# Využití monitorování toků v síťové bezpečnosti

## PV177 Laboratoř pokročilých síťových technologií

Mgr. Jan Vykopal

17. 3. 2010

Pakety a toky

Průzkum sítě

Detekce anomálií

Vizualizace

Holt-Wintersova metoda

Využití entropie

# Pasivní vs. aktivní monitoring sítě

Pasivní monitoring:

- Do sítě nijak aktivně nezasahujeme.
- Pasivně odposloucháváme síťový provoz, sledujeme statistiky skutečného provozu.
- Příklad: SNMP, NetFlow aj.

Aktivní monitoring:

- Do sítě posíláme testovací pakety a sledujeme reakci sítě.
- Příklad: ping, traceroute.

# Sběr a analýza paketů

- Tradiční IDS (např. Snort) provádí analýzu obsahu paketů.
- Už v gigabitových sítích není možné provádět tuto operaci v reálném čase bez podpory specializovaného hardware.
- Off-line analýza není možná: vysoké nároky na úložiště (velké objemy rychle přibývajících dat), neaktuálnost výsledku.
- Proto se zavádí určitý kompromis: sběr a následná analýza statistických informací o síťovém provozu.

# Síťové toky – definice

Definice z RFC 3954:

*A flow is defined as a unidirectional sequence of packets with some common properties that pass through a network device. These collected flows are exported to an external device, the NetFlow collector. Network flows are highly granular; for example, flow records include details such as IP addresses, packet and byte counts, timestamps, Type of Service (ToS), application ports, input and output interfaces, etc.*

- Toky poskytují informace o tom **kdo, s kým a jak dlouho komunikoval, kolik přenesl dat a jaký protokol použil**.
- Umožňují dlouhodobě sledovat síťový provoz v reálném čase.

# Síťové toky – pokračování


| Flow start | Duration | Proto | Src IP Addr:Port | | Dst IP Addr:Port | Flags | Packets | Bytes |
|---|---|---|---|---|---|---|---|---|
| 09:41:21.763 | 0.101 | TCP | 172.16.96.48:15094 | -> | 209.85.135.147:80 | .AP.SF | 4 | 715 |
| 09:41:21.893 | 0.031 | TCP | 209.85.135.147:80 | -> | 172.16.96.48:15094 | .AP.SF | 4 | 1594 |

# Síťové toky a agregace

- Toky představují střední úroveň agregace, na vysokorychlostních linkách ale i tak generují velký objem dat.
- Příklad: pětiminutový vzorek dat z mezinárodní 10GE linky do GÉANT2:
  - 578 944 960 bajtů
  - 6 163 012 paketů
  - 152 010 toků
  - 45 284 různých párů IP adres
- Počet toků závisí na parametrech sběru toků:
  - *inactive timeout* – ukončení toku
  - *active timeout* – průběžná data (vhodné pro dlouhotrvající toky)

# Dostupná řešení pro monitorování toků

Směrovače – CISCO, Juniper, Enterasys, . . .

- Zaneprázdněny směrováním, monitorování toků jako doplněk.
- Monitorování toků není implementované ve všech modelech.
- Fixní umístění, možný cíl útoků.
- Často nezbytné vzorkování, omezené pokročilé technologie.

# Dostupná řešení pro monitorování toků – pokračování

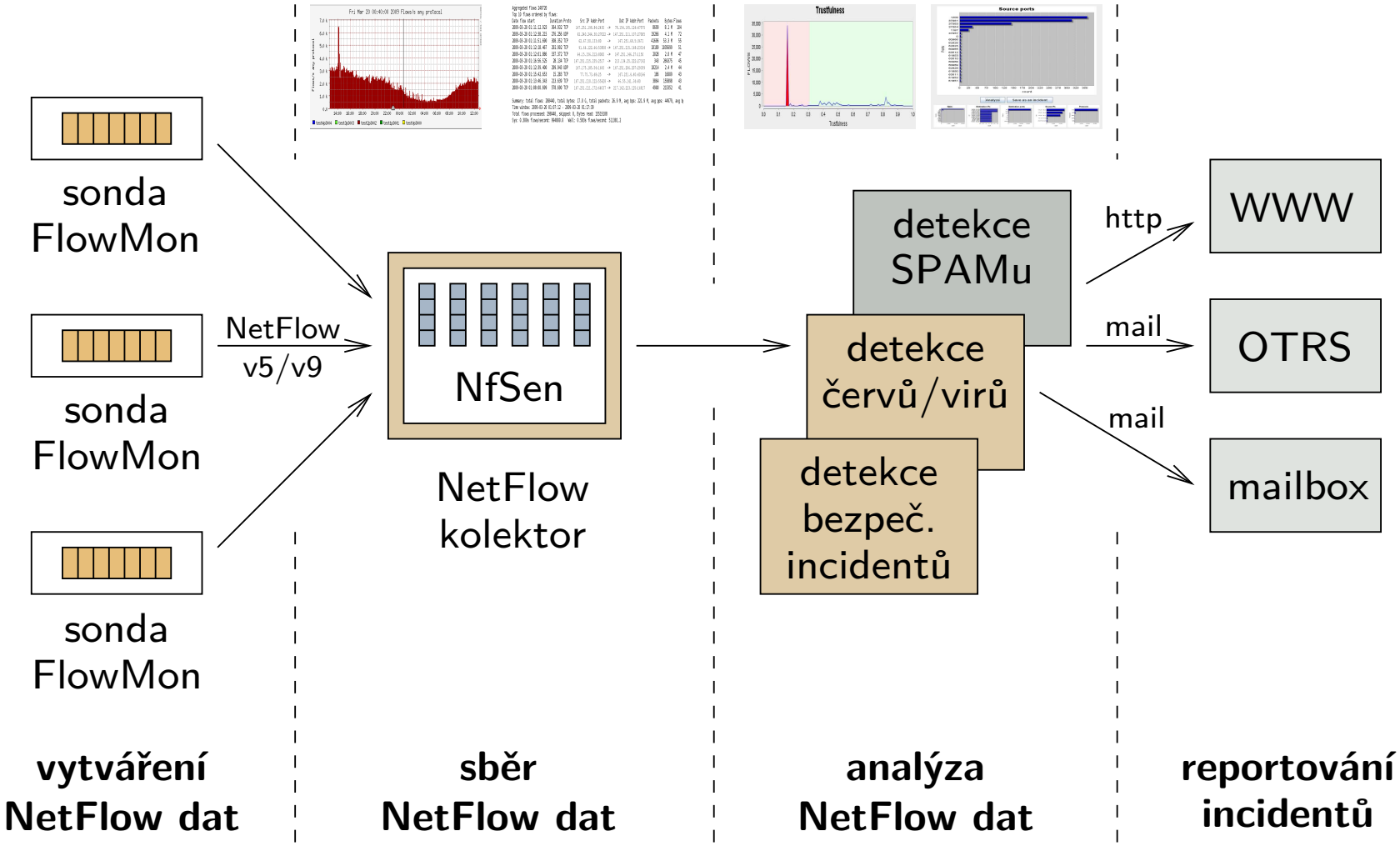
SW NetFlow Sondy – nProbe, fprobe, softflowd, . . .

