



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ

DEPARTMENT OF INFORMATION SYSTEMS

MONITOROVÁNÍ SÍTĚ - ISAMON

NETWORK MONITORING - ISAMON

SEMESTRÁLNÍ PROJEKT

TERM PROJECT

AUTOR PRÁCE

AUTHOR

VLADIMÍR JEŘÁBEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. JAN PLUSKAL

BRNO 2017

Obsah

1	Úvod	2
2	Hledání zařízení v síti	3
2.1	Způsob vyhledávání v lokální síti	3
2.1.1	Implementace skenování zařízení v lokální síti	3
2.2	Způsob vyhledávání ve vzdálené síti	4
2.2.1	Implementace skenování zařízení ve vzdálené síti	4
3	Skenování portů u aktivních zařízení	5
3.1	Skenování TCP portů	5
3.1.1	Implementace skenování TCP portů	6
3.2	Skenování UDP portů	6
3.2.1	Implementace skenování UDP portů	6
4	Použití programu ISAMON	7
4.1	Příklady použití	7
4.2	Návratové kódy	8
	Literatura	9

Kapitola 1

Úvod

Nacházíme se v době, kdy už i naše pračky a ledničky jsou připojeny k internetu a tím pádem s těmito zařízeními můžeme komunikovat vzdáleně. Avšak s přibývajícím počtem zařízení v internetu vyvstává otázka, jak moc jsou tyto jednotlivé zařízení přístupné i pro nepovolané osoby.

S touto problematikou sovisí také to, zda náhodou nemáme v konkrétní síti nějaké zařízení navíc, o kterém bychom nevěděli.

Tento článek pojednává o programu ISAMON, který má sloužit pro monitorování sítě a pomoci při již zmiňovaných úskalích, které souvisí s rychle přibývajícím počtem zařízení v síti.

V tomto článku však také nalezenete podrobnější informace o technice skenování, kterou tento program používá pro zjišťování například všech hostů, kteří jsou připojeni k internetu a nachází se v daném rozpětí sítě (více o tomto tématu v kapitole 2) a také jaké techniky jsou použity pro skenování otevřených portů jak za pomoci protokolu TCP, tak i protokolu UDP (více v kapitole 3).

Dále je zde popsán způsob práce s tímto programem, a také možné návratové kódy (více v kapitole 4).

Kapitola 2

Hledání zařízení v síti

Jednou ze základních služeb, která tento program nabízí uživatelům, tak je skenování zadané sítě pro nalezení všech aktivních hostů, kteří jsou připojeni k internetu.

Všichni hosti jsou povinni implementovat ICMP Echo server funkci, dle RFC 1122. Tato funkce zajišťuje, že host korektně přijme ICMP Echo request a vygeneruje a pošle korespondující ICMP Echo replay.^[2]

Jelikož ne všechna zařízení respektují toto RFC, bylo nutné implementovat dva rozdílné způsoby vyhledávání aktivních klientů. Jeden způsob je použit pro vyhledávání aktivních klientů v lokální síti a druhý způsob pak při vyhledávání klientů ve vzdálené síti, kde vzdálená síť je definována, jako síť ve které není přítomné zařízení, na kterém běží program ISAMON.

2.1 Způsob vyhledávání v lokální síti

Jak již bylo popsáno výše, tak ne každé zařízení respektuje implementaci ICMP Echo server, avšak vyhledávání v lokální síti je usnadněno existencí takzvaného Address Resolution Protocol (ARP), který zajišťuje překlad IPv4 adresy na MAC adresu konkrétního zařízení. Jelikož bez implementace tohoto protokolu, by zařízení nebylo schopné poslat paket na jiné zařízení v síti, ba dokonce ani na směrovač, je tento protokol použit při skenování lokální sítě.

2.1.1 Implementace skenování zařízení v lokální síti

Skenování je založené na ARP. Na začátku skenování je vygenerován *ethernetový rámec*, který je specifikován v RFC 826^[3] a obsahuje konkrétní IPv4 adresu hledaného zařízení v síti. Pro odeslání ARP rámce bylo zapotřebí použít takzvaný *RAW socket*.

