CC4303 - Redes
Profesora Ivana Bachmann
Auxiliar Vicente Videla
Ayudantes Gonzalo Alarcón, Valentina Esteban, Pablo Villar

CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE CHILE

Auxiliar 1

HTTP

22 de marzo, 2023

Conceptos nuevos pertinentes

- Conexiones persistentes: se reutiliza un socket de conexión para enviar varios datos entre el origen y el destino.
- Conexión no persistente: se cierra una vez fue utilizada.
- connection: header para indicar qué tipo de conexión utilizar. Sus valores son "keep-alive" para persistente, "closed" para no persistente.
- Round-Trip Time (RTT): tiempo en milisegundos de cuánto se demora un solicitud en ir de origen a destino, y luego de vuelta al origen.

P1. Servidores HTTP

Suponga que un web server corre en el host C en el puerto 80; este server usa conexiones persistentes y está recibiendo requests de dos hosts diferentes, A y B. ¿Se procesan todas las requests en el mismo socket de C? ¿Si se pasan a sockets distintos, tienen ambos sockets el puerto 80? Explique.

P2. Conexión persistente

Considere que establecer conexión en un socket orientado a conexión se demora 3RTT y cerrar dicha conexión se demora 2RTT. Suponga una página HTML referencia 8 pequeños objetos en el mismo server. Asumiendo que el tiempo de descarga de los objetos es despreciable y que no se pueden establecer conexiones paralelas. ¿Cuánto tiempo toma descargar la página si:

1. se usan conexiones no persistentes?
2. se usan conexiones persistentes?

P3. Sockets

1. ¿Qué datos básicos necesita un socket orientado a conexión y cuáles necesita uno no orientado a conexión para mandar cosas?
2. ¿Cuál es la gran diferencia entre un socket orientado a conexión y uno no orientado a conexión?
3. ¿Cómo funcionan los sockets no orientados a conexión? ¿Por qué alguien los preferiría?

P4. HTTP

1. ¿De qué forma viene la información en un mensaje HTTP?
2. ¿Si un agente malicioso bloquea el puerto 80 de su PC, podrá visitar páginas web?
3. ¿Qué ocurre si envío un mensaje HTTP sin el doble salto de línea "\r \n \r \n" que separa el HEAD del BODY?

P5. Verdadero y Falso

1. HTTP es un lenguaje de marcado y HTML es un protocolo de comunicación.
2. El HTTP response (la respuesta) se compone de headers o datos, no ambos.
3. El protocolo HTTP no guarda información de la conexión, para eso se usan los headers / cookies.
4. El protocolo HTTP utiliza comunicación no orientada a conexión, para evitar delays innecesarios.
5. Un servidor HTTP puede recibir una request y antes de responder, puede hacer consultas a otros servidores, formando una cadena de preguntas y respuestas.
6. Los headers siguen un orden determinado, por ejemplo 'Content-Type' siempre viene antes que 'Content-Length'.
7. El header 'Content-Length' indica el largo del mensaje HTTP, incluyendo los headers.
8. En caso de recibir de forma incompleta el BODY, puedo volver a hacer 'receive'.
9. Usando mensajes HTTP puedo recibir tanto texto plano como imágenes.