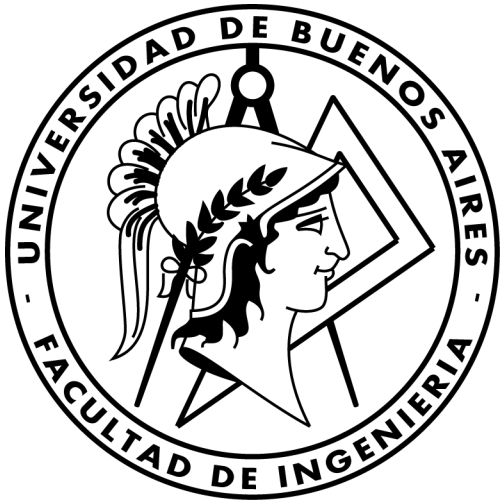

7543 - Introducción a los Sistemas Distribuidos

Facultad de Ingeniería de la universidad de Buenos Aires

Trabajo Práctico 1
DNS on HTTPS

Repositorio

Luciana Piazzi 90638
Gonzalo Marino 97794

Índice

1. Motivación de DoH

DoH es un protocolo para enviar consultas DNS y obtener respuestas DNS sobre HTTPS. Cada par de de consultas/respuestas DNS se asignan en un intercambio HTTPS. Además posee las características de HTTPS, las cuales son la redirección, la representación, el almacenamiento caché, autenticación y compresión. Este protocolo solventa la confiabilidad en los mensajes DNS. Un servidor se denomina "servido DoH" si este admite este protocolo, para diferenciarlo de un "servidor DNS". Del mismo modo, un cliente que admite este protocolo se lo denomina "cliente DoH".

La motivación principal para utilizar un sistema que permita resolver dominios de forma tal que no se acceda a la infraestructura pública de resolución DNS, DoH viene a resolver está situación en la mayoría de los casos. Uno de los objetivos principales de este método es aumentar la privacidad y la seguridad de los usuarios mediante la prevención de las escuchas ilegales y la manipulación de los datos DNS a través de ataques de intermediario. DoH cifra el tráfico DNS y requiere autenticación del servidor, esto mitiga la vigilancia pasiva y los ataques activos que intentar desviar el tráfico DNS a servidores no autorizados. Además el uso del puerto 443 predeterminado para HTTPS y la capacidad para mezclar tráfico DoH con otro tráfico HTTPS en la misma conexión puede disuadir los dispositivos en ruta no privilegiados de interferir con operaciones DNS y hacen que el análisis del tráfico DNS sea más difícil.

2. Comparación entre DoH, DoT y DNS

Sabemos que DoH como DoT y DNS son tres sistemas de nombre de dominio y cuyo propósito general es el de apuntar los dominios al server correspondiente. Estos sirven para traducir la dirección real, la cual es una relación numérica mejor conocida como dirección IP, en un nombre del dominio, dado que las personas cuando navegan por la web ingresar una URL (Uniform Resource Locator o Localizador Uniforme de Recursos) y no la dirección IP real. Un servidor DNS toma dicha URL y encuentra la dirección IP correspondiente. El servidor DNS local recibe una consulta desde el resolver del host cliente, para luego realizar las consultas de manera iterativa a los servidores correspondientes. De esta manera el servidor DNS local entrega la resolución al host inicial que pidió la información, haciendo que el resolver

del host cliente entregue la respuesta a la capa de aplicación correspondiente. Inicialmente esta forma de resolver las consultas no tuvo en cuenta las distintas cuestiones relacionadas a la seguridad pero esta tuvo que adaptase, modificando los requisitos de seguridad para proteger la integridad de los datos y la autenticación de los usuarios. De esta reforma surgieron las variantes DoH y DoT.

DNS over TLS (DoT) y DNS over HTTPS (DoH) son protocolos que podrían ser interpretados con propósitos diferentes, pero en realidad estos logran el mismo objetivo: ambos estándares cifran las solicitudes DNS, aunque hay una gran diferencia entre los protocolos que estos dos utilizan. DNS sobre TLS utiliza TCP como protocolo de conexión básico y capas sobre cifrado y autenticación TLS. DNS sobre HTTPS utiliza HTTPS y HTTP 2 para realizar la conexión. Esto es importante de mencionar ya que afecta al puerto por el cual operan. DoT tiene su propio puerto (puerto nro. 853) definido en [RFC 7858 Y RFC 8310], en cambio DoH utiliza el puerto 443 el cual es el puerto estándar para el tráfico HTTPS [RFC 8484]. Esta claro que ambos servicios llevan las consultas encriptadas a través de la red, pero en la actualidad no podría definirse una verdad absoluta acerca de cuál de estos dos protocolos es el mejor. Se podría pensar que la ventaja de DoT es la utilización de un puerto exclusivo para las consultas DNS a tráves de TLS, pero la ventaja de DoH es que las solicitudes pueden ocultarse en el resto del trafico cifrado, por lo que las solicitudes DoT al usar un puerto donde cualquier persona en el nivel de red puede ver las solicitudes fácilmente o incluso bloquearlas.

3. Suposiciones y/o asunciones

- Se supuso que aunque sea un estudio sobre DoH, se utiliza http para simplificar el trabajo, ya que en el código base propuesto por la cátedra, el servidor corría sobre http, por lo que en el fondo nuestro código es vulnerable a un "man-in-the-middle attack", lo que significa que, si el sistema corriera distribuido, cualquiera que esté capturando los paquetes que envíamos y recibimos conocerá las consultas DNS hechas al servidor.

- Se supuso que las IP que nos enviaran estarían en formato válido, ya que no se le hace ningún tipo de validación.

- Cuando se hace un PUT, suponemos que los dominios van a ser iguales.

4. Definición de la API utilizando la especificación OpenAPI

La definición se encuentra en el archivo: swagger.yml

5. Dificultades encontradas

- La dependencia entre swagger y el código hacía que a veces nuestro código ande corriendolo con cURL, pero al probarlo desde swagger no. Eso trajo algunos problemas ya que podían pasar errores desapercibidos (que en realidad eran errores en la documentación)
- La resolución de direcciones IP de servidores se hizo utilizando round robin. Esto puede no ser lo más eficiente.

6. Conclusión

Encontramos que hacer un servidor de DoH es sencillo y puede traer el beneficio de ocultar, utilizando mensajes cifrados, los pedidos DNS que hacen los clientes a los local resolvers. Si se utiliza un mismo local resolver para responder a una gran cantidad de clientes, entonces un atacante no sabría qué dominios está pidiendo cada cliente, ya que del lado del cliente vería un mensaje https hacia el servidor, y desde el servidor vería pedidos DNS, sin saber qué cliente pidió cuál. Sin embargo se se hace un control de tráfico exhaustivo, se podría llegar a saber qué dominios los clientes están consultando. El principal problema es que para esta solución necesitamos establecer una conexión TCP con el local resolver, en cambio en una resolución de DNS normal, ésta se hace por UDP, y en general es más rápida.