Pokud zařízení s konkrétní IPv4 adresou existuje v lokální síti, tak je vygenerována patřičná odpověď a program ISAMON tuto odpověď zachytí a přidá IPv4 adresu tohoto zařízení do instance třídy `IsamonLiveVect`, což je speciální vektor, který udržuje informace o aktivních zařízeních ve skenované síti.

Pokud však zařízení v síti neexistuje, program ISAMON žádnou odpověď nezachytí a po vypršení časovače přestane čekat na odpověď a považuje zařízení za neaktivní, tudíž toto zařízení není přidáno do instance třídy `IsamonLiveVect`, a nebude na něm prováděno případné další skenování.

Pozn: Při pokoušení se oskenovat sám sebe, není poslán ARP dotaz, protože by byl bezdůvodný (*tzv: Gratuitous*), přičemž by se nevygenerovala patřičná odpověď a tím pádem by se naše zařízení tvářilo jako by bylo neaktivní. V tomto případě se přistupuje k vlastnímu zařízení jako kdyby toto zařízení bylo ve vzdálené síti, avšak musí se naslouchat na takzvaném *loopback* zařízení.

2.2 Způsob vyhledávání ve vzdálené síti

Zde se nám nabízí jen několik málo různých přístupů k prohledávání vzdálené sítě a jedním z nich je použití protokolu **Internet Control Message Protocol (ICMP)**, avšak počítá se s tím, že v síti se některé pakety mohou ztratit (tzn. nedorazí k cíli), popřípadě mohou být zahozeny (vypršení TTL), takže tento způsob není vždy dokonalý.

2.2.1 Implementace skenování zařízení ve vzdálené síti

Při začátku skenování tohoto vzdáleného zařízení se na začátku vytvoří paket, dle popisu v **RFC 792**[5], tomuto paketu se přenastaví TTL na co nejvyšší hodnotu (konkrétně *255*), aby se předešlo zahazování paketů, a daný sestavený paket se pomocí specifického rozhraní pošle do sítě.

Pokud zařízení s konkrétní IPv4 adresou existuje a taktéž na tomto zařízení je korektně implementován **ICMP Echo server**, tak tak vygeneruje patřičnou odpověď (jedná se *ICMP Echo replay*) a tato odpověď je zachycena programem **ISAMON** a poté zpracována. Výsledkem zpracování je přidání adresy tohoto aktivního zařízení do instance třídy **IsamonLiveVect**.

Pokud však zařízení v síti neexistuje, nebo na tomto zařízení není korektně implementován **ICMP Echo server**, program **ISAMON** žádnou odpověď nezachytí a po vypršení časovače přestane čekat na odpověď a považuje zařízení za neaktivní, tudíž toto zařízení není přidáno do instance třídy **IsamonLiveVect**, a nebude na něm prováděno případné další skenování.

