

株式会社ミクシィ

TCAM利用量を 予測してみた

2018.07.12@JANOG42

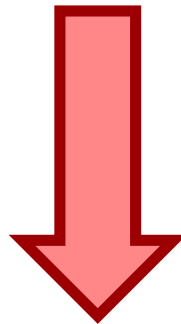
XFALG スタジオ

伊東 宏起



TCAM 溢れますよね？

ACL/Firewall Filter の追加・変更時にTCAMが溢れて想定外の事象が発生…なんて経験ありませんか？



追加・変更のメンテナンス実施前に
TCAM利用量の変化が予測できると嬉しい！

※TCAM(Ternary CAM): ルータ・L3 SW によく搭載されるメモリの一種

対策に悩む 🤔

案: ACL/Firewall Filter 追加・変更の度に
検証機に config を投入して TCAM の利用量を確認

- プロダクション環境投入前に完璧な消費量を知ることができる
- ✗ 検証機が潤沢にない場合、順番待ちなどチームのスピード感に影響が出てしまう
- ✗ 何よりも手間が多い

そうだ、自動化だ

案: config から TCAM の利用量を予測する
ツールを開発する

- config をツールに読み込ませるだけで
利用量が予測できるのでお手軽に使える！
- 検証機の必要なし！手元の PC でプログラムを
実行すれば OK
- ✗ あくまでも予測、大まかな値しか分からない

そうだ、自動化だ

xflagstudio/tcam_usage_calculator

- config から TCAM の利用量を予測する
コマンドラインツール
- Python3.6.x 製
- Junos に対応(特定の機器・バージョンで検証)

Usage

xflagstudio/tcam_usage_calculator

- config 全体の TCAM 利用量を予測

```
$ tcam_usage_calculator /path/to/config/RouterConfig  
TotalCost: 225
```

- filter 毎の TCAM 利用量を予測

```
$ tcam_usage_calculator /path/to/config/RouterConfig -d  
filterA: 30  
filterB: 3  
filterC: 32  
TotalCost: 65
```

計算アルゴリズム 🤔

- config ファイルからフィルター定義の行を抽出
- filter/term の名前一覧をデータ構造に変換
 - {'filterA': [{'termA1': 0}, {'termA2': 0}], filterB: [{'termB1': 0}]}
- prefix/protocol/next-header/port の定義から term 単位の生の TCAM 利用量を計算
 - この部分で次に紹介する最適化も考慮
- filter を適用した interface から、interface の重みを計算
- 生の TCAM 利用量 と interface の重みから、term 単位の実際の 利用量を予測
- term 単位の利用量から全体の利用量を計算

工夫ポイント 🍌

- 連番のポート番号のフィルターが定義されている場合の最適化の考慮
 - SSH(22)/Telnet(23) がルールとしてある場合
 - 2進数にすると、22-> 0001 0110, 23-> 0001 0111 となる
 - 1111 1110 をビットマスクして、0001 0110 が取り出せるのはこの2つのみ
 - つまり、0001 0110/7 に該当するポートという1ルールで2ポート分のルールを表現できる(= 最適化されるはず！)

まとめ

- ACL/Firewall Filter が消費する TCAM を事前に確認しておきたい
- config を解析して大まかにTCAM 利用量を計算するコマンドラインツールを実装
- 大きな差分を投入する前に簡単にインパクトを把握できるようになった

PR 募集！

xflagstudio/tcam_usage_calculator

- 予測精度向上！
- Junos 以外にも対応！
- 機能追加！

お待ちしております！

Thank You!