- Založeno na běžném HW – PC a běžných síťových kartách.
- Limitovaný výkon (PCAP, PCI-X) a problémy stability.
- Vyžaduje expertní úpravy a nastavení měřicího systému.
- Zaplňují mezeru kde potřebujeme monitorovat a není čím.

# Hardwarově akcelerované sondy FlowMon

- Sběr statistických informací o tocích je taktéž náročný na použitý hardware $\Rightarrow$ vznik specializovaných HW sond.
- Základní výzkum proběhl v rámci aktivity Programovatelný hardware sdružení CESNET ve spolupráci s VUT a MU.
- Projekt EU FP6 GEANT2 $\Rightarrow$ vznik *GN2 Security Toolset*, který tvoří sonda *FlowMon* a kolektor *NfSen*.
- Technologický transfer do společnosti INVEA-TECH a.s.
- Dnes probíhá vývoj sondy *Flexible FlowMon* – rozšíření klasické pětice tvořící klíč toku o další, uživatelem definované položky.

# Architektura systému



# Shrnutí

- Pro bezpečnostní aplikace je výhodné trvalé pasivní monitorování.
- Nasazení SNMP či sběr Syslogu vyžaduje konfigurovat dotčené prvky.
- Analýza síťového provozu je pro uživatele transparentní.
- Prohledávání obsahu paketů je ve vysokorychlostních sítích nemožné.
- Používá se monitorování síťových toků (typicky na síťové až transportní vrstvě), které poskytuje kýženou agregaci a přitom zachovává informační hodnotu (narozdíl od SNMP).

# Zjemnění statistik SNMP

- Lze získat statistiky o počtu přenesených toků, paketů a bajtů – sumární, pro určité podsítě či dokonce jednotlivé stroje.
- Původní využití NetFlow bylo právě pro účtování (accounting).
- Tyto statistiky lze ale použít i u „objemově výrazných“ anomálií (např. detekce botnetu Chuck Norris).
- Stejně jako v ostatních aplikacích je klíčové umístění sondy v infrastruktuře.

# Kontrola reverzních DNS záznamů

- RFC 1912:

  *Every Internet-reachable host should have a name. The consequences of this are becoming more and more obvious. Many services available on the Internet will not talk to you if you aren't correctly registered in the DNS.*
- Absence reverzního záznamu může ukazovat na zapomenutý a tedy i potenciálně nebezpečný stroj.

# Kontrola reverzních DNS záznamů

- Předpoklad: trvale monitorujeme síťový provoz.[1]
- Zjistíme, jaké stroje komunikovaly v daném časovém okno.
- Vhodně využijeme agregaci.
- Máme seznam komunikujících strojů.
- Zavoláme překlad IP na jméno (např. *host*).
- Jsme hotovi? Všechno funguje bezvadně?
- Výsledky může ovlivnit firewall v cestě.
- Zajímá nás provoz reálných strojů, ne reakce firwallu na skenování.
  Jedním z možných řešení je omezení na TCP a filtr na TCP SYN pakety.

[1] Opět je klíčové kde.

# Detekce spamerů

- Poštu smějí odesílat jen vybrané relaye v síti.
- Firewall na hranici sítě blokuje veškerý provoz nepovolených relayí.
- Nakažený stroj je obvykle použit i pro šíření nevyžádané pošty.
- V popsané konfiguraci spam neprojde za hranici sítě . . .
- . . . ale také nemáme šanci zjistit špatnou konfiguraci firewallu.

# Detekce spamerů – výhodné využití dvou zdrojů dat

- Jedna sonda bude ve vnitřní síti, před hraničním firewallem.
- Druhá („nepovinná”) za firewallem.
- Detekce spočívá v „odečtení” provozu na konkrétním portu (TCP/25) uvnitř sítě od provozu za firewallem.
- Výsledkem rozdílu jsou neúspěšní spameři (ideálně prázdná množina :-)).

- Při detekci spammerů z vnějšku a otevřených relayí uvnitř nutno odlišit skutečné odeslání pošty od odmítnutí typu „já poštu neposílám, běž na a.b.c.d”.

