Universidad de Chile
Facultad de Ciencias Físicas y Matemáticas
Departamento de la Universidad

Examen

Profesor: José M. Piquer
Auxiliar: David Miranda
Estudiante: Samuel Chavez F.

Fecha de realización: 12 de diciembre de 2023
Fecha de entrega: 14 de diciembre de 2023
Santiago de Chile

Predisposiciones al caso analizado

En el desarrollo del examen se nos establece una aplicación de monitorio de personas **sanas**, no obstante pese a la indicación explicita y tomando el ejercicio como un caso real, esto realmente no es muy relevante en nuestro objetivo de como queremos mejorar la comunicación entre el dispositivo y nuestros servidores, el caso va a lo siguiente. Un fallo de una aplicación de monitorio rutinario a una persona normal realmente no es critico, no obstante es implícito que esto no sera aceptable sea por que los consumidores lo usaran de una forma no especificada en pacientes que requieran mejor monitorio que este producto o por que el producto en si sea implícita o explícitamente publicado para estos casos clínicos.

0.1. Comunicacion estable

Se nos presenta el dilema ético que ante todo, lo que no puede fallar en nuestra lista de prioridades es la comunicación servidor cliente, cualquier fallo entremedio de estas se considerara critico y nuestra primera prioridad e incluso nos puede empujar a incluir elementos de redundancia para que la comunicación siempre exista.

0.2. Comunicacion agil

Así mismo, nos aseguramos que la comunicación ocurra, necesariamente rápido, cumpliendo la función de alerta temprana que requiere, por lo que nuestra segunda prioridad sera la agilidad en el traspaso de información de monitorio.

0.3. Seguridad en la comunicacion

Privacidad y atacantes externos, en post de lo anterior descrito como vamos a diseñar la comunicación es critica la confidencialidad de pacientes, lo que lleva mas a el manejo de datos como tal que al transporte, pero si es remarcable la posibilidad de ataques externos a nuestro diseño, intentando que nuestra comunicación sea todo lo posible segura end to end, de tal forma minimizar todo vector de ataque entre el servidor y cliente que se pueda producir.

0.4. Resumen

Finalmente nuestras prioridades de predisposición a como se contestara el resto del ejercicio sigue lo antes descrito por las razones detalladas, finalmente, priorizamos la comunicación en si ante todo, luego la agilidad de esta, y en todo lo posible la seguridad de transporte.

1. Red Fisica

1.1. Análisis

1.1.1. Opción 1

Es una buena opción, mantener el dispositivo como esta, a esto le sumamos el hecho de que bluetooth es una tecnología que es eficiente en energía, lo que me imagino es importante para un dispositivo como el que se tiene en mente, portátil que monitor los signos vitales, esta realmente debería ser la primera opción del dispositivo, pero no necesariamente la única a tomar con el fin de asegurar la conexión el servidor. Es importante también eso si especificar que seguramente la razón por la que la aplicación del celular debe estar consumiendo tanta energía, es por que hasta el momento se esta utilizando como proxy que además tiene una configuración UTP todo el tiempo intentando mantenerla o peor, iniciarla cada vez que la persona se vaya moviendo, cuando realmente deberíamos tenerlo en una configuración de requests a este, el que envié paquetes UDP manufacturados por el dispositivo con el protocolo QUIC que veremos mas adelante.

1.1.2. Opción 2

Realmente inviable por si solo, obligar a que se use WiFi en el dispositivo le quitaría toda la movilidad y practicalidad que entrega el dispositivo, luego una persona que decida moverse, tiene que utilizar Hotspot del celular para entregar conexión con el servidor, realmente esta opción no me gusta para nada, esto también agregaría mas complejidad física al dispositivo para entregarle la opcion de WiFi.

1.1.3. Opción 3

Esta opción la verdad me tienta, si bien agrega complejidad al dispositivo, tanto física como lógica, nos entrega la opción de que sea util por si mismo. pero no me parece que necesariamente esta opción deba ser exclusiva, como bien mencionamos esto hara inevitablemente que la complejidad física del dispositivo sea mayor pero con la seguridad que el dispositivo monitorizara hasta que se quede sin energia.

