenbaum, secciones 7.4.3). Hablando de este nuevo protocolo de aplicación de otras aplicaciones de streaming, este no afectaría el hecho de cómo se distribuye la carga a nivel de red, debido a que son las mismas transmisiones y el contenido no afectará a dónde se dirige el Load Balancer la carga.

Pregunta 2 (30 pts)

La siguiente imagen muestra una porción de un mapa de curvas de nivel del territorio costarricense, el mismo fue obtenido en el web site Topographic Map.



La empresa Direct Connect requiere una propuesta de capa física que permita establecer un enlace directo bidireccional entre los puntos 1 y 2, es importante mencionar que debido a las características del terreno y meteorológicas el uso de enlaces satelitales no es una opción, Direct Connect tiene un budget limitado por lo cual es necesario reducir al mínimo la inversión para establecer este enlace. En calidad de Network Engineer presente una propuesta de tecnologías de capa física que utilizaría para establecer este enlace, justifique su respuesta cuidadosamente, tome en cuenta el terreno, acceso a fuentes de energía e infraestructura.

En primer lugar, para empezar esta propuesta, se quiere empezar con que el medio a utilizar debe ser no guiado, es decir inalámbrico. Esto debido a que es muy probable que debido al terreno tan elevado, sería costoso y difícil instalarlo en este. Si se ignora el terreno y se decide rodear estas elevaciones, la distancia a recorrer debilitaría la señal y requeriría muchos repetidores para llevar la señal, lo cuál va en contra del principio de reducir costos. Debido a esto, y tomando el libro de Tanenbaum, sección 2.3.2 y 2.3.3 páginas 103-105, se concluye que se necesita un medio inalámbrico el cual sea de bajo costo y que cubre grandes distancias para alcanzar los puntos realizando una línea lo más recta para evitar desviarse y requerir instalaciones extra. La propuesta principalmente se centra en el uso de radiotransmisión de baja frecuencia o Microondas, y a continuación se explica el razonamiento de cada una según las condiciones disponibles. La propuesta principal es Microondas, debido a que pueden cubrir grandes distancias y debido a que la construcción de torres puede ser sencilla y menos costosa que utilizar medios guiados. Aprovechando el terreno, se pueden utilizar las montañas para construir estas torres en lugares altos para que queden en línea vista unas de otras y ser lo suficientemente alta para evitar que reboten en los árboles. Uno de los únicos problemas de esta propuesta es que el espectro no está tan disponible, entonces si (en el escenario) se puede conseguir una transmisión de una frecuencia suficiente para la labor que se quiere usar, se recomienda esa; de otra manera se debe considerar la alternativa de utilizar radiotransmisión. Esta propuesta, en bajas frecuencias que le permita atravesar obstáculos en estas montañas y de fácil generación, permite los viajes a distancia necesarios para las siguientes torres que

no ocupan estar en línea vista como microondas. Además, las frecuencias bajas permiten penetrar obstáculos, pero también se pueden usar altas (que ya no serían tan de bajo costo como microondas) y construir las torres necesarias en el terreno intermedio para recibirla. Por ser un terreno tan alejado podemos también evitar muchas interferencias por otros factores como serían motores u otros usuarios. A pesar de esto ambas propuestas son afectadas por la lluvia, factor que se puede inferir del enunciado, entonces en ambas se deben tomar en cuenta medidas para evitar la absorción de las señales. De esta manera, si se encuentra una frecuencia y el presupuesto lo requiere, se escogería microondas, pero la opción más sencilla sin tomar en cuenta muchos factores sería radiotransmisión.

Pregunta 3 (20 pts)

La privacidad en redes es un tema sumamente importante, en el modelo OSI existen varias capas en las cuales podemos implementar encriptación de datos siendo las más comunes: capa de aplicación y capa de sesión, como parte de una prueba de concepto se quiere explorar la posibilidad de implementar encriptación punto a punto en capa de Data Link (enlace de datos) mediante la implementación de tarjetas de red especializadas que generen y administren llaves PGP de longitud 2048 bits, antes de iniciar una comunicación entre dos estaciones se deben intercambiar las llaves públicas entre las dos computadoras que desean comunicarse, estas llaves se almacenarán en un cache como triplets de la forma {MAC, public key, ttl}, las tramas antes de salir son encriptadas y al ingresar son desencriptadas, la encriptación/desencriptación únicamente existe entre la computadora originaria de la comunicación y la receptora de la misma, todos los dispositivos intermedios por los cuales pasan las tramas no se dan cuenta del uso de este mecanismo de seguridad por ende no requieren tener estas tarjetas especializadas.