# Detekce TCP SYN skenů

- Obecně nás zajímají spíše stroje v naší síti, pod naší správou (ty lze nějak „usměrnit”).
- Skenování portů je typickým projevem šíření červů, často je velmi agresivní (mnoho pokusů za krátký čas).
- Velmi jednoduchou a obecnou(!) metodou je sledování TCP provozu, konkrétně TCP SYN toků.
- Pokud stroj překročí v daném časovém okně nastavený počet pokusů, je označen za skenera.
- Pozor ale na falešné poplachy způsobené např. navázáním spojení s již vypnutých strojem.
- I když jde o jednoduchou medotu, v praxi je velmi účinná se zanedbatelným počtem falešných poplachů.

# Shrnutí

- Agregace provozu v podobě síťových toků je dobrým stavebním kamenem jednoduchých metod.
- Metody zpracovávající záznamy o tocích jsou v porovnání s inspekcí paketů velmi rychlé.
- I jednoduché metody (např. detekce TCP SYN skenů) jsou v praxi velmi užitečné.

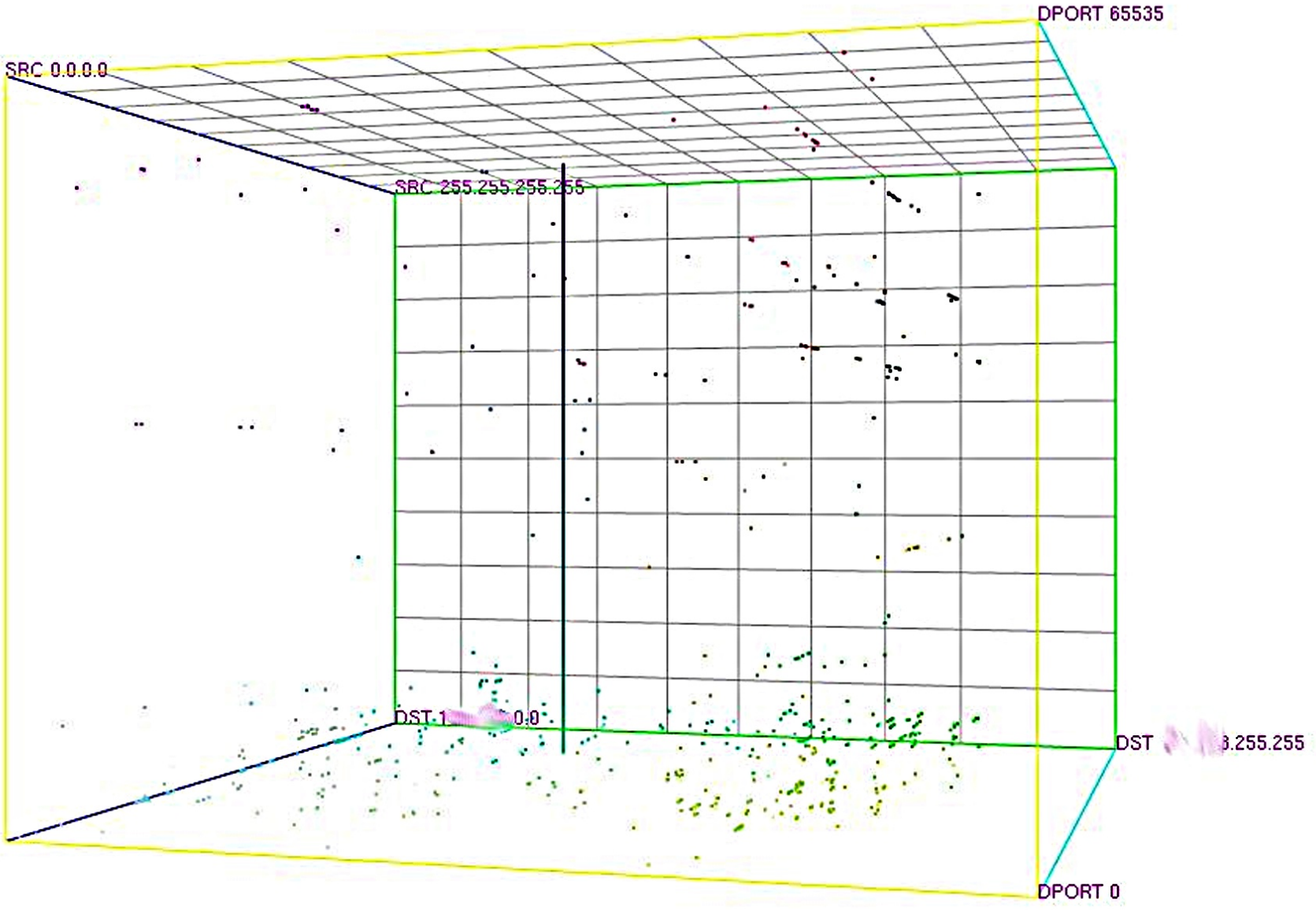
# Grafy

- Vizualizace je pro člověka stravitelnější než textové výpisy ve formě *co tok, to řádek*.
- 2D grafy – přirozená metoda vizualizace.
- Na osu x nanášíme většinou čas a na osu y sledovanou veličinu.
- Vhodně vybrané veličiny a podsítě mohou pomoci s odhalením anomálie pouhým okem.

# Využití stereoskopického vnímání

- „Převádí” vzory provozu na grafické vzory a útvary.
- *The Spinning Cube of Potential Doom*
  - osa x: lokální adresový prostor,
  - osa z: globalní adresový prostor,
  - osa y: čísla cílových portů.
  - Úspěšná TCP spojení bíle, neúspěšná v barvě duhy.

