



Internet Control Message Protocol

ICMPv4

KIV/PSI - semestrální práce

student: Radek VAIS
os. číslo: A17N0093P
mail: vaisr@students.zcu.cz
datum: 10.6.2018

1 Zadání

Úloha bude mít dvě části. Část analytickou a část syntetickou. V analytické části budete analyzovat pakety nebo rámce vybraného protokolu. Jako zdroj poslouží nějaký standardní program. V Syntetické části vygenerujete vlastní pakety nebo rámce, odešlete je ověřenému (standardnímu) protějšku, který vygeneruje odpověď. Jak Vaši výzvu, tak i odpověď zachytíte prvním programem a zobrazíte. Pochopitelně první i druhá část může tvořit jeden celek.

Předkládané programové vybavení musí obsahovat i část ladicí, která nebude samoúčelná, ale bude sloužit k odladění programu i k odladění výměny zpráv. Ladicí výpis by měl obsahovat zejména časový údaj, dále pak informaci o úrovni ladění a vlastní ladicí text. Můžete se inspirovat Syslogem.

Programová a uživatelská dokumentace bude součástí komprimovaného balíčku, který odevzdáte. Balíček bude kromě toho také obsahovat zdrojové kódy v samostatných adresářích a makefile pro Linux. Dokumentace v tištěné podobě se neodevzdává. Semestrální práce se odevzdávají na Courseware.

Zvoleným protokolem je ICMP - Internet Control Message protocol ve verzi 4.

2 Analýza

Dle RFC-792 Protokol ICMPv4 slouží k technické kontrole sítě a úpravě toku dat a reportování chyb. K přenosu zpráv mezi uzly je použit protokol IP, z toho důvodu ICMP nemůže detekovat chyby IP protokolu. Typickou situací generování ICMP zprávy je nedoručitelný datagram z důvodů malé vyrovnávací paměti uzlu na cestě. Nejedná se o spolehlivý protokol, ani není jeho cílem udělat IP protokol spolehlivým. Hlavním důvodem vzniku protokolu bylo předávání zpětné vazby o problémech v komunikaci. Na ICMP zprávy se nelze spolehnout při implementaci spolehlivosti vyšší vrstvy ze dvou důvodů. Prvním je, že v případě chyby ICMP zpráva nemusí být vygenerována. Druhým je fakt, že při chybě přenosu ICMP zprávy se negeneruje nová ICMP zpráva. Takové chování by způsobovalo zahlcení "poškozené" sítě.

IP záhlaví zprávy ICMP protokolu vypadá následujícím způsobem:

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|Ver = 4|  IHL  |Ty. of Ser. =0 |                               Total Length                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Identification                               |Flags|       Fragment Offset       |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|  Time to Live | Protocol = 1 |                               Header Checksum                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Source Address                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Destination Address                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Data....
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

