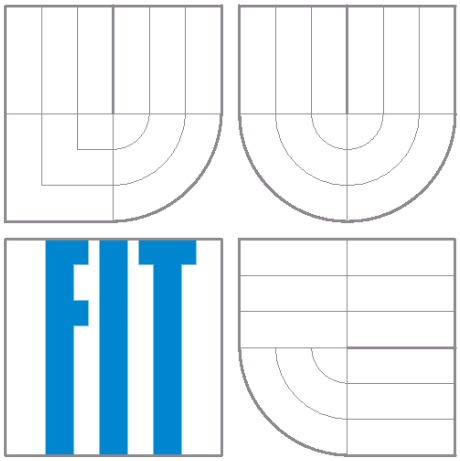

Dokumentace k projektu z předmětu ISA

Sledování sítě s využitím NetFlow v5

6. Prosince 2009

Petr Mrázek (xmraze03)

Zadání

Vytvořte jednoduchý exportér NetFlow v5 dat. Exportér bude sbírat veškerý provoz na síťovém rozhraní a vytvářet statistiky o probíhající TCP, UDP komunikaci - tocích. Tyto statistiky budou exportovány pomocí NetFlow v5 protokolu na externí kolektor. Jako referenční kolektor doporučuji využít NFDUMP tools.

Pozn.: Přenos mezi exportérem a kolektorem pomocí NetFlow v5 implementujte efektivně, tj. více záznamů je exportováno v jednom NetFlow paketu. Zároveň však nesmí docházet k uvíznutí záznamů určených pro export v exportéru na více než sekundu.

Programovací jazyk: C, BSD sockets, PCAP, C++, STL
Prerekvizity: root a libpcap-1.0.0

Rozbor požadavků

Ze zadání je zjevné, že program lze rozdělit na několik problémů. V první řadě je to samotné zachtávání paketů. Zde je k dispozici knihovna PCAP. Dále je potřeba zpracovat zachycené pakety a vytvořit z nich netflow toky. Požadavek na to, aby nedocházelo k uvíznutí na déle jak vteřinu přidává všemu podstatně odlišný rozměr.

Zachytávání packetů pomocí knihovny PCAP

Dokumentace viz 'man pcap'.

Po dlouhém testování jsem došel k několika závěrům:

1. Není dobrý nápad zachytávat celý paket, stačí hlavičky. Při velkém objemu komunikace může docházet ke ztrátě paketů.
2. 'timeout' u volání pcap_loop a pcap_dispatch nelze použít jako zdroj hodin, protože na některých operaćních systémech tyto funkce čekají na první paket.
3. Použití neblokujícího režimu a select() vede také k extrémní ztrátě paketů. Je možné, že to jde nějak ohnout, ale dokumentace ke knihovně PCAP je na to příliš chudá.

Je tedy zřejmé, že se musí použít blokující režim a jiný zdroj hodin. Časování tedy bude řešeno pomocí POSIX signálu SIGALRT a časovače vytvořeného pomocí setitimer(). Toto si vyžádá také maskování signálů během jejich zpracování.

Ukončení programu bude také přes signály – SIGTERM, SIGINT a SIGHUP. Při ukončení programu by se měly všechny aktivní toky exportovat. Ukončení smyčky pak proběhne pomocí pcap_breakloop(), kde může dojít k čekání na poslední paket (program se neukončí hned, ale až tento paket přijde).

Dále jsem se rozhodl, že zachytávání/časování a zpracování od sebe oddělím.

Protokol NetFlow v5

Dokumentace viz.: http://www.cisco.com/en/US/docs/ios/solutions_docs/netflow/nfwhite.html

Netflow v5 je proprietální protokol pro zasílání informací o 'tocích' na 'kolektor'. Je však velmi rozšířený. Tok je definován jako množina paketů, které prošly rozhraním v určitém časovém rozpětí v jednom směru a mají stejné klíčové parametry.

NetFlow v5 pakety se posílají přes UDP.

Zde jsou dostupné klíčové parametry:

- Zdrojová a cílová IP adresa (Ipv4)
- Zdrojový a cílový port
- IP Protokol (TCP/UDP/ICMP/...)
- IP TOS (Type Of Service)

Není podporováno ani Ipv6 ani jiný další protokol.

Struktura NetFlow v5 paketu je dobře popsána zde: http://netflow.caligare.com/netflow_v5.htm

Toky se exportují hlavně podle toho, kdy přišel první a poslední paket. Pro první platí tzv. aktivní timeout, pro poslední tzv. neaktivní timeout. Dalším kritériem může být podrobnější rozbor TCP komunikace nebo přetečení hodnot ve FLOW RECORD.

Úkolem je tedy vytvářet toky ze zachycených paketů a ty pak vkládat do NetFlow paketů, které se posílají na kolektor.