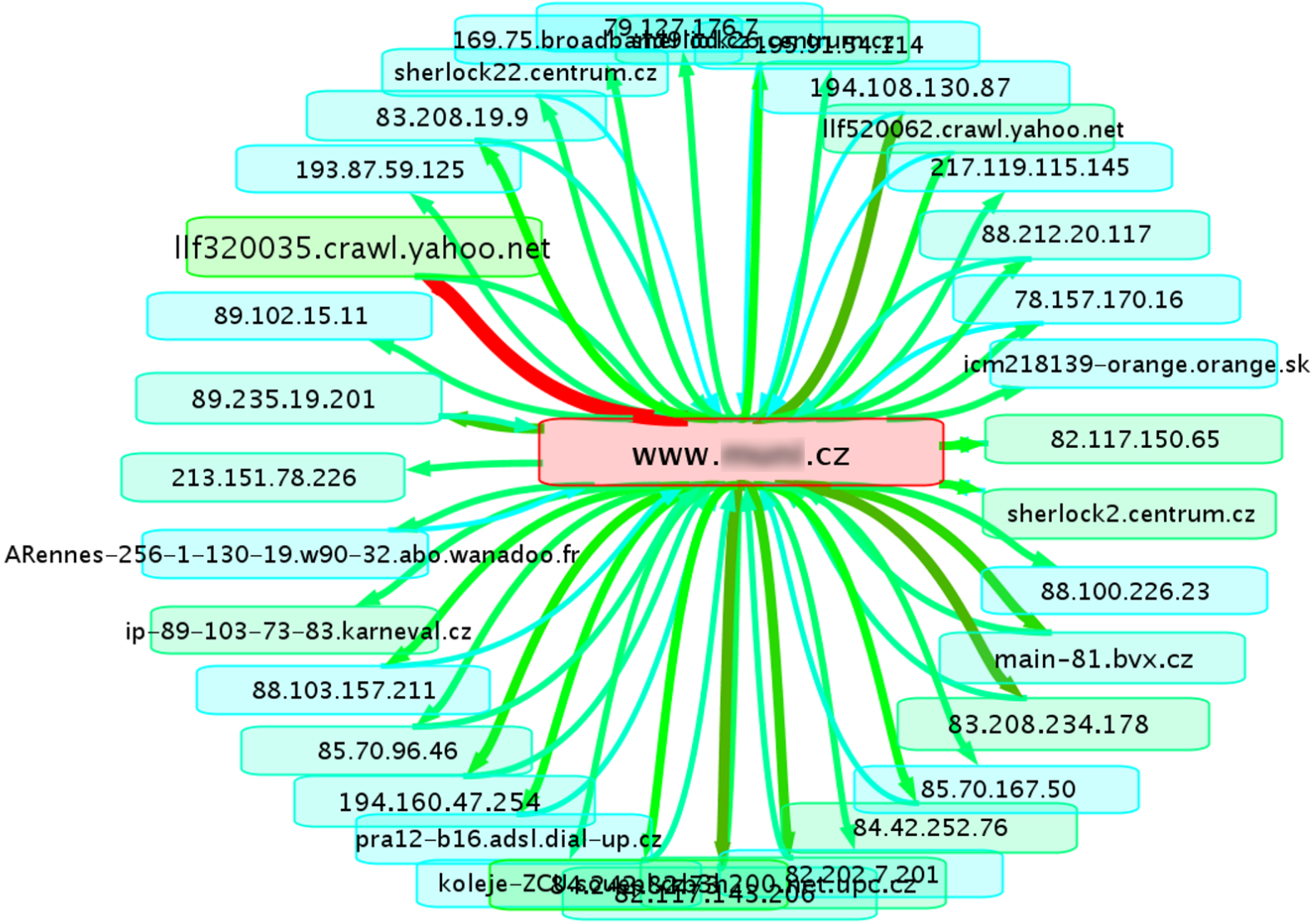
# Skenování portů v kostce



# Stroje a toky jako orientovaný graf

- Vrcholy grafu tvoří stroje (IP adresy).
- Hrany zobrazují jednotlivé toky nebo jejich agregace.
- Velikost, zabarvení. . . vrcholu/hrany odráží nějakou jeho/její charekteristiku (např. počet přenesených bajtů).
- Pohled na to, ***kdo*** *kolik čeho* ***kam*** *přenášel*.

# Orientovaný graf zobrazující síťový provoz



# Holt-Wintersova metoda: úvod

- V literatuře také jako *triple exponential smoothing*.
- Po mnoho let se používá pro **předpovědi vývoje *časové řady***.[2] Poprvé navržena v roce 1957 Holtem, v roce 1965 vylepšena Wintersem.
- Metoda stojí na předpokladu, že časová řada může být rozložena na tři komponenty: *základnu, lineární trend a sezónní trend*.
- Metoda předpokládá, vývoj těchto komponent v čase a postupně **upravuje jejich hodnoty podle historie**.

[2] Posloupnost pozorování jedné nebo více náhodných veličin uspořádaná v čase. Předpokládáme ekvidistantní časový interval.