1.1.4. Opción 4

Honestamente, si hubiese una forma tal que no se tuviese que tener una antena parabólica de alguna forma en el dispositivo, no seria mala idea para siempre tener la conexión con este, pero antes descrito, es necesaria una antena parabólica y muchas veces starlink se despliega como una red WiFi en la que los dispositivos se conecten. Mala idea por si misma e inviable.

1.2. Decisión final

Realmente me inclino por la primera opción, no obstante no me parece que deban ser exclusivas la primera y cuarta, usando la primera como principal opción de conexión a internet, y cuando falle esta se intente activar la red 3G/4G en vista a nuestro primer objetivo con el dispositivo de asegurar en todo momento la conexion con el servidor para su correcto uso.

2. Redes IP

2.1. Análisis

2.1.1. Opción 1

Opcion probada y confiable, el internet funciona en base a IPv4, si bien cambiandose, es realmente muy dificil que veamos que el protocolo se deje de usar simplemente por temas de compatibilidad dentro de mucho tiempo, por lo que si es una opcion a considerar.

2.1.2. Opción 2

Esta opción es buena, así mismo como eficiente y nos olga con las direcciones a nosotros proveyendo el servicio, realmente es una excelente opción y que impulsaría a tomar como primera decisión, pero no puede estar sola, simplemente por que estamos en momento de transición y la red global completamente no esta enteramente lista para tomar todo el trafico por IPv6, no obstante esto no quita a que es una opción a la que usar que nos dará la libertad entera para direcciones, sobretodo en nosotros como servicio mas que el cliente dado que como veremos mas adelante, de donde nos contacte el cliente nos sera irrelevante con nuestro protocolo se transporte.

2.1.3. Opción 3

Bueno en acordanza a nuestra selección anterior, me gusta la idea de mantener redundancia en nuestras posibilidades para generar la conexión, y esto va a ser un requisito si también se añade la opción de conexión 3G/4G en el propio dispositivo, añade complejidad al mismo en tiempo computacional cuando se este usando de esta forma pero es un costo que en lo personal siento es importante refiriéndonos a nuestro primer objetivo de asegurar la conexión.

2.1.4. Opción 4

En definitiva, esta opción no es viable, no solo por que el uso stand alone del dispositivo de esta forma acortara el tiempo de vida, sino también por que nos quitaría la capa de tener redundancia en conexión. Inviable.

2.2. Decisión final

Bueno la opción acá me decanto por usar la primera, simplemente por que es la mas confiable de las tres, no obstante, en nuestro objetivo de asegurar conexión yo si añadirira las opciones 3 y 2 en el orden respectivo, la primera por la redundancia que entrega haciendo que el dispositivo sea robusto en uso, y la segunda para olgarnos dentro de la complejidad de un mundo ya limitado por IPv4.

3. Transporte

3.1. Análisis

3.1.1. Opción 1

No es mala opcion, si es que el dispositivo no fuese movil, en general es un protocolo probado que nos da asegura la conexion y envio de mensajes dentro de lo mejor posible, pero tenemos mejores opciones.

3.1.2. Opción 2

Esta opción no es viable de ninguna forma por el hecho de utilizar S-W, simplemente nos dará un bando de ancha terrible para el dispositivo y estariamos desperdiciando ancho de banda y agilidad en los mensajes dentro de la red, esta opcion es descartada completamente

3.1.3. Opción 3

Luego de leída la revisión de QUIC, este caso simplemente es ideal para su uso, cae justamente como un dispositivo dentro de la red de IOT, nos traerá increíbles ventajas como su eficiencia en handshake y establecimiento así como agilidad que nos puede proveer UDP, pero sin abusar de la red, ya que tiene incorporado su balanceador de conexión así como seguridad por TLS como portabilidad de conexión, por lo que ahora el dispositivo sera libre de ser portátil y cambiar la 5-tupla que identifica una conexión regular, saltándose el paso de identificación al tener el token de conexión. En realidad no hay mucho mas que argumentar mas que nos costara un poco mas computacionalmente al ser un protocolo en application layer pero todas las ventajas que contraen completamente desvanecen esto ultimo, mas aun con nuestro gran avance en microchips.