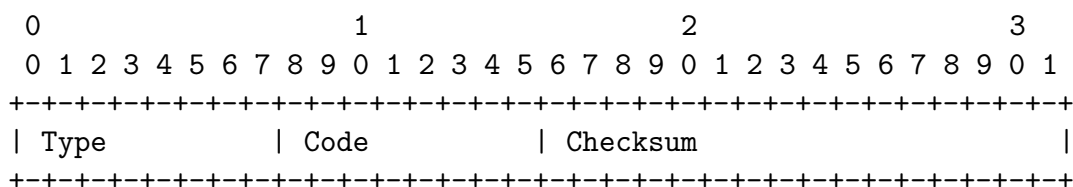
```

Jednotlivá pole IP záhlaví jsou definovány následujícím způsobem:

- Ver = version = verze IP protokolu - hodnota 4 pro IPv4.
- IHL = internet header lenght = velikost IP záhlaví.
- Ty. of Ser. = Type of service = nároky na zpracování zprávy - hodnota 0 pro ICMP zprávy.
- Total Length = délka celé zprávy.

- Identification, Flags, Fragment Offset - pole pro informace o fragmentaci zprávy.
- Time to Live = doba života zprávy.
- Protocol = Protokol nesený IP zprávou - hodnota 1 pro ICMP.
- Checksum = kontrolní součet - doplněk součtu 16 bitových slov.
- Source Address = zdrojové adresa IPv4.
- Destination Address = cílová adresa IPv4.
- Data = prostor pro data nesená ICMP protokolem.

Záhlaví ICMP zprávy je dlouhé 8 bytů, většina zpráv využívá pouze první 4byty, které jsou definované následujícím způsobem:



Jednotlivá pole ICMP záhlaví jsou definovány následujícím způsobem:

- Type = Typ zprávy ICMP.
- Code = Přesnější zařazení ICMP zprávy.
- Checksum = Doplněk součtu 16 bitových slov celé ICMP zprávy.

2.1 Typy zpráv

Přehled ICMP zpráv:

1. Echo Reply (Type = 0)
2. Destination Unreachable (3)
3. Source Quench (4)
4. Redirect (5)
5. Echo (8)

6. Time Exceeded (11)
7. Parameter Problem (12)
8. Timestamp (13)
9. Timestamp Reply (14)
10. Information Request (15)
11. Information Reply (16)

2.1.1 Destination unreachable

Pokud nebyl nalezený záznam o žádané síti, protokolu, zařízení nebo portu je odeslána zpráva "Destination unreachable". Pole Code obsahuje identifikaci, jaký element nebyl nalezen. Dále je v datech ICMP zprávy uložena hlavička původního IP packetu a 64 bitů dat.

2.1.2 Time Exceeded

Pokud packetu vypršela doba života, uzel je povinen ho zahodit. A může notifikovat zdroj zprávou "Time Exceeded", která v datech obsahuje IP záhlaví originální zprávy + 64 bitů dat. Pokud k vypršení času došlo v důsledku čekání na jednotlivé fragmenty zprávy, bude pole Code nastaveno na hodnotu 1.

2.1.3 Parameter problem

Pokud uzel při přijetí IP packetu zjistí, že některá z hodnot IP záhlaví je chybná odešle zprávu "Parameter problem". V datech této zprávy bude uvedeno IP záhlaví originální zprávy + 64 bitů dat. Dále tento typ zprávy využívá další byte záhlaví, pro označení ve kterém bytu záhlaví zprávy došlo k chybě.

2.1.4 Source quench

V případě, že uzel zahodil zprávu z důvodů malého bufferu pro zpracování, může notifikovat zdroj zprávou "Source quench". V datech této zprávy bude uvedeno IP záhlaví originální zprávy + 64 bitů dat.

2.1.5 Redirect

Zpráva slouží k upravení směrování na lokálním segmentu. V případě, že je do hraničního směrovače oblasti odeslána zpráva, kterou směrovač odešle jinému směrovači ve stejné síti, zároveň odešle zprávu "Redirect". V dalších 4bytech záhlaví je uložena IPv4 adresa nové brány a data obsahují originální IP záhlaví + 64 bitů dat.

2.1.6 Echo a Echo Reply

Zpráva slouží k ověření, zda cílový uzel je v provozu. Tato ICMP zpráva využívá všech 8byťů záhlaví. První dva doplňující byty obsahují identifikační hodnotu zprávy druhé dva pak sekvenční číslo jednotlivých zpráv. Dotazovaný uzel je povinen odeslat zpět celou část uvedenou v poli data zpět zdrojovému uzlu.

2.1.7 Timestamp a Timestamp Reply Message

Zpráva slouží ke zjištění odhadu časového průchodu zprávy sítí. Stejně jako zpráva Echo využívá zpráva Timestamp celé záhlaví ICMP se shodným významem identifikace a sekvenčního čísla. V datech pak využívá 3x32 bitů pro uložení časů odeslání, přijetí na vzdáleném uzlu a odeslání na vzdáleném uzlu. Čas je uložen ve formátu počet *ms* od půlnoci času UTC.

