Síťové aplikace a správa sítí
2021/2022

Přenos souboru skrz skrytý kanál

12. listopadu 2021, Brno

Lukáš Plevač (xpleva07)

1. Úvod

Toto je manuál pro TUI aplikaci secret, která slouží pro skrytý šifrovaný přenos dat pomocí ICMP echo packetů

1.1. použití

secret -r <file> -s <ip|hostname> | -l [-h]

program lze spustit ve dvou módech. Jednim je mód naslouchací, který zapnete parametrem -l, který vyžaduje parametr -r pro zadani vystupniho souboru. Druhy mod je mod odesilaci který je zapnut implicitně a vyžaduje parametry -r a -s.

1.2. parametry

-r urcuje soubor se kterým chceme pracovat - přijmout nebo odeslat
-s určuje adresu (IP nebo hostname) cile kam chceme data poslat
-l spuštění naslouchacího módu (Nutné spustit jako root nebo s opravnenim k interface)
-h vypíše nápovědu

1.3. return kódy

pokud aplikace selze spíše na stderr informace o chybe a ukončí se s kódem který je rozdílný od 0, pokud aplikace vse dokončí jak má, skončí s kódem 0.

2. Ukázka použití

2.1. PočítačA (10.0.0.10) příjemce

#./secret -r image.jpg -l

2.2. PočítačB (10.0.0.11) odesílatel

$./secret -r image.jpg -s 10.0.0.10

2.3. průběh přenosu

Po spuštění aplikace na příjemci, byla aplikace spuštěna i na odesílateli. Po spuštění aplikace v listen módu aplikace čeká na init packet a vytvoří si soubor pro ukládání dat z přenosu, pokud existuje tak jej přepíše. Init tento packet odesílá aplikace v módu odesílatele při spuštění. Po tom co aplikace dostane init packet začne přijímat packety od počítače co jí tento init packet poslal. Po této synchronizaci se na obou počítačích začne na STDOUT vypisovat kolik packetů zbývá do dokončení přenosu. Po dokončení se obě aplikace ukončí.

3. Limity

aplikace kvůli limitům svého protokolu dokáže přenést pouze soubory o maximální velikosti 1600GiB, nicméně pro rychlé sítě může být problém i 200MiB, kdy může dojít k nenávratným ztrátým paketů způsobenými přetečením bufferu pcap. Tento buffer je nastaven na 100MiB, takže maximální velikost souboru který by měl jít vždy bezproblémově přenést je 100MiB. Tento buffer lze přenastavit v server.h až na 2GiB.

Ztrátovost protokolu na sítích se ztrátovostí je jen částečně ošetřena, takže při přenosu může dojít ke ztrátě dat

4. Vývoj a závislosti

4.1. Závislosti

aplikace je implementována v c++ s knihovnami crypto, pcap. Pro překlad jsou nutné aplikace g++ a make

4.2. Makefile

4.2.1. build [default]

zkompiluje zdroje do secret

4.2.2. debug

zkompiluje zdroje do secret-debug s podporou ladících hlášek a informací

4.2.3. install

zkompiluje a nainstaluje aplikaci do systému

4.2.4. uninstall

odinstaluje nainstalovanou aplikaci z systému

4.2.5. clean

vyčistí repozitář od binárních souborů

5. Protokol

Protokol je navržen velmi jednoduše. Při přenosu jsou používány pouze ICMP echo packety, do jejich BODY je uložen obsah který má být zaslán. První vyslaný packet (init packet) slouží jako hlavička a má následující strukturu

```
typedef struct {
    u_char protocol[10];  //jméno protokolu a verze
    uint32_t blocks_count; //počet packetů které budou zaslány
    uint32_t block_size; //max velikost jednoho packetu
    u_char iv[MAX_IV_LEN]; //inicializační vektor pro dešifrování
} icmp_enc_transf_hdr;
```

další packety již obsahují pouze data zašifrovaného souboru

6. Implementace

6.1. Zpracování argumentů

Zpracování argumentů probíhá pomocí getopts, argumenty jsou přiděleny do příslušných proměnných a následně pomocí několika if větví je rozhodnuto zda jsou argumenty správné.

6.2. Odeslání PING packetu (ICMP ECHO)

Odesílání packetů je implementováno pomocí DGRAM socketů. Odesílání se provádí pomocí funkce send která oddelene existuje pro ICMPv6 a ICMP. správná funkce send se vybere podle typu adresy která byla přeloženo pomocí

addrinfo. Odesílání souborů je implementováno jako opakované volání této funkce send. Přičemž jako první bude odeslán init packet a následně v cyklu bude čteno ze souboru šifrováno a odesílano. Po každém odeslání se čeká na ICMP REPLY, pokud nebude doručeno packet se pošle znova. Po třech pokusech aplikace ukončí pokus na současnou adresu a zkusí další adresu s listu adres z addinfo.

6.3. Příjem packetů

Příjem packetů je implementován pomocí pcap. Po vytvoření objektu server dojde k otevření zařízení “ANY” pokud se tato operace nezdaří aplikace se ukončí s chybou. Před otevřením se nastaví filtr pro pouze příchozí ICMP packety typu ECHO a ostatní budou zachovány. Následně vždy bude odchycen packet dekodovan do urovne ICMP následně přečten a zjisti se zda se nejedna o INIT packet pokud ano bude zapamatován a přejde se do stavu přijímání packetů. Následně funguje tak, že se prijme packet, zjistí se jisti má stejnou src IP a id jako init packet a pokud ano dešifruje se a uloží se do souboru.

7. Závěr

Základní funkce aplikace byly úspěšně implementovány, nicméně zbyla spousta prostoru na zlepšování aplikace, především v oblasti bezztrátovosti při přenosu.