3.1.4. Opción 4

Realmente a este punto no es mala idea mas haya de la complejidad que traerá usar anycast, pero como QUIC ya usa UDP, y es además un mejor protocolo para IOT, se ve completamente eclipsada esta opción

3.1.5. Opción 5

Realmente la opción que nos dará una mejor flexibilidad para nuestro servicio, el poder utilizar anycast nos dará aun mas ventajas con QUIC como protocolo usando UDP. pero conllevara unos desafíos que hay que tomar que veremos con DNS.

3.2. Decisión final

Realmente la opción ganadora y que simplemente gana al ser una tecnología probada confiable y eficiente acá es QUIC, no hay mucho mas que ahondar, el añadir anycast también sera una buena forma de darnos un respiro en arquitectura de redes para nuestro servicio que nos podrá dejar escalarlo apropiadamente.

4. DNS

4.1. Análisis

4.1.1. Opción 1

Dentro del caso se nos aclara que DNS tiene problemas, realmente esto tiene soluciones y no podemos dar la respuesta de solo no usar DNS, nos daría una restricción muy grande pero las siguientes opciones nos ayudara a apalear estos problemas y mas aun, intentar sobre todo y en alineamiento a nuestro primer objetivo hasta en los últimos casos intentar establecer conexión.

4.1.2. Opción 2

Esta opción es buena, pero no en una implementación de hardcodear esto, sino como una variable que podamos guardar para saltarnos el DNS de forma regular, esto ya lo hacen los propios DNS, la tabla que nos traduce nuestras direcciones, en nuestro caso solo nos vamos a conectar a un servidor, o una red de servicios por lo que no añadirá mucha complejidad y tiempo computacional el simplemente resolver nuestra dirección de servicio, cachear y utilizarla regularmente hasta que expire el cache o no tenga respuesta. Esto no es la ultima solución sino mas bien como un layer dentro de opciones que podemos tomar y una medida que nos ayudara con nuestro objetivo de intentar hacer la comunicación lo mas agil posible.

4.1.3. Opción 3

Esto es una buena opción que deberíamos tener, pero usar un resolver propio también nos dará la restricción de mantenerlo por el tiempo, mas, esto podra ser muy util para nuestro caso donde la familia de dispositivos tendrán a su disposición su propio resolver que idealmente sea mucho mejor que uno cualquiera.

4.1.4. Opción 4

Usar el DNS de google, excelente opción, es una buena red de seguridad para nuestro protocolo, ya que realmente es uno estable, y no dependiente de los proveedores donde podría incluso producirse un envenenamiento de cache, logrando así también alcanzar un poco mas la realización de nuestro tercer objetivo de seguridad de la conexión.

4.2. Decisión final

Acá realmente me decanto por las ultimas tres opciones, la redundancia sera clave en poder entregar un dispositivo que sea confiable en concordancia con nuestro primer objetivo, en donde si falla, es por que realmente las condiciones externas están muy mal. Especifico que como bien dije, usar la IP directa es una opción buena, pero que la implementación seria igual a como lo hace un DNS corriente, cachear con expiración las direcciones y consultar nuevamente por la resolución del dominio. luego en la resolución de dominio consultar primero a nuestro resolver, y en el peor de los casos, sea por que el resolver no responde o se demora mucho en la respuesta, consultar al DNS de google, que realmente si llega a fallar las opciones son que la conexión del dispositivo es pobre antes

de que google falle y asi reportar un error debidamente, ya que si llega a suceder el caso de que google falla y nuestro resolver también, seguramente van haber problemas mas graves en el mundo a que nuestro dispositivo no pueda reportar correctamente.

5. Diseño servidor

5.1. Análisis

5.1.1. Opción 1

No es mala opción si no fuese por que por como se generaran las conexiones serán de ráfagas de información por QUIC, por lo que la creación de procesos pesados si bien nos darían agilidad de servicio, es negligente a lo que nos demoraremos creando procesos pesados cada vez que nos llegue información. No es necesariamente malo, solo que tenemos mejores opciones disponibles.

5.1.2. Opción 2

Esta opción me gusta, tener los servidores configurados con threads le darán la agilidad en memoria necesaria al servidor para poder mantener múltiples ráfagas distintas de clientes y procesarlos correctamente casi sin costo para el sistema operativo y así utilizar eficientemente los recursos.