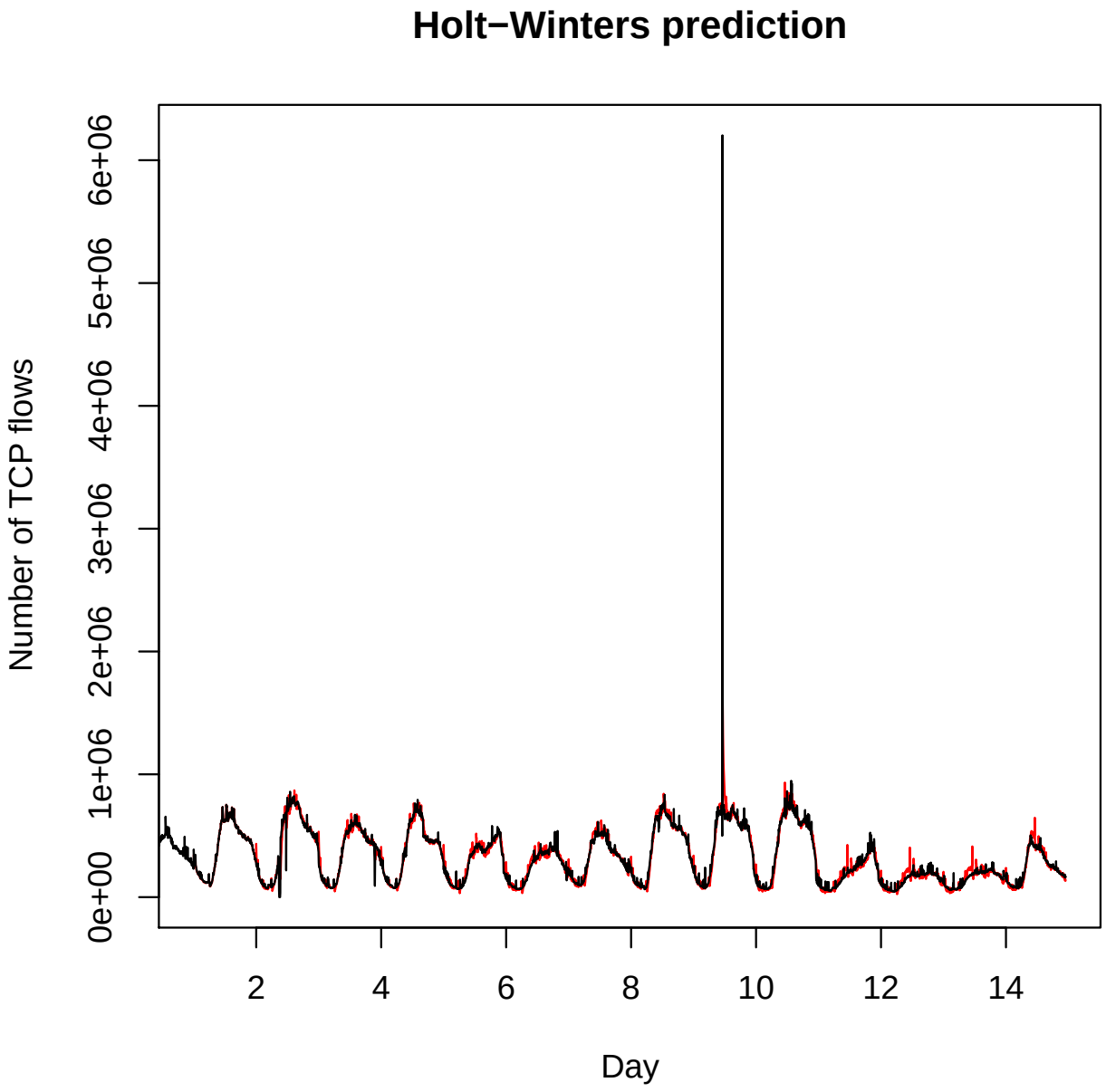
# Holt-Wintersova metoda a síťový provoz

- Metoda je vhodná i pro síťový provoz a toky.
- Časové řady mnohých služeb totiž vykazují toto chování (příklad):
  - **Trend**: postupné zvyšování požadavků na nějakou službu v čase.
  - **Sezónnost**: nejvíce požadavků se objevuje dopoledne, odpoledne méně a v noci obvykle minimum.
  - **Sezónní proměnlivost**: ve špičce je zaznamenána velká fluktuace počtu požadavků, kdežto v noci ne.
  - Postupný vývoj všech předcházejících složek v čase: vliv letního času na počty požadavků v průběhu dne.

# Holt-Wintersova metoda a síťová bezpečnost

- Cílem je automatizovat detekci anomálii *okem* (pohled na graf a následné nalezení špiček).
- Předpověd použijeme pro predikci (očekávaného) vývoje řady.
- Pokud se však aktuální hodnota (výrazně) liší od předpovídané, jde o anomálii; typicky i bezpečnostní.
- Viz příklad na následující slídě: počet TCP toků po dobu několika dnů (ekvidistantní časový interval je 5 minut).
- Pozor však na možnost „otrávení metody” ve fázi učení!

# Vizualizace předpovědi Holt-Wintersovou metodou



# Shrnutí

- **Holt-Wintersova metoda** (HW) se používá pro předpověď vývoje časové řady na základě historických dat a trendů.
- Statistiky o komunikaci strojů či používání síťových služeb lze chápat jako časovou řadu.
- Tyto řady vykazují jistý **trend** a **sezónnost**, která se navíc **mění v čase**. Proto lze použít HW.