Para decidir la viabilidad de esta prueba de concepto, se requiere dar respuesta a las siguientes interrogantes:

- Desde un punto de vista de diseño del modelo OSI, ¿Es viable la implementación de este tipo de encriptación? (10 pts)

Desde el punto de vista del modelo OSI ésta implementación no es viable, debido a que la capa específica donde se quiere implementar este modelo, DataLink, no cumpliría con la estandarización OSI. Esto basándose en el concepto visto en clase de que las capas del estándar no pueden acceder a las capas inferiores. Esto considerando que la capa datalink no maneja características de enrutamiento para intercambiar llaves con el equipo de destino; ya que eso sucede en la capa 3 (Según Tanenbaum en sus capítulos 1, sección 1.4.1, 3 y 5). Cambios como estos, incluso el simple concepto de realizar modificaciones en datalink, va en contra del modelo que propone OSI entonces los hace inviables. Esto no quiere decir que la prueba de concepto sería completamente inviable, debido a que el modelo OSI es un sistema de referencia, y existen protocolos que no lo siguen al pie de la letra para realizar sus labores. Por eso, al usar la tarjeta de red necesaria para implementar el concepto, se estaría utilizando un nuevo estándar para las estaciones por comunicar, por ende la comunicación y desencriptación de los datos seguiría siendo posible con ambas teniendo el mismo estándar, solo que tal vez no funcionaría de la misma manera con estaciones externas que no la tuvieran y lleguen a intercambiar paquetes con estas.

- Describa en detalle ¿cómo se podrían obtener las llaves PGP antes de iniciar la comunicación? Para dar respuesta a esta pregunta es sumamente importante tener claro que nos encontramos en la capa de enlace de datos y no en la capa de red. (10 pts)

Siguiendo la respuesta anterior, donde establecimos que sería necesario enrutar los paquetes para hacer la transferencia correcta y segura utilizando PGP en datalink, diría que habría que modificar la estructura de las capas para poder enrutar los paquetes en un inicio antes de iniciar la conexión. Dentro de mi propuesta aplicaría algo similar a lo que realiza IPsec, según Tanenbaum en su sección 8.6.1, que sucede en capa 3 (“orientado a conexión”) para crear una “conexión” con el otro equipo que tiene la tarjeta de red en desarrollo y crear un túnel entre ambos (esto en modalidad túnel donde se encapsula en un nuevo paquete IP para redirigirlo a donde debe, según la página 773). De esta manera se crearía una conexión entre estos 2 puntos que puede ser validada mediante las llaves y la MAC que tienen. Para el intercambio en sí, esta conexión se haría con la estación de destino para realizar un intercambio de llaves que se pasarían por medio de campos en los encabezados hasta la capa de datalink y que finalmente podría establecer la comunicación segura y empezar.

Pregunta 4 (20 pts)

El proyecto Gemini, se encuentra desarrollando un protocolo que permita tomar tramas de Ethernet clásica y convertirlas en tramas Gemini, las tramas Gemini son idénticas a las tramas Ethernet clásica, pero los datos que transporta no pueden exceder los 140 octetos, son de tamaño fijo (si se tienen que enviar menos de 140 bytes se tiene que rellenar con 0s) y además se le agrega un campo con un número consecutivo de 4 bits que llamaremos el número Gemini y otro de 7 bits que llamaremos el tamaño Gemini, estos se ubicarán entre los campos Tipo/Longitud y Datos.

El número Gemini lleva un consecutivo de trama Gemini, esto permitirá hacer una entrega en orden de tramas Gemini.

El tamaño Gemini indicará cuantos caracteres tiene el payload (0 – 140 bytes).

Su tarea como diseñadora o diseñador de protocolos, es la elaboración de un pseudo código de un protocolo de capa de enlace de datos con parada a nivel de trama Ethernet clásica, debe contar con confirmación de recepción y verificación de datos a nivel de trama Gemini. El protocolo recibe una trama Ethernet clásica, la divide en fragmentos de máximo 140 bytes (cantidad de tramas Gemini), el protocolo entonces envía todas las tramas Gemini sin esperar y espera por la confirmación de recepción de cada una de ellas para confirmar que la trama Ethernet clásica fue enviada (parada a nivel de trama Ethernet clásica). El protocolo debe contar con un mecanismo de reenvío en caso de no recibir confirmación o en caso de recibir una verificación de datos negativa.