2.1.8 Information Request a Information Reply Message

Zprávy slouží ke zjištění adresy sítě, kde je host připojen. Je možné odeslat zprávu s nevyplněnou cílovou IP adresou. Opět je využito celé ICMP záhlaví ve smyslu identifikace a sekvenčního čísla (viz Echo). Data nejsou odesílána.

2.2 Zranitelnosti

Základní zranitelností ICMP protokolu je nízká úroveň filtrace ICMP zpráv. Protože korektní fungování ICMP protokolu na síti je prospěšné správnému fungování, nejsou ICMP zprávy filtrovány firewally. Lze tak použít ICMP zprávu ECHO k přenosu dat na jiný kompromitovaný uzel. Protože je ICMP neautorizovaný, je často využíván k DoS útokům. Nejznámější DoS útok pomocí ICMP zpráv je Smurf attack (Šmoulí útok). Při Smurf útoku útočník odesílá pakety ICMP ECHO, které mají jako zdrojovou adresu uvedenou broadcastovou adresu cílové stanice. Útočník očekává, že napadená stanice vygeneruje ICMP ECHO REPLY s B/C adresou svého segmentu a zahltí je tak svým provozem. Dalším DoS útokem s pomocí ICMP je Black

Nurse. Tento útok slouží k zahlcení firewallů pomocí generování zpráv ICMP DESTINATION UNREACHABLE (port unreachable). Útočník očekává, že se firewally zahlčí při zpracovávání informací o nedostupnosti.

3 Implementace protokolu

V rámci praktické části semestrální práce vznikl program, který přijímá a zobrazuje příchozí ICMP zprávy. Dále je možné pomocí tohoto programu generovat zprávy Echo, Timestamp a Information request.

3.1 Návrh aplikace

Aplikace se skládá ze dvou modulů `icmp` a `gui`. Modul `icmp` obsahuje definice, logiku přijímání a odesílání zpráv. Modul `gui` obsahuje uživatelské rozhraní.

3.1.1 Modul ICMP

Základem modulu ICMP je definice struktury ICMP záhlaví a následně objektu `ICMPMessage`, který poskytuje metody, pro nastavení parametrů záhlaví a uložení dat zprávy. Poslední funkcionalitou tohoto objektu je serializace nastavených dat do pole unifikovaných bytů (`uint8_t`).

Asynchronní příjem a odesílání zpráv je zaručen paralelním během dvou vláken (jedno pro odesílání, druhé pro příjem). Každé vlákno před spuštěním vytvoří instanci ICMP socketu pomocí volání `socket()`. Vlákno odesílání pracuje nad frontou zpráv dle návrhového vzoru producent konzument. Vlákno příjmu, neprodleně propisuje přijaté ICMP zprávy do GUI prostřednictvím rozhraní definovaném třídou `GUIInterface`. Celý proces běhu vláken je spravován pomocí řídicí třídy `Messenger`. Tato třída obsahuje vlajky pro běh jednotlivých vláken unikátní instanci fronty odesílaných zpráv.

Fornta odesílaných zpráv je instance objektu `MessageQueue`. Tento objekt je implementace blokující fronty. Pro synchronizaci jsou využity prostředky systémové knihovny C++ `<mutex>`, pro uložení dat je využita knihovna `<list>`.

3.1.2 Modul GUI

K implementaci uživatelského rozhraní byly použity prostředky knihovny *Qt* verze 5. Uživatelské rozhraní obsahuje dvě okna. Pro strukturální popis oken je použito XML pro definici UI ve frameworku *Qt*¹. Modul `gui` dále obsa-

¹Definice Qt UI XML - <http://doc.qt.io/archives/qt-4.8/designer-ui-file-format.html>

huje definici datového modelu pro zobrazované zprávy a definici objektů zobrazovaných zpráv. Posledním elementem tohoto modulu je definice rozhraní `GUIInterface`, které slouží ke komunikaci obou modulů.

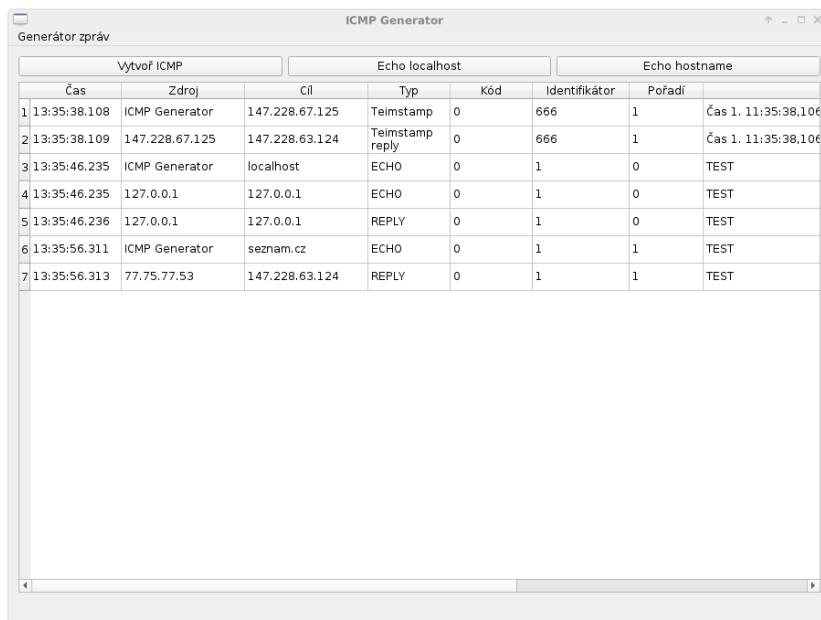