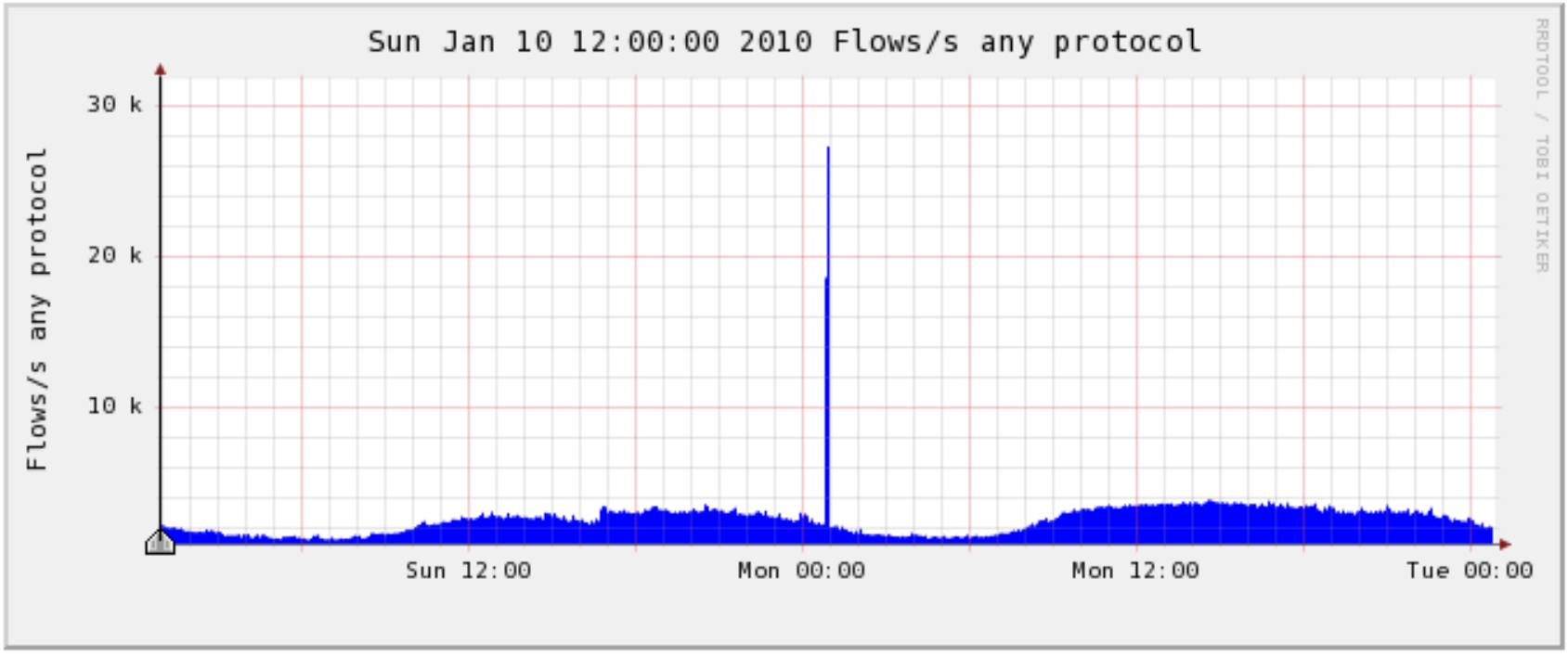
# Motivace

Je vše v pořádku? Zkuste zde najít nějaký útok:

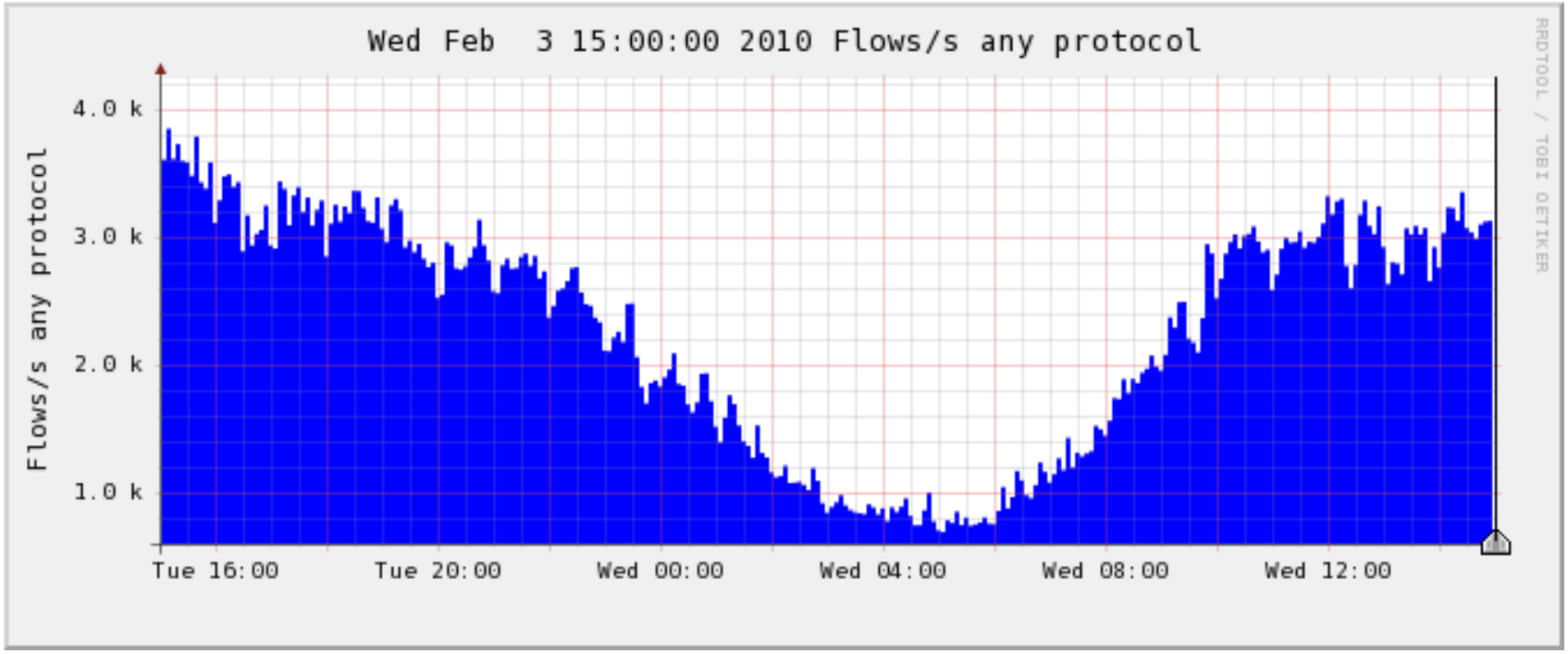
| Date flow start | Duration | Proto | Src IP Addr:Port | | Dst IP Addr:Port | Flags | Tos | Packets | Bytes | Flows |
|---|---|---|---|---|---|---|---|---|---|---|
| 2010-01-10 00:49:29.111 | 298.128 | TCP | 158.218.210.178:61158 | -> | 144.13.69.204:55501 | .AP.S. | 0 | 27 | 2829 | 1 |
| 2010-01-10 00:49:29.150 | 298.309 | TCP | 144.13.69.204:55501 | -> | 158.218.210.178:61158 | .AP.S. | 0 | 28 | 3076 | 1 |
| 2010-01-10 00:49:30.769 | 296.201 | TCP | 144.13.198.112:2208 | -> | 64.250.152.124:80 | .AP..F | 0 | 18 | 3953 | 1 |
| 2010-01-10 00:49:31.483 | 270.721 | TCP | 144.13.198.201:3888 | -> | 64.250.152.125:443 | .AP..F | 0 | 17 | 3687 | 1 |
| 2010-01-10 00:49:32.156 | 293.013 | TCP | 83.26.2.151:55357 | -> | 144.13.44.112:51132 | .AP.S. | 0 | 136 | 8963 | 1 |
| 2010-01-10 00:49:35.209 | 276.796 | TCP | 144.13.69.204:54176 | -> | 108.68.248.66:43601 | .AP.S. | 0 | 24 | 3073 | 1 |
| 2010-01-10 00:49:37.033 | 290.869 | UDP | 87.95.51.145:46589 | -> | 144.13.32.140:123 | ..... | 0 | 10 | 760 | 1 |
| 2010-01-10 00:49:37.588 | 266.636 | UDP | 37.96.225.23:59722 | -> | 144.13.217.99:39573 | ..... | 0 | 8651 | 477776 | 1 |
| 2010-01-10 00:49:38.019 | 299.592 | TCP | 144.13.217.61:61463 | -> | 85.208.45.56:10273 | .AP.S. | 0 | 52 | 5899 | 1 |
| 2010-01-10 00:49:38.038 | 299.769 | TCP | 85.208.45.56:10273 | -> | 144.13.217.61:61463 | .AP.S. | 0 | 75 | 7113 | 1 |