Kapitola 3

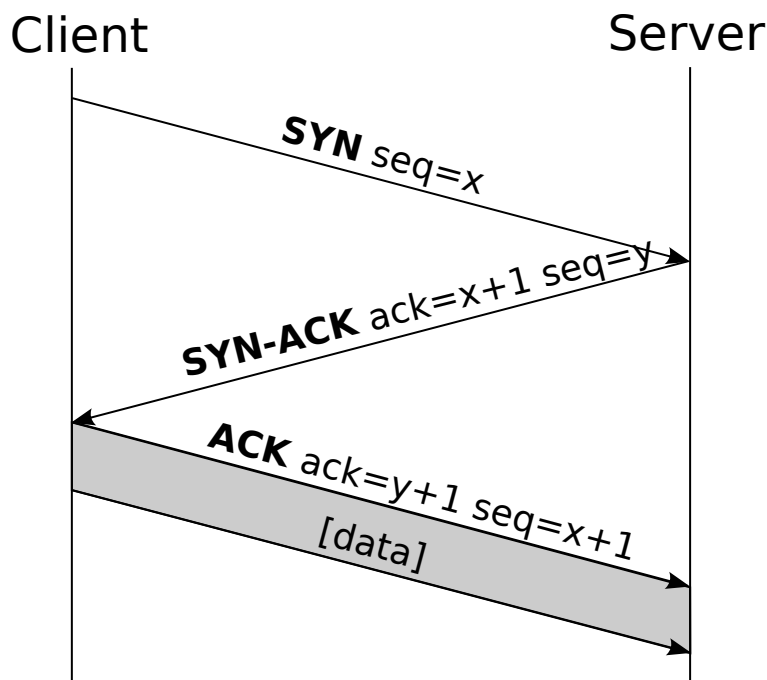
Skenování portů u aktivních zařízení

Jednou z dalších důležitých služeb, které jsou implementovány v programu ISAMON je možnost skenovat otevřené porty na jednotlivých aktivních zařízeních za pomoci rozdílných protokolů, jako je UDP [4] nebo TCP [1].

Jelikož přístupování k těmto protokolům je značně odlišné, jsou jednotlivé techniky skenování popsány ve zvláštních kapitolách.

3.1 Skenování TCP portů

Protokol TCP zajišťuje *spojení* dvou zařízení, která se nachází v internetové síti a tento protokol garantuje spolehlivé doručování zpráv oběma směry. Na začátku spojení dvou zařízení pomocí tohoto protokolu je vždy takzvaný *TCP handshake* (viz obrázek), který zajišťuje, že o sobě dané dvě zařízení ví a jsou připraveny na vzájemnou komunikaci.



Tohoto způsobu je využito i při skenování TCP portů. Na začátku program **ISAMON** zašle TCP paket s příznakem *SYN* na konkrétní port u aktivního skenovaného hosta a očekává, že přijde paket s nastavenými příznaky buď *SYN-ACK* (v tomto případě je port otevřený), a nebo přijde paket s nastaveným příznakem *RST* (v tomto případě jde o zavřený port).

3.1.1 Implemetace skenování TCP portů

Na začátku skenování pomocí TCP je otevřen *TCP socket*, kterým zajistíme, že odeslaný paket bude mít formát a strukturu pospanou v RFC 793 [1].

Dále je nastaveno internetové rozhraní, které bude sloužit pro komunikaci mezi zařízeními. Jelikož požadujeme, aby skenování bylo co nejrychlejší, tak je použit neblokující *connect*, který nám dovoluje nečekat na potvrzovací pakety a popřípadě zasílat na jiný port či jiného aktivního hosta další požadavky.

Příchozí pakety jsou odchyťávány za pomoci aplikačního rzhraní zvaného *libpcap*, které dokáže zpracovávat odchycené pakety. Pokud je tedy odchycen paket, který má nastavený příznak *SYN-ACK*, pak je tento port přidán jako otevřený ke konkrétnímu aktivnímu hostu.

3.2 Skenování UDP portů

Protokol UDP je, oproti TCP protokolu, *nespojovaný* a taktéž nezaručuje doručení paketů. V UDP protokolu neexistuje ani nic jako v TCP *TCP handshake*, který by nás informoval o povedení spojení, či ne. Jediné však co je generované a podel čeho lze usoudit, zda daný UDP port je otevřen, či zavřen, tak je informace ohledně nedostupnosti portu, kde tato informace je přenášena protokolem ICMP a nazývá se obecně *Port Unreachable*[2]. Jediný možný způsob tedy je, že jsou odeslána data na dané aktivní zařízení a konkrétní UDP port a čeká se jestli se navrátí *ICMP Port unreachable*, a nebo se nevrátí nic.

3.2.1 Implemetace skenování UDP portů

Na začátku skenování pomocí UDP je otevřen *UDP socket*.

Dále je nastaveno internetové rozhraní, které bude sloužit pro komunikaci mezi zařízeními. Poté se již odešle prázdná zpráva s délkou 0 na daný UDP port daného aktivního klienta.