5.1.3. Opción 3

No es mala opción, y es muy buena si queremos cumplir con una red grande de dispositivos a los que dar servicio, pero se nos presenta la mejor opción de utilizar la nube, donde nos podemos ahorrar la complejidad del bare metal y su mantención con la flexibilidad que nos entrega de escalado.

5.1.4. Opción 4

Acá como antes expuesto esta opción parece ser idónea para un escalado orgánico de nuestro caso, que nos dejara crecer según la demanda y configurado de forma correcta también así nos dejara distribuir la carga del servicio. una excelente opción en mi opinión.

5.2. Decisión final

Esta pregunta fue analizada mucho mas por el lado económico de las cosas, mas que en nuestros objetivos, ya que dado por como hemos entablado todo hasta el momento nos otorga la opción de pensar en el servicio en si mismo mas que en la confiabilidad del momento, es por eso que la opción 4 y 2 son las que elegiría, no solo al saber que es IOT y no estaremos usando grandes procesos para análisis en un solo cliente es algo innecesario ya que se generaran en ráfagas de información y no de forma continua por lo que aprovecharemos mucho mejor los recursos del servidor, sino que también nos permite escalarlo con la opción 2 de forma ilimitada según demanda, ahorrando coste computacional y energético, según como vaya creciendo la demanda de clientes dado el numero de productos que estén en servicio y el tiempo del día.

6. Seguridad

6.1. Análisis

6.1.1. Opción 1

Terrible opción que va en contra nuestro tercer principio, si bien esto toma poder computacional en el protocolo, por como se da el producto intrínsecamente es necesaria encriptacion en el protocolo

6.1.2. Opción 2

Esto ultimo no es mala idea, de fabrica anotando los dispositivos con rol único similar a lo que vendría a ser un MAC, nos ahorramos bando de ancha que nos puedan robar en un ataque DDoS moderado, y aseguramos que el servicio funcione de forma privativa o restringida al publico.

6.1.3. Opción 3

Afortunadamente por como se escogió el protocolo QUIC, esto esta dado, y en general es algo a lo que debemos apuntar en el diseño por los principios que nos impusimos al realizar el ejercicio por lo que esto es necesario, como antes mencionado esto va dado ya de por si por el protocolo con TLS 1.3 la ultima tecnología LTS dentro del protocolo TLS por lo que es bastante fuerte.

6.1.4. Opción 4

Realmente esta no es opción al fijar las IPs que vamos a dejar ingresar, si bien mucho mas barato en procesamiento de ataque que la segunda opción, nada que realmente no podamos hacer reaccionando dinámicamente a las condiciones que se presenten al servidor. No me parece terrible en el caso de que se sufra un ataque DDoS critico en el que directamente no tenemos forma de verificar conexiones genuinas, en cuyo caso probablemente del protocolo QUIC al ser layer de aplicación podríamos configurarlo para solo entregar servicio a las ultimas conexiones legitimas que vimos, pero por si solo simplemente no es viable, y le quitaría la opción de portabilidad al producto.

6.2. Decisión final

Acá realmente las opciones a tomar son la segunda y tercera, con la ultima excepción que se podría generar en el caso explicado también la cuarta, pero esto ya recaería en un análisis de ciberseguridad. dentro de lo que nos respecta guiándonos por nuestros objetivos anteriores serian justamente la opción de dispositivo/password y de encriptacion, que viene de caja con nuestro protocolo de transporte QUIC

Overview de selecciones y cohesion

Finalmente el producto en visionado en el ejercicio toma en cuenta los objetivos que se explicaron en un inicio, intentando resguardar en primera opción la comunicación estable, comunicación ágil y dentro de toda medida la seguridad.

El dispositivo diseñado intenta por todos los medios posibles primero, tener conexión a internet para el servicio, usa tecnología confiable para hacer este un producto robusto en cuanto a la comunicación IP, en transporte se usa adecuadamente el protocolo que muestra ser superior para nuestro caso al generar una red de IOT, dentro de los problemas DNS se busca la redundancia y la agilidad de forma inteligente para asegurarnos de la conexion cliente servidor, el diseño de servidores es apropiado para algun modelo de negocio dado y para la correcta cobertura ágil que se podría requerir, y dentro de la seguridad se toman todas las opciones razonables para su correcto resguardo.