El protocolo debe garantizar que la trama enviada es exactamente igual a la trama recibida. Se debe implementar los siguientes métodos:

- void geminiSender(void) (10 pts)

- void geminiReceiver(void) (10 pts)

Elaboren diagramas de la forma en la cual se ven las tramas Gemini y de la forma en la cual funciona su pseudo código.

Pueden asumir que existen las siguientes funciones:

- void from_ethernet(buffer): la cual obtiene una trama Ethernet clásica.
- bool verify_data(buffer): recibe una trama Gemini y verifica que los datos estén correctos.
- void from_physical_layer(buffer): obtiene una trama Gemini de la capa física.
- void to_physical_layer(buffer): que envía una trama Gemini a la capa física.
- void to_ethernet(buffer): entrega una trama Ethernet clásica.
- void wait_for_event(buffer): espera el arribo de una trama o expiración de un timer.
- void start_timer(): inicia un timer

Trama Gemini

8	1	6	6	2	4	7	140	4
Preámbulo	Inicio de trama	DST	SRC	Tipo/ Longitud	# Gemini	Tamaño Gemini	Payload	Checksum

Pseudocódigo

A continuación se presenta el pseudocódigo desarrollado, con comentarios explicando cada fase. Esta se basó en el protocolo símplex de parada y espera del Tanenbaum (sección 3.3.2, páginas 206-207). Adicional a las funciones supuestas, se añaden unas cuantas:

- void get_gemini_payload(buffer, offset): toma los primeros 140 bytes de la trama Ethernet, con el offset se permite seguir desde los últimos 140 .
- int gemini_payload_len(gemini_payload): calcula el largo en bytes del payload sin el relleno.
- void fill_payload_zeros(gemini_payload): Crea el relleno de ceros si se tomaron menos de 140 bytes.
- void appendtobuffer(agree): concatena a un buffer las tramas geminin para formar la ethernet una vez se transmita en su totalidad.

```

void geminiSender(void) {
    gemini_nr next_frame_to_send; /* número de secuencia de la siguiente trama de salida */
    geminiframe f; /* variable de trabajo */
    packet buffer; /* búfer para un paquete de salida */
    event_type event;
    int offset = 0;
    from_ethernet
    next_frame_to_send = 0; /* inicializa números de secuencia de salida */
    while(true){
        from_ethernet(&buffer); /* obtiene el paquete Ethernet Clásico*/
        gemini_payload = get_gemini_payload(&buffer, &offset) /*toma 140 bytes a partir de un offset, el inicial es 0*/
        gemini_payload_bytes = gemini_payload_len(gemini_payload) /*calcula cuántos bytes tomó*/
        if(gemini_payload_bytes<140){ /*si se tomaron menos de 140 se llena el payload con ceros*/
            fill_payload_zeros(&gemini_payload)
        }
        while (true){
            f.info = gemini_payload; /* construye una trama para transmisión */
            f.gemseq = next_frame_to_send; /* Inserta el número gemini */
            f.gemsize = gemini_payload_bytes; /*Inserta el tamaño Gemini en la trama*/
            to_physical_layer(&f); /* la envía a su destino */
            start_timer(); /* Empieza el timer en caso de timeout */
            wait_for_event(&event); /* Espera un evento: frame_arrival o timeout */
            /*Si es timeout reenvía*/
            if (event == frame_arrival){
                from_physical_layer(&f); /* obtiene la confirmación de recepción */
                if (f.ack == next_frame_to_send){
                    stop_timer(f.ack); /* desactiva el temporizador */
                    if(gemini_payload_bytes<140){ /*Revisa si ya no quedan más de 140 bytes (terminó)*/
                        break; /*Termina y trae el próximo paquete Ethernet*/
                    }else{
                        gemini_payload = get_gemini_payload(&buffer, &offset)
                        gemini_payload_bytes = gemini_payload_len(gemini_payload)
                        if(gemini_payload_bytes<140){ /*si se tomaron menos de 140 se llena el payload con ceros*/
                            fill_payload_zeros(&gemini_payload)
                        }
                    }
                }
                inc(next_frame_to_send); /* invierte next_frame_to_send */
            }
        }
    }
}

