SNMP Trapとの格闘の日々



2018/12/21
サイバートラスト株式会社
OSS技術本部応用技術部
阿部　勝久

はじめに（自己紹介）

cybertrust

- 阿部　勝久（あべ　かつひさ）
- サイバートラスト株式会社 (CTJ) OSS技術本部応用技術部所属
- 職種 ソフトウェアエンジニア
  - net-snmpのIPv6 transport関連のpatchを書いたことがある過去
  - CTJに入る前は，Android/iPhoneアプリの開発に従事
  - 運用管理に直接携わっていないです…
- 2018千葉アクアマリンマラソン DNF (6時間+数十秒) 2019の東京マラソン出走予定
- 出没地域







本日のテーマ



お客様

あなたのところでは，どのくらいの
SNMP Trapさばけるの？

前提 - SNMPTrap送信ツール

cybertrust

- net-snmpのsnmptrapコマンド
  - xargs コマンドを使用して並列実行
  - 350 trap/秒ぐらいしか出せず
- tcpreplay
  - 200000 trap/秒 はいけた
  - 同じ内容のTrapしか送信できない
- 自作Pythonスクリプト
  - pysnmp + asyncio
    http://snmplabs.com/pysnmp/examples/hlapi/asyncio/agent/ntforg/common-notifications.html
  - ProcessPoolを用いて並列実行: 950 trap/秒
  - 送信するtrapのvarbindをカスタマイズできる（今後の測定で重要な役割）

自環境

- 使用マシン: マウスコンピューター LM-iHS320S2-SH2 https://www.mouse-jp.co.jp/desktop/lms/lm-ihs320s2-sh2_spec.html
- インテル® Core™ i7-7700, インテル® H110 チップセット, 32GB メモリ, 480GB SSD (6Gbps対応)
- 使用OS: Ubuntu 18.04.1 LTS, Linux kernel version 4.15.0-42-generic
- snmptrap version: NET-SNMP version 5.7.3
- tcpreplay version: 4.2.6
- python version: 3.6.7
- pysnmp version: 4.4.6

UDP Packet Dropの原因

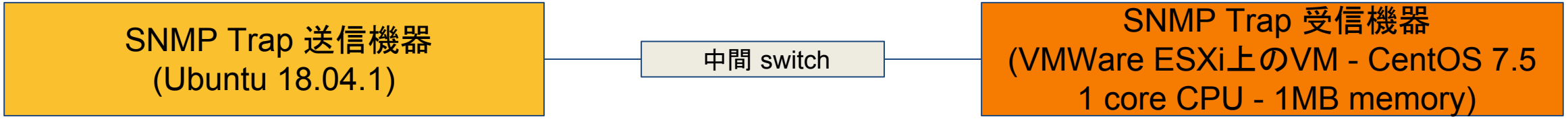
cybertrust

- Julia Evans - Why do UDP packets get dropped? https://jvns.ca/blog/2016/08/24/find-out-where-youre-dropping-packets/
  - “EVERYTHING IS BUFFERS”
    - OS socket send buffer, receive buffer
    - 中間機器（ルータ，スイッチ）バッファ，ネットワークカードバッファ…

測定してみよう - その1

cybertrust

- 目的 どこでPacket Dropが発生しているのかを確認する
  - 送信側か，中間機器か，受信機器か？
- 手法
  - 受信Trap数と送信Trap数を比較する
  - このとき受信機器の統計情報（UdpRcvbufErrors）の値もみてみる
- 実験環境



測定結果 - その1

250,000 Trapを約5秒で送信してみたところ…

```
# wc -l /var/log/snmptt/snmptt.log
202042
# nstat -nz | grep UdpRcvbufErrors
UdpRcvbufErrors  47958     0.0
# ethtool -S <interface>
…
     drv dropped rx total: 0
```