a dalších **614 264** řádků.

# A teď?


Sun Jan 10 12:00:00 2010 Flows/s any protocol
Flows/s any protocol
30 k
20 k
10 k
Sun 12:00
Mon 00:00
Mon 12:00
Tue 00:00
RRDTOOL / TOBI OETIKER

# Někdy to ale není tak snadné:



# Entropie

Půjdeme na to vědecky...

- Nechť $X = \{n_i,\ i = 1, \ldots, N\}$, kde hodnota $i$ nastává $n_i$ krát v tomto vzorku. $N$ je počet různých hodnot.
- Entropie vzorku (*sample entropy*) je pak definována takto:

$$H(X) = -\sum_{i=1}^{N} (\frac{n_i}{S}) \log_2(\frac{n_i}{S}),\ S = \sum_{i=1}^{N} n_i$$

- $S$ je celkový počet pozorování.
- Hodnota entropie leží v intervalu $[0, log_2 N]$.
- Entropie je rovna nule, právě tehdy když jsou všechna pozorování stejná.
- Entropie je $log_2 N$, právě když jsou četnosti různé, tj. $n_1 = n_2 = \cdots = n_N$
- *Relativní* entropie: $h(X) = \frac{H(X)}{H_{max}} = \frac{H(X)}{log_2 N}$

# Příklad entropie

# Detekce červů a anomálií pomocí entropie

- Real-time **analýza velkého množství dat není možná** (tj. ve vysokorychlostních sítích).
- Často ani nevíme, **co** detekovat.
- Předpokládáme, že provoz generovaný červy (typicky skenování) je *odlišný* od normálního provozu.
- Jak na to? A nepočítat přitom „ošklivé” vzorečky?
- Využijeme známé kompresní algoritmy. Proč?
- Zkomprimujeme sekvenci. Velikost tohoto objektu je úměrná entropii původní sekvence!
- I když nejde o přesný výpočet entropie (ale spíše odhad), dostačuje to. Implementaci metody jako plugin do kolektoru NfSen nabízím jako BP/DP.