3.2 Překlad

Cílovou platformou programu je *OS GNU/Linux*, z důvodů závislosti na implementaci knihoven pro práci se sockety (`<sys/socket.h>`). Další závislostí je přítomnost frameworku *Qt* na sestavovací stanici.

K překladu je nutné využít mechanismů frameworku *Qt*. Nejprve je potřeba spustit program `qmake` ve složce s projektovým souborem `ICMPSniffer.pro`. Výstupem tohoto programu je soubor *Makefile*, který dodržuje závislosti modulů a respektuje nastavení stanice, na které byl spuštěn. Druhým krokem pro sestavení programu je spuštění `make` nad vygenerovaným *Makefile*. Výstupem tohoto volání je spustitelný program.

3.3 Ovládání

Pro běh programu je nutné mít nainstalovanou knihovnu *Qt* ve verzi 5 a vyšší. Program lze spustit z grafického rozhraní i z konzole. V případě spuštění z konzole bude mít uživatel k dispozici log událostí aplikace.



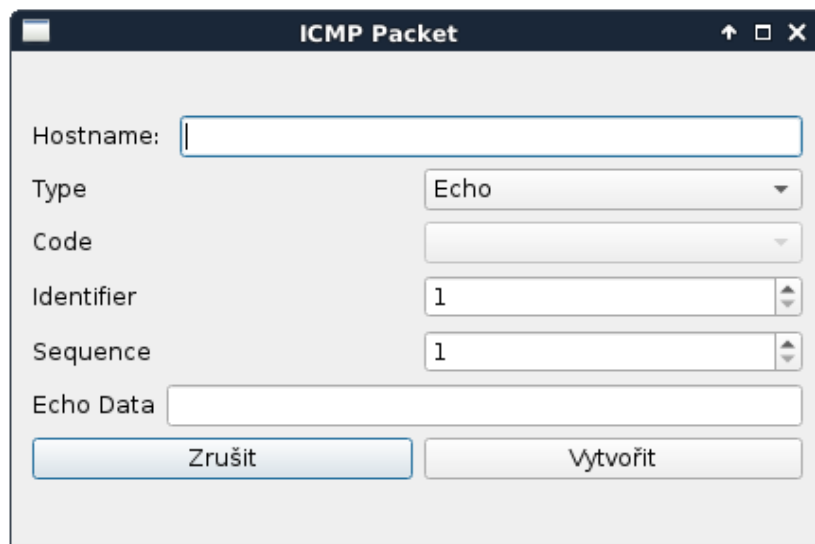
	Čas	Zdroj	Cíl	Typ	Kód	Identifikátor	Pořadí	Čas 1
1	13:35:38.108	ICMP Generator	147.228.67.125	Timestamp	0	666	1	Čas 1. 11:35:38.106
2	13:35:38.109	147.228.67.125	147.228.63.124	Timestamp reply	0	666	1	Čas 1. 11:35:38.106
3	13:35:46.235	ICMP Generator	localhost	ECHO	0	1	0	TEST
4	13:35:46.235	127.0.0.1	127.0.0.1	ECHO	0	1	0	TEST
5	13:35:46.236	127.0.0.1	127.0.0.1	REPLY	0	1	0	TEST
6	13:35:56.311	ICMP Generator	seznam.cz	ECHO	0	1	1	TEST
7	13:35:56.313	77.75.77.53	147.228.63.124	REPLY	0	1	1	TEST

Obrázek 1: Uživatelské rozhraní aplikace.

Spuštění programu může selhat v případě, že se nepovede otevřít ICMP socket. Návrátový kód programu je 1 v případě, že se nepodařilo otevřít socket pro odesílání zpráv a 3 v případě, že se nepodařilo otevřít a správně nastavit IP socket pro přijímání zpráv.

Po spuštění uživatelského rozhraní se zobrazí okno viz Obrázek 1. Uživatel má k dispozici tři akce. První je odeslat zprávu ECHO s předdefinovaným textem "TEST" na adresu localhost. Druhou je odeslání zprávy ECHO s předdefinovaným textem na adresu dle určení. Poslední možností je vytvořit vlastní ICMP zprávu.

Možnost vytvoření ICMP zprávy poskytuje uživateli možnost generovat tři typy zpráv: Echo, Timestamp a Information request. Uživatel má plnou kontrolu nad vyplněním parametrů jednotlivých zpráv.



The image shows a window titled "ICMP Packet" with a standard Windows-style title bar (minimize, maximize, close buttons). The window contains several input fields and two buttons. The fields are: "Hostname:" followed by a text input box; "Type" followed by a dropdown menu showing "Echo"; "Code" followed by a dropdown menu; "Identifier" followed by a text input box containing "1"; "Sequence" followed by a text input box containing "1"; and "Echo Data" followed by a text input box. At the bottom of the window are two buttons: "Zrušit" (Cancel) and "Vytvořit" (Create).

Obrázek 2: Rozhraní pro generování packetů.

4 Závěr

V rámci této práce byl vytvořen program pro přijímání, zobrazování a syntézu ICMP zpráv. Pro zobrazování zpráv bylo vytvořeno uživatelské rozhraní ve frameworku *Qt*.

Během testování aplikace byla ověřena funkčnost syntézy a příjmu zpráv ECHO proti standardní implementaci protokolu (např. zařízení na adrese seznam.cz). Pouze v laboratoři byla ověřena funkčnost zpráv Timestamp, proti standardní implementaci ICMP ve Windows 10 stanic v laboratoři. Zpráva Information request byla pouze zachycena a prozkoumána programem *Wireshark*, bohužel standardní Information response se nepodařilo zachytit.