Příchozí pakety jsou odchyťávány za pomoci aplikačního rzhraní zvaného *libpcap*, které dokáže zpracovávat odchycené pakety. Pokud je odchycen *ICMP* packet, jehož *type* a *code* odpovídá *Port unreachable*, pak je zřejmé, že tento port je zavřen.

Avšak pokud do vypršení časovače nedostaneme odpověď, pak si jen můžeme domyslet, že daný port je otevřen.

Pozn: Skenování UDP portů je velmi nepřesné a taktž velmi zdlouhavé. V realém nasazení né každý host má naimplementovaný *ICMP Destination port unreachable*, protože tato funkcionalita je pouze doporučená a né příkázaná. Dalším úskalím, které se oběvuje při skenovní UDP portů, tak je omezní počtu poslaných *ICMP* zpráv za jednotku času.

Kapitola 4

Použití programu ISAMON

Jak lze spouštět:

```
isamon [-h] [-i <interfc>] [-t] [-u] [-p <port>] [-w <ms>] -n <net_addr/mask>
```

-h --help ⇒ zobrazení nápovědy
-i --interface <interface> ⇒ rozhraní na kterém bude nástroj scanovat
-n --network <net_address/mask> ⇒ ip adresa sítě s maskou definující rozsah pro scanování
-t --tcp ⇒ použije TCP
-u --udp ⇒ použije UDP
-p --port <port> ⇒ specifikace scanovaného portu, pokud není zadáný, scanujte celý rozsah
-w --wait <ms> ⇒ jak dlouho se bude čekat na odpověď (výchozí nastavení jsou 2 sekundy)

4.1 Příklady použití

Pár příkladů:

```
$ isamon -h ⇒ vypíše se nápověda
```

```
$ isamon -i eth0 -n 192.168.1.0/24
```

⇒ provede se scanování sítě a zobrazí se aktivní klienti za použití rozhraní eth0

```
$ isamon -n 192.168.1.0/30
```

⇒ provede se scanování sítě a zobrazí se aktivní klienti za použití všech rozhraní

```
$ isamon -n 192.168.1.0/28 -t -p 22
```

⇒ provede se scanování sítě a zobrazí se aktivní klienti s otevřeným TCP portem 22 za použití všech rozhraní

```
$ isamon -n 192.168.1.0/30 -t -u -w 5
```

⇒ provede se scanování sítě a zobrazí se aktivní klienti a všechny otevřené TCP a UDP porty za použití všech rozhraní, pokud klient neodpoví do 5ms, daný port je považován za uzavřený

4.2 Návrátové kódy

Program ISAMON může skončit s těmito návratovými kódy:

- 0 : Všechno probělo v pořádku
- 1 : CHYBA - v parsování argumentů
- 2 : CHYBA - v nastavení adresy sítě či masky

- 5 : CHYBA - způsobená při získávání informací o rozhraních
- 6 : CHYBA - ve vytváření nebo odesílání ICMP packetu
- 7 : CHYBA - při vytváření TCP socketu
- 8 : CHYBA - při vytváření či odesílání UDP packetu

- 10 : CHYBA - při odchyťávání packetu (pcap chyba)

Literatura

- [1] *TRANSMISSION CONTROL PROTOCOL*. 1981, [Online; navštíveno 19.11.2017].
URL <https://tools.ietf.org/html/rfc793>
- [2] Braden, R.: *Requirements for Internet Hosts – Communication Layers*. Říjen 1989, [Online; navštíveno 19.11.2017].
URL <https://tools.ietf.org/html/rfc1122#page-42>
- [3] Plummer, D. C.: *An Ethernet Address Resolution Protocol*. Listopad 1982, [Online; navštíveno 19.11.2017].
URL <https://tools.ietf.org/html/rfc826>
- [4] Postel, J.: *User Datagram Protocol*. Srpen 1980, [Online; navštíveno 19.11.2017].
URL <https://tools.ietf.org/html/rfc768>
- [5] Postel, J.: *INTERNET CONTROL MESSAGE PROTOCOL*. 1981, [Online; navštíveno 19.11.2017].
URL <https://tools.ietf.org/html/rfc792>