

Zadání cvičení 02 – PB156cv

Petr Holub

1 Provedení experimentů

Doporučení: nechte si dostatek času na měření propustnosti: nabrání dostatečného množství dat zabere cca 45 minut čistého času měření. Pokud si tuto část zautomatizujete, tak můžete využít času k analýze provozu příp. přípravě protokolu.

1.1 Zachytávání síťového provozu

- Umlčet rozhraní enp0s31f6
 - sudo -s
 - service avahi-daemon stop
 - service network-manager stop
 - echo 1 >/proc/sys/net/ipv6/conf/enp0s31f6/disable_ipv6
 - ifconfig enp0s31f6 0
 - ifconfig enp0s31f6 down
 - ifconfig enp0s31f6 up
- Zachyťte minimálně 100 rámců, doporučeno více.
 - tcpdump -i enp0s31f6 -w mydump.pcap
- Stáhněte si získaný soubor.

1.2 Měření propustnosti sítě

Sběr dat v této části cvičení proveďte ve dvojicích (tj. každá dvojice bude sdílet výsledný soubor dat), ale vyhodnocení proveďte individuálně.

- Připojte dva notebooky k 100Mbps přepínači
- Nastavte pevně danou IPv4 adresu na ethernetovém rozhraní
 - service network-manager stop
 - ifconfig
 - zjistěte si jméno rozhraní
 - ifconfig rozhrani adresa/24
 - příčemž použijete privátní adresy z rozsahu 192.168.0.0/16 a domluvíte se mezi sebou na jednom přepínači, abyste si nepřidělili duplicitní adresy
- Na jednom z počítačů nainstalujte nuttcp server
 - nuttcp -S
- Na druhém spusťte měření propustnosti UDP v závislosti na velikosti paketů pro velikosti bufferu 16, 32, 64, 128, 256, 512, 1024, 1500 v obou směrech (měření v opačném směru zajišťuje přepínač -r):
 - nuttcp -u -l delka -R100M -i -T60 cil_ip
 - nuttcp -u -l delka -R100M -i -r -T60 cil_ip

Všimněte si, co se stalo pro velikosti < 32 a > 1448 a v protokolu vysvětlete.

- Zvažte použití automatizace shellovým skriptováním, např.:


```
– for delka in 16 32 64 128 256 512 1024 1500; do \
  nuttcp -u -l ${delka} -R100M -i -T60 cil_ip >nuttcp-${delka}.log; done
```
- Měření celkem 3× zopakujte pro ověření stability výsledků. Zamyslete se, co byste měli správně provést, abyste provedli nezávislá měření (m.j. systém uvedli do výchozího stavu).
- Stáhněte si dosažené výsledky.

2 Analýza výsledků a protokol

2.1 Zachytávání síťového provozu

- Charakterizujte typy provozu, které se vám podařilo zachytit – jaké protokoly vidíte a k čemu se používají.
- Vyberte 2 pakety různých typů (z různých toků) a analyzujte co nejpodrobněji hlavičky: co jste schopni z nich vyčíst.
- Do odevzdáárny nahrajte také výsledný PCAP soubor.

2.2 Měření propustnosti sítě

- Výsledkem měření propustností jsou časové řady: 60 s běhy jsou rozděleny na 1 s intervaly. Data z logu můžete extrahovat např. takto:


```
for i in *.log; do cat $i | perl -pe 's/^\.*s+(\d+\.\d+)\sMbps\s+.*$/\1/' | \
  head -n -2 >${i%.log}.values; done
```
- Načtěte časové řady do R, např.


```
nut <- scan("run1/nuttcp-128.values", what=numeric())
```
- *Hypotéza: časové řady mají normální rozdělení.* Lze nebo nelze tuto hypotézu vyvrátit pomocí Shapiro-Wilkova testu normality?


```
shapiro.test(nut)
```

 Srovnajte vizuálně s histogramem a Q-Q diagramem:


```
hist(nut)
qqnorm(nut); qqline(nut)
```
- Lze-li tuto hypotézu vyvrátit, dokážete ji vyvrátit i v případě, pokud se do měření nezahrnou hodnoty, které byly naměřeny při současně registrovaných výpadech – tedy hodnoty zatížené hrubou chybou (outliers)? Jde-li pouze o první hodnotu (častý případ – ale ověřte!), můžete jednoduše použít výběr `nut[2:60]`.
- *Hypotéza: propustnost síťové karty a sítě nezávisí na velikosti paketů.* Lze tuto hypotézu vyvrátit nebo nelze? Zdůvodněte. Pokud jsou rozdělení přibližně normální, využijte průměrné hodnoty měření pro jednotlivé velikosti paketů s odhadnutou chybou měření při zvolené hranici intervalu spolehlivosti (např. 95%). V opačném případě použijte medián s uvedením rozsahu 1.–3. kvartilu. Příklad v R pro normální rozdělení:


```
library("Rmisc")      – příp. install.packages("Rmisc"), pokud je třeba instalace
CI(nut, ci=.95)
```

 a pro jiné než normální rozdělení


```
summary(nut)
quantile(nut, prob = seq(0, 1, length = 21))
```
- *Hypotéza: opakovaná měření poskytují výsledky vzorkované ze stejného rozdělení.* Lze tuto hypotézu vyvrátit nebo nelze? Zdůvodněte. Můžete využít např. Anderson-Darling k-vzorkového testu.


```
library("kSamples")
nut <- scan("run1/nuttcp-forth-128.values", what=numeric())
nut2 <- scan("run2/nuttcp-forth-128.values", what=numeric())
ad.test(nut,nut2)
```
- *Všechny prezentované výsledky musí být řádně zaokrouhleny a musí být uvedeny jednotky měření.*