VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Síťové aplikace a správa sítí

Přenos souboru skrz skrytý kanál

15. listopadu 2021

Daniel Olearčin

Obsah

1 Úvod

Tento dokument slúži na pochopenie projektu spracovaného v predmete ISA. Cieľom projektu je vytvoriť klient/server aplikáciu ktorá umožní prenášat súbor cez skrytý kanál. Data majú byť prenášané v ICMP Echo-Request/Response správach. Súbor sa má šifrovať pomocou šifry AES, dostupnú v knižnici ssl kde ako kľúč sa požije login. V prípade väčších súborov je potrebné súbor rozdeliť na osobitné pakety a na strane serveru ich naspäť spojiť do jedného súboru a uložiť pod rovnakým menom. Celý projekt musí byť v C/C++ programovacom jazyku.

2 Implementácia

2.1 Spracovanie argumentov

Hned pri spustení programu sa vchádza do funkcie `arguments_parsing`, v ktorej sa ošetrujú argumenty ktoré zadal používateľ. Funkcia ukladá do premenných `file`, `host` a `l_arg` zadané argumenty a v prípade zadania zlých argumentov vypisuje chybovú hlášku v podobe krátkeho ozrejmenia použitia správnych argumentov. Pri zadaní argumentu -l sa na ostatné argumenty neberie ohľad a spúšťa sa server na počúvanie prichádzajúcich paketov.

2.2 Vytvorenie servera

Pri vytváraní serveru som sa zo značnej časti inšpiroval súborom sniff-filter.c, ktorý nám bol dostupný v examples v predmete ISA. Na začiatku som sa pozrel na všetky dostupné prostredia na zariadení (funkcia `pcap_findalldevs`) a vyberal prvé dostupné. Následne som uložil informácie (IP a masku) o prostredí (funkcia `pcap_lookupnet`) a otvoril prostredie (funkcia `pcap_open_live`). Nastavil som filter na icmp (funkcia `pcap_compile` a následne `pcap_setfilter`). Nakoniec pomocou funkcie `pcap_loop` som začal nekonečný cyklus ktorý zachytáva pakety a predáva ich funkcii `mypcap_handler`, ktorá sa stará o :

- Identifikovanie paketu a zistenie podľa prvých znakov či sa jedná o správny paket.
- Následné získanie mena súboru pre uloženie a počtu znakov ktoré je potrebné z paketu na konci odstrániť.
- Dešifrovanie šifrovanej časti paketu.
- Otvorenie súboru / vytvorenie súboru a následné pridávanie dešifrovanej časti paketu.
- Dealokácia premenných.

Po celý čas som musel pracovať s dátami binárne pretože inak dochádzalo k nesprávnemu dešifrovaniu napríklad obrázkov...

2.3 Vytvorenie klienta

Na začiatku sa pokúšam dostať informácie o IP adrese alebo doménovom mene zadanom v argumentoch. Následne sa pokúšam otvoriť socket pomocou týchto informácií. Takisto sa snažím otvoriť súbor daný v argumentoch. Pri zlyhaní vypisujem chybové hlášky. Ak všetko prejde v pohode tak začínam načítavať súbor po znakoch do jednotlivých paketov. Nastavené to mám na maximálne 1024 znakov na jeden paket aby som si bol istý že nepresiahnem limit (1500). K týmto znakom pridávam na začiatok:

- 1. << aby som vedel identifikovať paket na strane serveru.
- 2. Meno súboru ktorý klient odosiela na server.
- 3. < Aby som vedel kde končí meno súboru.
- 4. Dvojciferné čislo od 1 do 16 ktoré udáva koľko znakov bolo umelo pridaných k textu pre správne šifrovanie.
- Príklad: `<<filename.txt<05`

Tieto pridané znaky spolu s 1024 znakmi so súboru zašifrujem a zašifrovaný text nakopírujem do predom pripravené paketu. Pomocou funkcie `sendto` tento paket odošlem na server. Nakoniec sa dealokuje všetka alokovaná pamäť. Takisto ako pri serveri som musel pracovať s datami binárne aby nedochádzalo k chybám pri šifrovaní.

3 Návod na použitie

Program sa spúšťa s roznymi parametrami ktoré sú vysvetlené nižšie.
`./secret -r <file> -s <iphostname> [-l] [--help]`

- `-r <file>` : Argument `-r` označuje že následujúci argument obsahuje cestu ku súboru ktorý sa bude prenášať alebo meno súboru ak je v aktuálnom priečinku.
- `-s <iphostname>` : Argument `-s` označuje že následujúci argument udáva IP adresu alebo doménové meno kam sa má súbor zasielať.
- `-l` : Argument `-l` značí že chceme pustiť serverovskú časť programu. Pri zadaní argumentu `-l` sa všetky ostatné argumenty ignorujú a spúšťa sa server.(Samozrejme nieje možne to spustiť spôsobom `./secret -r -l` pretože vtedy berieme že argument -l je názov súboru. Takisto to platí aj pri argumente -s.)
- `--help` : Argument `--help` Vypíše krátku nápovedu.

3.1 Kompletný príklad

Otvoríme si 2x terminálové okno.V obidvoch oknách sa presunieme do zložky kde máme preložený program secret. V aktuálnej zložke si vytvoríme novú zložku s názvom examples a tam presunieme pripravený súbor ktorý chceme prenášať(V tomto prípade meno tohto súboru bude 4k.jpg). Zistíme IP adresu napríklad na stránke

https://whatismyipaddress.com/. ktorú si niekam zapíšeme.
V jednom okne spustíme server s príkazom
`sudo ./secret -l`
V druhom okne pošleme tento paket na server pomocou príkazu
`sudo ./secret -r ./examples/4k.jpg -s IPADDRESS.`
Overíme že do akutálnej zložky sa nakopíroval súbor 4k.jpg zo zložky examples.

4 Literatúra

- https://beej.us/guide/bgnet/html/client-server-background.
- https://man.openbsd.org/AES$_encrypt$.3.
- https://www.tcpdump.org/manpages/pcap.3pcap.html.