```

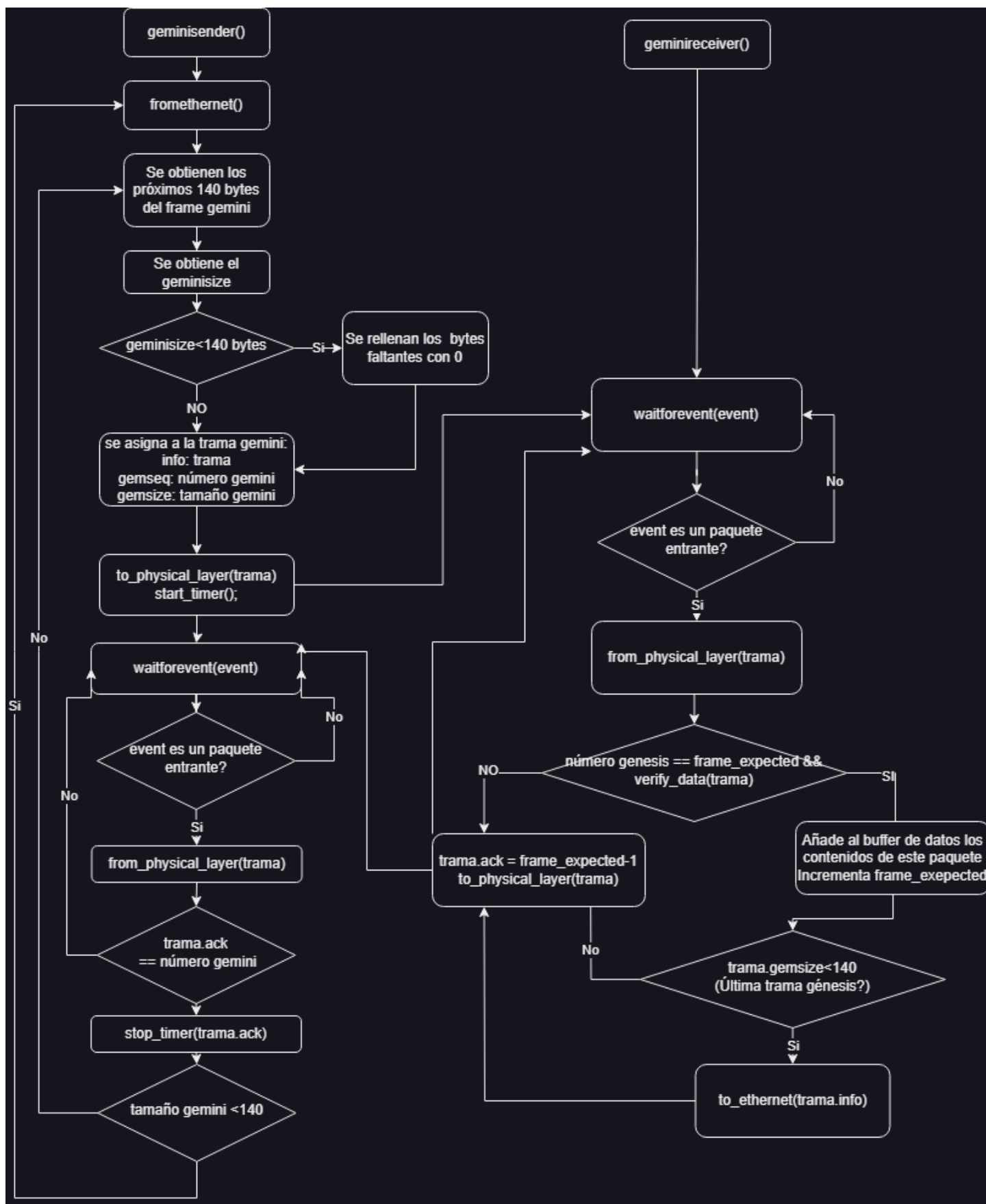
```

void geminiReceiver(void)
{
    gemini_nr frame_expected;
    geminiframe r, s;
    event_type event;
    packet buffer;
    frame_expected = 0;
    while (true){
        wait_for_event(&event); /* espera a que llegue un paquete */
        if (event == frame_arrival){ /* ha llegado una trama válida. */
            from_physical_layer(&r); /* obtiene la trama recién llegada */

            if (r.gemseq == frame_expected && verify_data(&r)){ /* esto es lo que hemos estado esperando. */
                append_to_buffer(&r.info, &buffer) /* Se añade la información a un buffer para ir armando la trama ethernet */
                inc(frame_expected); /* para la próxima se espera el otro número génesis */
                if (r.gemsize < 140){ /* Busca la secuencia de finalización y. */
                    to_ethernet(&buffer); /* pasa el paquete reconstruido a ethernet */
                }
            }
        }
        s.ack = frame_expected-1; /* indica la trama cuya recepción se está confirmando */
        to_physical_layer(&s); /* envía confirmación de recepción */
    }
}

```

Diagrama de pseudocódigo



Referencias

Tanenbaum, Andrew S., 1944- Computer networks / Andrew S. Tanenbaum, David J. Wetherall. -- 4th ed.