# Šíření červa Blaster

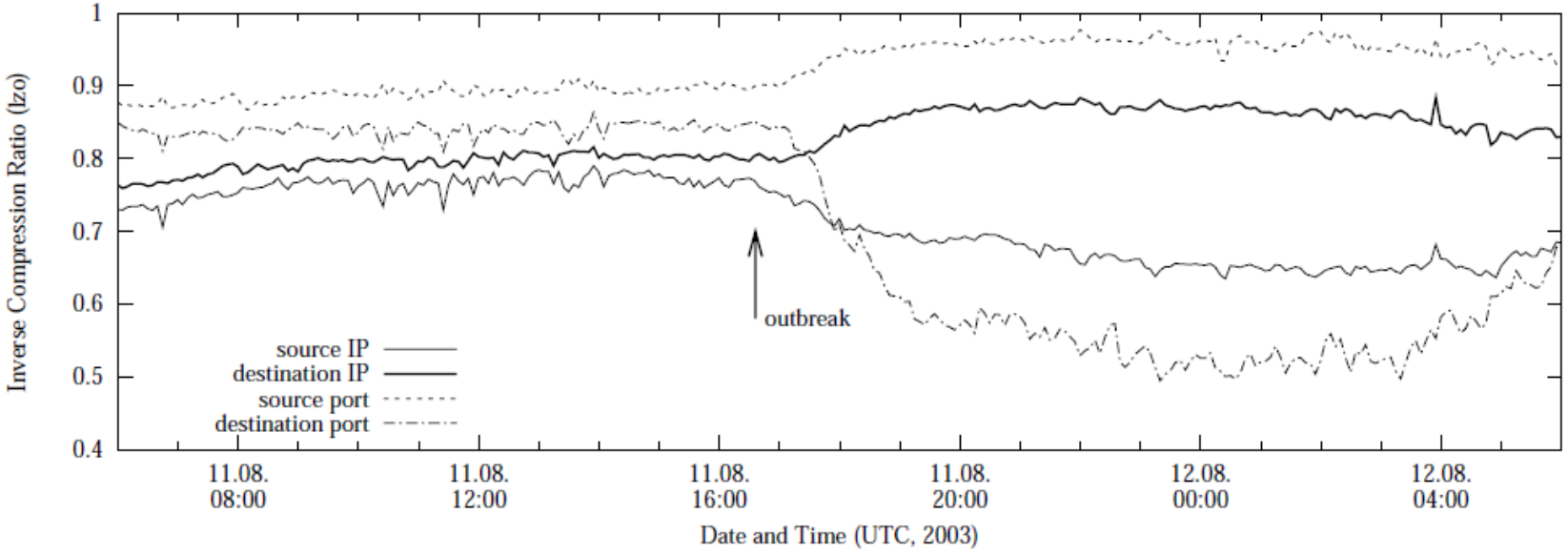



**Figure 1. Blaster - TCP address parameter compressibility**

Zdroj: *Entropy Based Worm and Anomaly Detection in Fast IP Networks*:
`http://www.tik.ee.ethz.ch/~ddosvax/publications/papers/wetice05_entropy.pdf`

# Šíření červa Witty

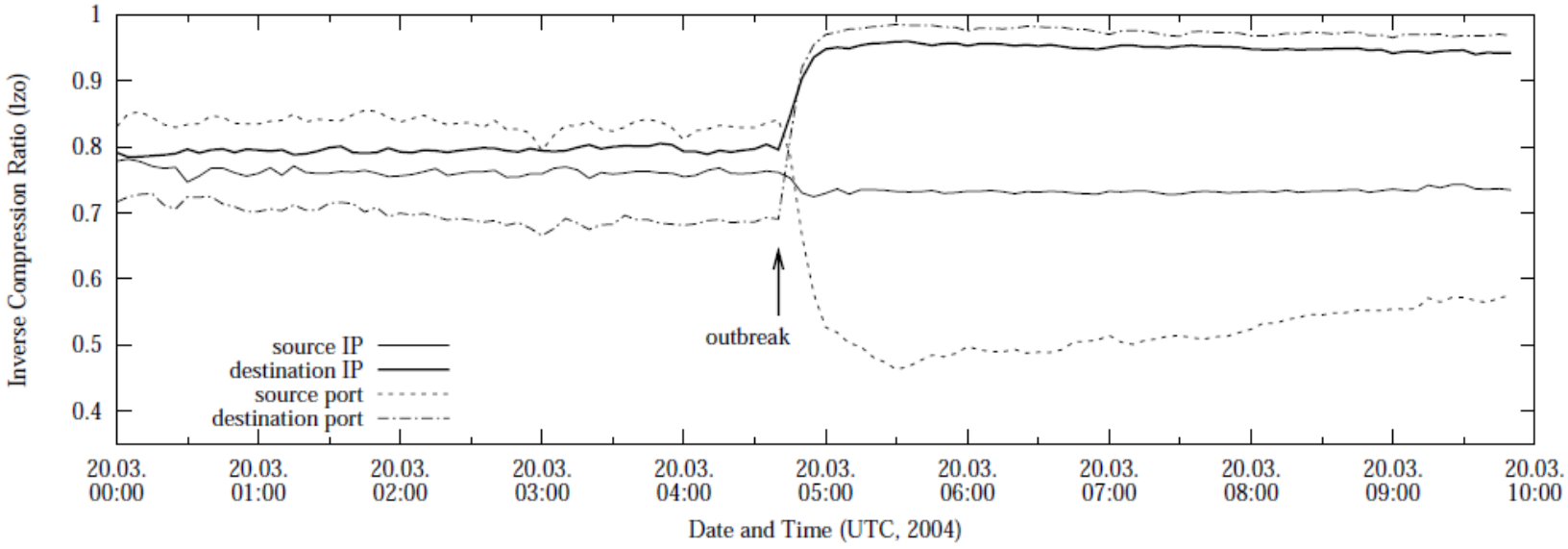



**Figure 2. Witty - UDP address parameter compressibility**

Zdroj: *Entropy Based Worm and Anomaly Detection in Fast IP Networks*:
`http://www.tik.ee.ethz.ch/~ddosvax/publications/papers/wetice05_entropy.pdf`

# Závěr

- Detekce průniků na úrovni sítě je transparentní pro uživatele a výhodné pro správce.
- Když nemáme přístup ke koncovým sítím, je to zároveň jediná možnost.
- Monitorování toků má široké uplatnění v síťové bezpečnosti (výborně škáluje).
- Jednoduché metody detekce anomálií jsou často velmi účinné (**K**eep **I**t **S**imple and **S**tupid).