

Preparacion C2

Control 2 - otoño 2024

Profesor: Rodrigo Arenas
Auxiliar: Francisco Almeida
Ayudantes: Cristian Romero , Felipe Alfaro

1. P1: DNS

1.1. Servidores raíz

Hace algunos años se hizo una campaña mundial para instalar servidores raíz espejo en todos los países del mundo, usando anycast. En Chile instalamos varios. Esto permite que casi todas las consultas DNS que necesitan un servidor raíz se quedan dentro del país. En teoría, nunca hay muchas consultas de ese tipo. Sin embargo, en la práctica hay muchas. Averigüe:

1. ¿qué tipo de consultas generan mucho tráfico hacia los servidores raíz?
2. ¿en qué ayuda entonces instalar copias locales en todas partes?

1.2. Servidores secundarios

Cuando creo un dominio popular, siempre existe la pregunta de cómo configurar su DNS. Suponga que Ud debe decidir cómo configurar el DNS de su empresa, que se ha vuelto súbitamente popular en el mundo y requiere respuestas rápidas de todas partes y no debe fallar nunca. Los ingenieros le proponen distintos esquemas, analice sus pros y contras y elija uno de ellos justificando su elección (en este caso, también vale proponer una mezcla entre ellos):

1. Armar una lista de 6 servidores de nombres repartidos por el mundo
2. Armar una nube anycast en Chile y poner ahí todos los servidores (un sólo NS en la lista que apunta a esa nube)
3. Contratar una nube anycast mundial existente y poner ahí todos los servidores (un sólo NS en la lista que apunta a esa nube)

Suponga que los costos monetarios de la implementación son irrelevantes para el proyecto en este caso.

1.2.1. Magia del DNS

El DNS es terriblemente resiliente y funciona incluso cuando está mal configurado un dominio. Un ejemplo es el dominio: itau.cl que vieron en una actividad evaluada.

Haga un análisis de su configuración actual y sus errores:

1. ¿Qué está mal configurado?
2. ¿Qué problema le puede generar al banco este error?
3. ¿Cómo pueden tener este error hace años y no haberlo notado?

2. P2: Protocolos Clásicos

2.1. Algoritmos clásicos

En general, se considera que Stop-and-Wait es mejor con delay muy bajo y poca pérdida, Go-Back-N es mejor con delay alto y poca pérdida y Selective Repeat es mejor con delay alto y pérdida alta.

Esto, porque no vale la pena pagar la complejidad de un algoritmo avanzado si el más simple funciona bien.

Ahora, supongamos que un físico loco inventó un enlace instantáneo (delay cero) pero con altísima pérdida (digamos 60 %). ¿Qué protocolo recomendaría usar? ¿Por qué?

2.2. Números de secuencia

Al usar números de secuencia finitos, existe el riesgo de recibir un paquete demasiado atrasado cuyo número de secuencia coincida con el rango de la ventana que espero ahora y aceptarlo erróneamente. Responda las siguientes preguntas, suponiendo que estamos en un Selective Repeat:

1. Si me equivoco y acepto ese paquete erróneo: ¿Qué error genero en el stream que llega? ¿Qué pasa con el paquete correcto que llevaba ese mismo número?
2. ¿Me ayuda usar ventanas más grandes?
3. ¿Qué podemos hacer para disminuir la probabilidad que estos errores ocurran?
4. Si conozco un límite máximo de tiempo que un paquete demora en la red en llegar a su destino: ¿Puedo encontrar una solución que garantice que nunca voy a recibir un paquete erróneo?

3. P3: Simulador

3.1. BDP y Pérdida

Un problema con las ventanas de los protocolos clásicos es definir qué tamaño óptimo deben tener cuando conozco el BDP y la probabilidad de pérdida. Claramente es bueno que sean mayores al BDP, pero ¿qué tanto más grande?

Un ingeniero postula la conjetura que hay que usar ventanas de tamaño $BDP \cdot (1+p)$. Es decir, agrandar la ventana proporcionalmente a la pérdida. Es decir, si la pérdida es 0% usamos ventana de tamaño BDP. Si es 20% usamos $1.2 \cdot BDP$ (un 20% más grande) y así. Use el simulador del curso para probar esta hipótesis y reporte sus resultados: ¿puede ser correcta esa conjetura? Si es correcta, trate de explicar por qué será. Si es incorrecta, trate de postular una propuesta alternativa.

El objetivo a lograr en el simulador es aproximarse al Useful Bandwidth obtenido sin pérdidas. Entonces, en una configuración con 5.000 de delay, 10.500 de timeout y 59 paquetes/minuto, una ventana de 11 paquetes cubre el BDP y (sin pérdidas) me da casi 1 de Useful Bandwidth. Al haber pérdidas, aumentará el consumo de ancho de banda pero bajará el Useful Bandwidth.

Usen el simulador: <http://users.dcc.uchile.cl/jpiqueer/srgbn2.html> para poder definir ventanas grandes. No activen la ventana de congestión y pongan el Network Capacity al máximo (2).

OJO: El simulador corre en javascript en el navegador, y eso hace que si la pestaña del simulador no está visible, el navegador la inactiva y deja de ejecutar hasta que es visible otra vez. Por lo tanto, para hacer pruebas un poco más largas, deben dejar la pestaña en primer plano y visible en la pantalla para medir bien los anchos de banda.

3.2. Consumo de Ancho de Banda

En el simulador del curso, se muestran tres valores para el ancho de banda: el Consumo Total (gasto de ancho de banda acumulado), el consumo Actual (consumo total en un delta de tiempo) y el Consumo útil (transmisión real lograda, sería como lo que mediría la aplicación).

Al haber pérdidas y retransmisiones, es normal que el consumo total suba, mientras que el útil baja. Aunque la generación de paquetes sea fija (típicamente 1 paq/s en nuestros ejemplos), el consumo total puede ser mayor que 1, ya que las retransmisiones se hacen por sobre la generación de paquetes nuevos. Ojo que la probabilidad de pérdida se aplica tanto a los paquetes como a los ACKs respectivos.

Experimentando con el simulador, y pensando en cómo funcionan los protocolos de retransmisión, responda las siguientes preguntas:

1. ¿Cuál sería el valor óptimo que uno quisiera lograr de consumo de ancho de banda total si hay una pérdida con probabilidad p ?

2. ¿Cuál sería el valor óptimo de Consumo útil con pérdida p ?

HINT: No existe una respuesta correcta publicada para estas preguntas, o, al menos, nosotros no la conocemos. Lo que importa es cómo argumentan y explican su idea.

HINT2: Para esta pregunta usen el simulador `srgbn2` también.