#(SNMPTrap送信数 - SNMPTrap受信数)
= #UdpRcvbufErrors

SNMP Trap のDropは受信側のBufferで発生している
（中間機器で発生しても対処できないので一安心）

測定してみよう - その2

cybertrust

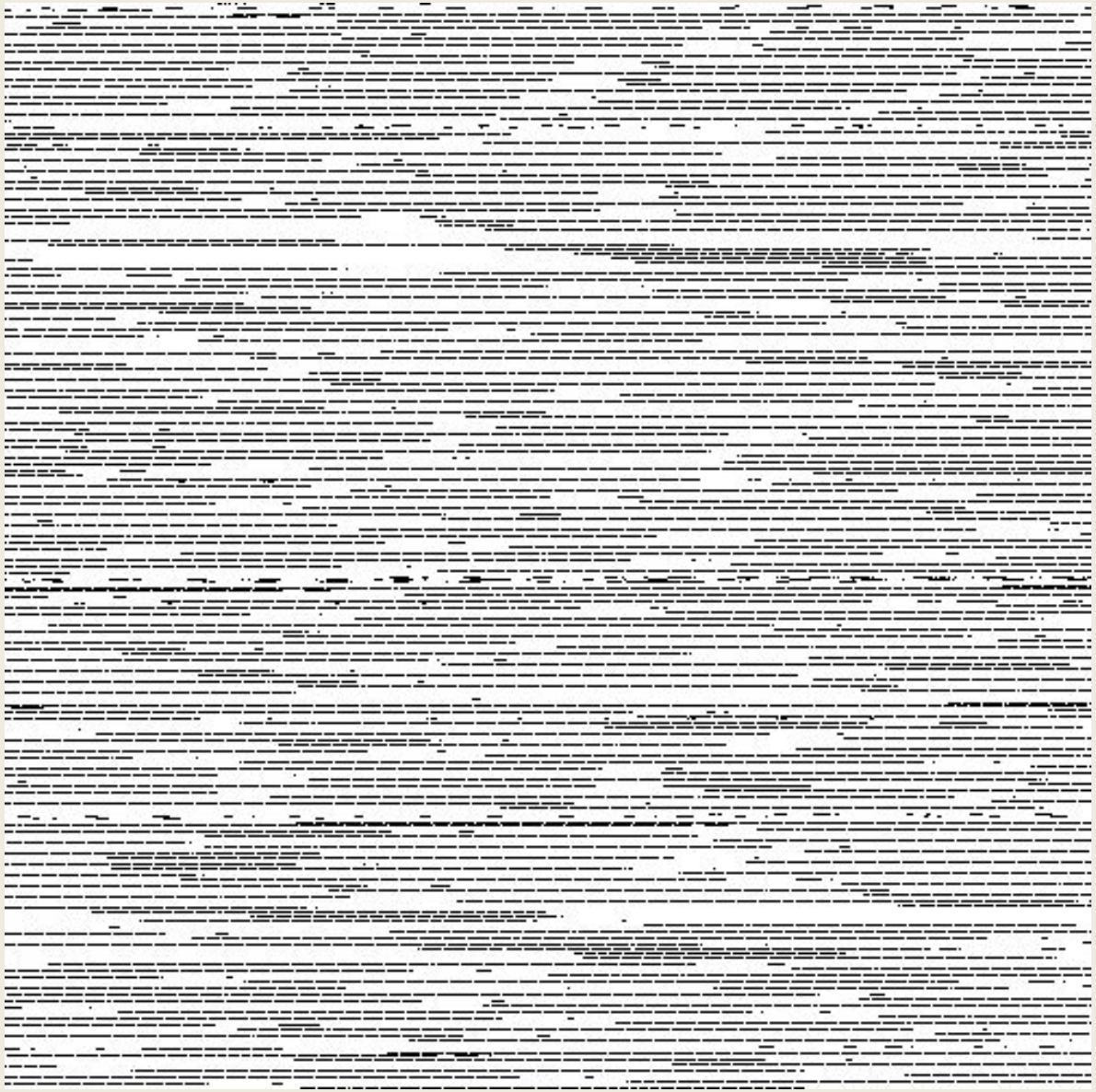
- Trapの取りこぼし状況を可視化してみる
  - 連番をvarbindに含めてtrapを送信

```
$ snmptrap -v2c -c public <<receiver>> \
 NET-SNMP-EXAMPLES-MIB::netSnmpExampleHeartbeatNotification \
 netSnmpExampleHertbeatRate i <<seq_num>>
```