Popis implementace

Implementace první části (main.cpp)

Zachytávání paketů, funkce pro zpracování signálů a zpracování parametrů příkazové řádky lze nalézt v souboru *main.cpp.* Inicializace proběhne tak, že se zpracují parametry, nastaví hodiny pomocí *setitimer()* - a zpracování signálů, vytvoří objekt typu *nfv5* a inicializuje zachytávání. Následně program setrvává ve čtecím cyklu *pcap_loop*, který pak volá přes prostředníka metodu *nfv5::process()*.

K odesílání paketů pak dochází při příchodu SIGALRM signálu, který je spouštěn hodinami zhruba každou vteřinu (990ms). Volá se *nfv5::expire()* pro vyexportování expirovaných toků.

K ukončení programu dochází při zachycení SIGTERM, SIGHUP nebo SIGINT. Pak se ukončí chod hodin a volá se *nfv5::expireAll()* pro vyexportování všech toků a pcap_breakloop pro opuštění čtecího cyklu. V *main() se* pak uvolní prostředky.

Rozhraní objektu nfv5 (nfv5.h)

nfv5();	– konstruktor, předávají se mu zpracované vstupní parametry
process();	– zpracování zachyceného paketu
expire();	– export expirovaných toků
expireAll ();	– export všech toků

Struktury a použité datové typy (nfv5util.h)

nf5_hlavicka	– hlavička NetFlow v5 paketu.
nf5_flow_desc	– NetFlow v5 tok
nf5_packet	– struktura obsahující *nf5_hlavicka* a 30x *nf5_flow_desc*. Použita pro sesavení NetFlow paketu.
Flow	– obálka pro *nf5_flow_desc*, aby se s ním dalo lépe pracovat, definuje řadu operátorů

Implementace druhé části (nfv5.cpp)

Zde je implementováno rozhraní *nfv5*. Privátní čast (*nfv5::Private*) je skryta za ukazatelem a vyskytuje se pouze v *nfv5.cpp*.

Toky jsou ukládány do seznamu, nové se přidávají se na začátek a při při příchodu nového paketu se na začátek posunují. Hledání probíhá také od začátku.

Každý příchozí paket se v *process()* zpracuje na tok, který je následně vyhledán v seznamu a buď sloučen s již existujícím toku se stejnými parametry, nebo přidán. Toky s příliš velkou velikostí (3GB) jsou nuceně expirovány – je jim nastaven příznak a při dalším volání *expire()* jsou odeslány.

expire() exportuje všechny toky, které expirovaly (vypršení času, nucená expirace) a *expireAll()* exportuje všechny. Export probíhá tak, že se toky lehce poupraví (volání htons a htonl u některých hodnot), vloží do struktury NetFlow paketu a odešlou pomocí *sendto().*

Návod k použití

Program se spouští z terminálu a tak nemá žádné grafické rozhraní. Pro spuštění je nutné mít root práva.

Popis parametrů:

nf5exporter -h

Zobrazí popis parametrů

nf5exporter -n <interface>

Použije se specifikované rozhraní

nf5exporter -d <ip-address> -p <port>

Adresa kolektoru, povinná. Port je nepovinný a při jeho neuvedení se pužije 2055.

nf5exporter -i <inactive-timeout-ms>

Délka neaktivního timeoutu pro toky v milisekundách. Default je 15 vteřin.

nf5exporter -a <active-timeout-ms>

Délka aktivního timeoutu pro toky v milisekundách. Default je půl hodiny.

nf5exporter -o <output-file>

Při uvedení tohoto přepínače se bude ukládat záznam o expirovaných tocích do souboru.

nf5exporter -O

Při uvedení tohoto přepínače se bude záznam o expirovaných tocích vypisovat do standardního výstupu.

Příklad použití:

sudo nf5exporter -n eth0 -d 192.168.1.127 -p 9017 -o /var/log/nf/1.log

Bude se sledovat rozhraní eth0, exportované toky se posílají na kolektor který běží na stroji s adresou 192.168.1.127 na portu 9017 (je dobré se ujistit, že běží, nf5exporter to nekontroluje). Také se zaznamenávají do souboru /var/log/nf/1.log

Popis výstupu do souboru:

<YYYY-MM-DD> <HH:MM:SS.MS> <duration> <proto> <src-ip> <dst-ip> <src-port> <dst-port> <packets> <bytes>
kde:
<duration> délka trvání toku v sekundách (formát SS.MS)
<src-ip> zdrojová IP adresa toku
<dst-ip> cílová IP adresa toku
<proto> tcp/udp
<src-port> zdrojový port toku
<dst-port> cílový port toku
<packets> počet paketů toku
<bytes> počet bytů toku

Reference

https://wis.fit.vutbr.cz/FIT/st/course-sl.php.cs?id=406088&item=23657&cpa=1 (zadání ve WISu)

http://netflow.caligare.com/netflow_v5.htm (Popis struktury NetFlow v5 paketu)

http://www.freebsd.org/doc/en/books/developers-handbook/sockets-essential-functions.html

http://www.tcpdump.org/pcap3_man.html

http://www.cisco.com/en/US/docs/ios/solutions_docs/netflow/nfwhite.html

a man stránky dalších zmíněných příkazů.