ISA – Síťové aplikace a správa sítí

Varianta: Tunelování datových přenosů přes DNS dotazy

Zuzana Hrkľová

ak.r.2022/2023 xhrklo00

Úvod do problematiky

Cieľom je vytvorenie klient/serverovej aplikácie, ktorá bude vedieť tunelovať dáta cez DNS dotazy.

Na to, aby bolo možné posielať akékoľvek dáta zo vstupu pomocou DNS dotazov je potrebné vytvorenie validneho DNS paketu a zakódovanie posielaných dát do povolených znakov doménového mena.

Server musí následne prijaté dáta vedieť rozkódovať.

Popis implementácie

Klient

Ako prvé prevedie kontrolu vstupných parametrov. Pokiaľ nebol spúšťaný so vstupným parametrom `[-u UPSTREAM_DNS_IP]`, IP adresu na ktorú bude DNS pakety zasielať, získa zo súboru `/etc/resolv.conf`. Pomocou funkcie `encode_basehost()` prevedie basehost do predpísanej formy DNS doménového mena, funkciou `encody_bytes_to_base16()` zakóduje cestu, pod ktorou sa dáta uložia na servery, do hexadecimálnej podoby a nakoniec spojí zakódovanú cestu spolu s predpripraveným basehostom, čím vytvorí jedno doménové meno.

Klient použije štruktúry `dns_packet` a `dns_end` na zostavenie celého DNS paketu a zašle ho na server pomocou UDP.

Pri zasielaní prvého DNS paketu sa vždy nastaví ID paketu na 0, aby bol server schopný rozoznať, že daný paket je prvý a teda obsahuje cestu, pod ktorou má prijaté data ukladať.

Po odoslaní prvého paketu klient postupne číta celý vstup po 20B, prekonvertuje ich do hexadecimálnej podoby pomocou spomínanej funkcie a zasiela na server.

Po postupnom spracovaní načítavaných dát klient odošle ešte jeden posledný DNS paket indikujúci koniec komunikácie so serverom so správou `close.basehost` kde `close` nešifruje do hexadecimálneho tvaru.

Po odoslaní tohto paketu ukončí svoju činnosť.

Server

Po spustení server začne čakať na prichádzajúce správy. Ako náhle prijme správu od klienta, rozdelí basehost od ostatných dát a porovná ho s tým, ktorý mal zadaný na vstupe. Po prípadnej zhode skontroluje, či dáta pred basehostom neobsahujú správu `close`. Ak áno, znamená to koniec komunikácie s aktuálnym klientom a teda zavrie otvorený súbor, kde zapisoval dáta z predchádzajúcej správy a čaká na ďalšiu správu.

Ak však správa neobsahuje `close`, rozšifruje ju pomocou funkcie `decode_base16_to_bytes()` a skontroluje ID v DNS hlavičke. Ak je ID rovné 0, znamená to, že daný paket je prvý v poradí od aktuálneho klienta a teda obsahuje cestu. Klient skontroluje či daná cesta existuje a prípadne ju vytvorí. Následne otvorí súbor, aby bol pripravený na zápis.

Ak je v ID DNS hlavičky iné číslo ako 0, rozkódované dáta sa zapíšu do otvoreného súboru.

Nedostatky riešenia

V aktuálnom riešení server nepodporuje komunikáciu s viacerými klientami zároveň. Taktiež nie je vytvorená žiadna kontrola na strane klienta, či bol odoslaný paket doručený, a tak môže dôjsť k strate dát pri prenose.

Návod na použitie

Klient:

```
./dns_sender [-u UPSTREAM_DNS_IP] {BASE_HOST} {DST_FILEPATH} [SRC_FILEPATH]
```

-u : slúži na vynútenie vzdialeného DNS serveru, ak nie je špecifikovaná program využije DNS server nastavený v systéme

BASE_HOST : slúži k nastaveniu bázovej domény

DST_FILEPATH : cesta pod ktorou sa dáta uložia na servery

SRC_FILEPATH : cesta k súboru, ktorý bude odoslaný, ak nie je špecifikovaná použije sa STDIN

Server:

```
./dns_receiver {BASE_HOST} {DST_DIRPATH}
```

BASE_HOST : slúži k nastaveniu bázovej domény na prijímanie dát

DST_DIRPATH : cesta pod ktorou sa budú všetky prichádzajúce dáta ukladať

Testovanie

Testovanie bolo prevedené na `.txt` a `.jpg` súboroch. Správnosť DNS dotazov bola overená pomocou programu Wireshark.