  - snmptt.log で欠番している番号を抽出してみる

測定結果 - その2

cybertrust



測定してみよう - その3

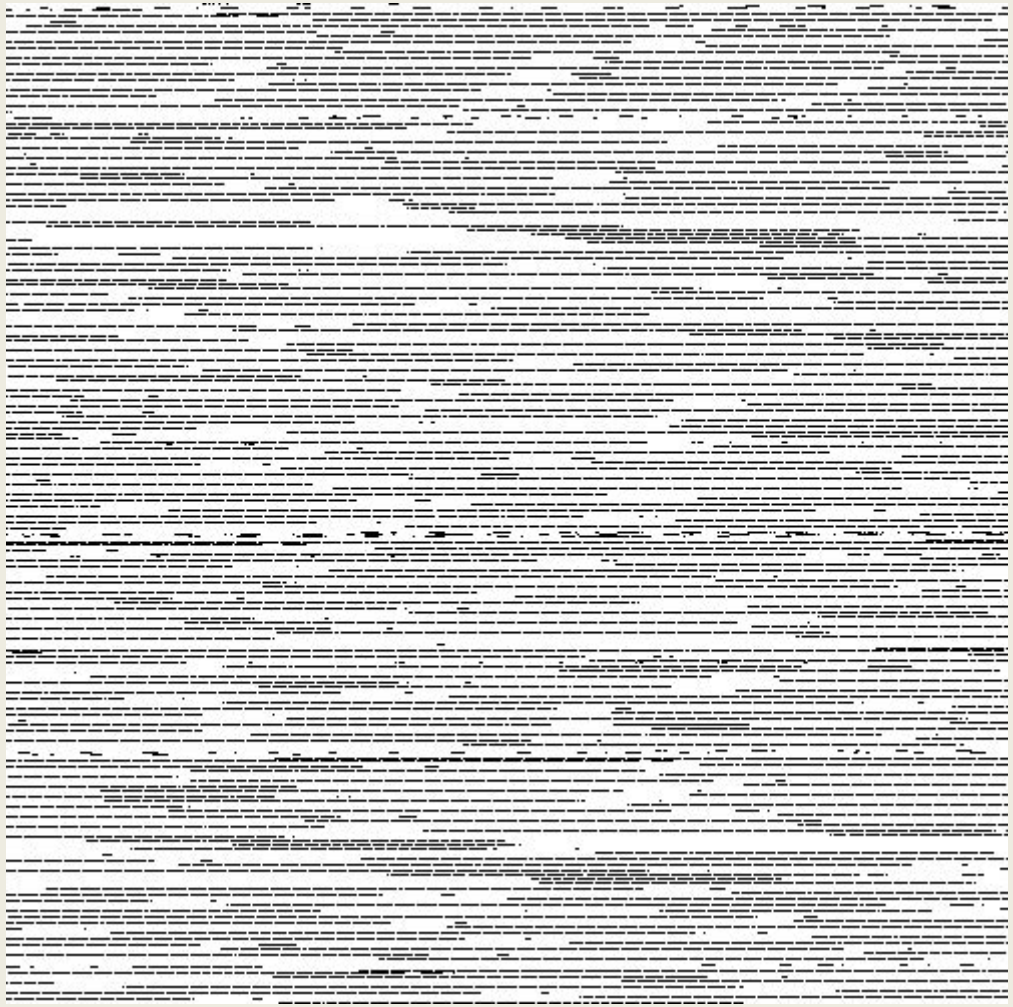
cybertrust

- ソケットの受信バッファサイズ (net.core.rmem_default と net.core.rmem_max)を増やしてみる

default 値

```
# sysctl net.core.rmem_default
net.core.rmem_default = 212992
# sysctl net.core.rmem_max
net.core.rmem_max = 212992
```

測定結果 - その3

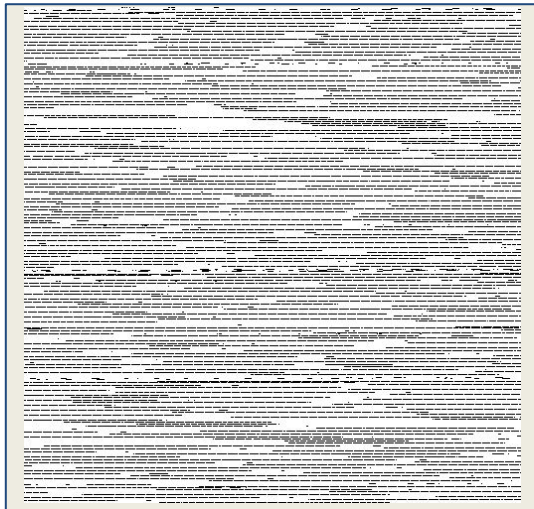


default buffer size: 208 KB
欠損率 19.97%

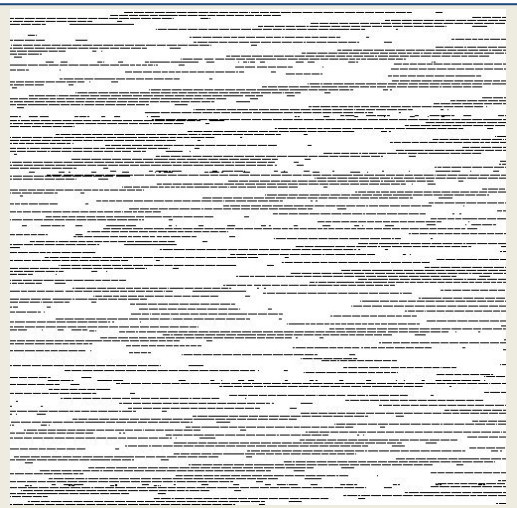


buffer size: 25 MB
欠損率 4.27%

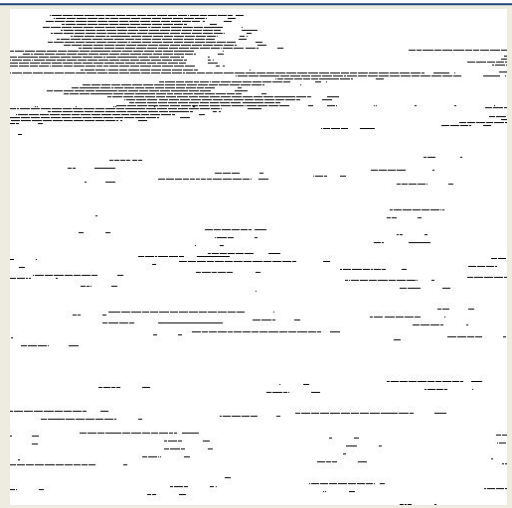
測定結果 - その3



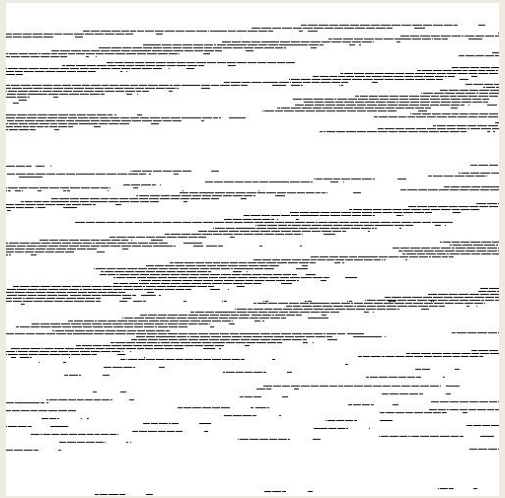
default buffer size: 208 KB
欠損率 19.97%



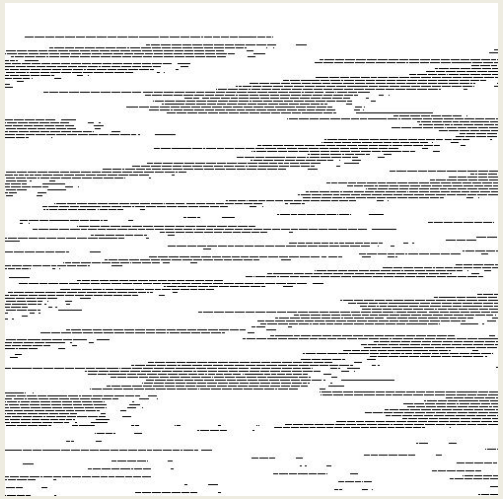
default buffer size: 512 KB
欠損率 12.68%



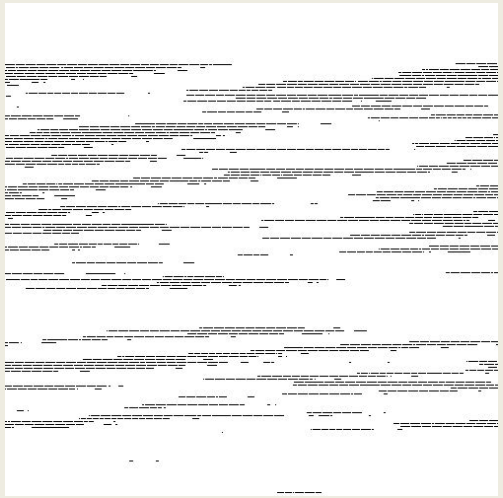
default buffer size: 1MB
欠損率 3.54%



default buffer size: 2MB
欠損率 7.12%



default buffer size: 4MB
欠損率 9.34%

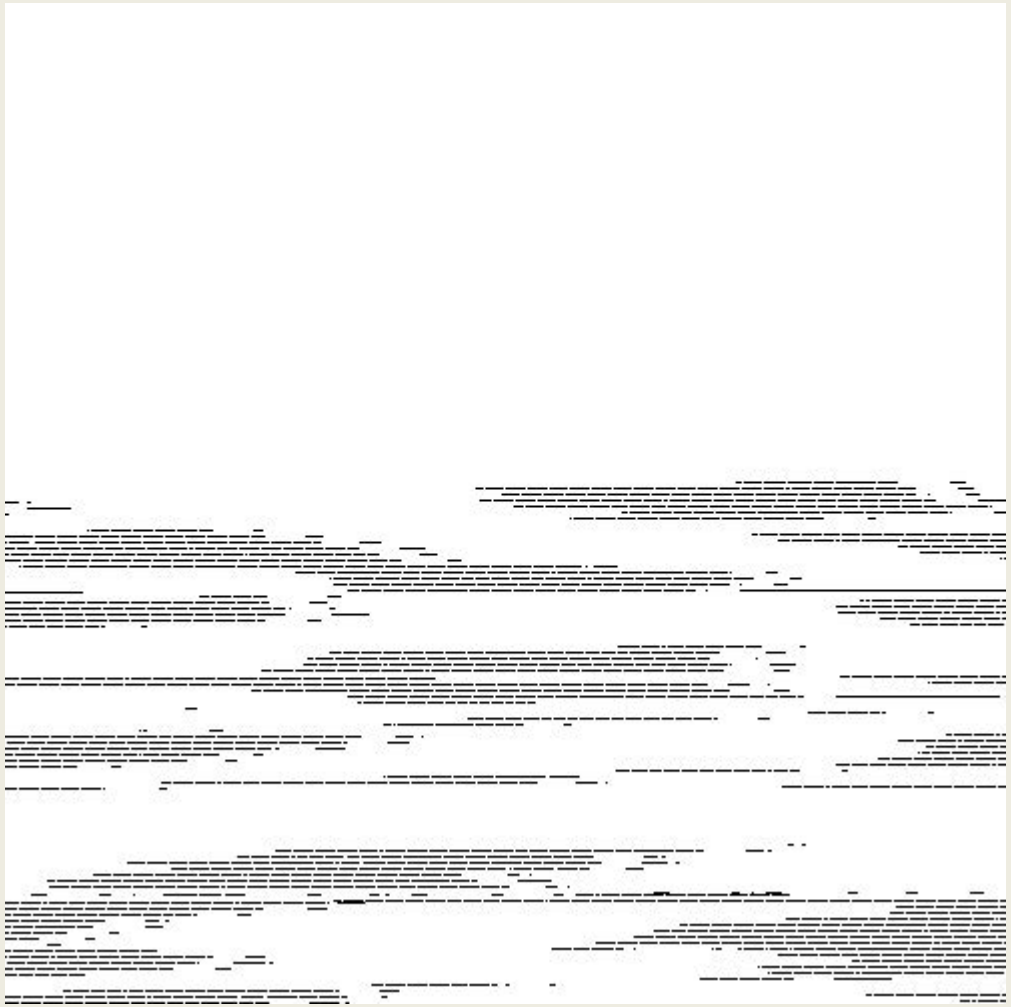


default buffer size: 8MB
欠損率 6.02%

測定結果 - その3



buffer size: 25 MB
欠損率 4.27%



buffer size: 50 MB
欠損率 5.32%

社内フィードバック

cybertrust

- まずは計測
  - ボトルネックがどこにあるか把握する
- 受信側のソケットバッファサイズを増やすことで改善
  - しかし，頭打ち（snmptrapd + snmptt の限界か？）
  - 一方で，CTJ（ミラクルリナックス）は新たな方法を模索
    - 詳細は弊社営業・プロダクトマーケティングへ

cybertrust

信頼とともに

留意事項

本資料に記載されている会社名、製品名、サービス名は、当社または各社、各団体の商標もしくは登録商標です。
その他本資料に記載されているイラスト・ロゴ・写真・動画・ソフトウェア等は、当社または第三者が有する知的財産権やその他の権利により守られております。
お客様は、当社が著作権を有するコンテンツについて、特に定めた場合を除き、複製、改変、頒布などをすることはできません。
本資料に記載されている情報は予告なしに変更されることがあります。また、時間の経過などにより記載内容が不正確となる場合がありますが、当社は、当該情報を更新する義務を負うものではありません。