A person in a dark suit and tie is holding a tablet. The tablet screen is filled with futuristic, glowing blue and white graphics. These include a world map, various hexagonal shapes, numbers like '235', '123', and '52315231', and a small inset image of a person at a podium. The background is a bright blue gradient with light rays.

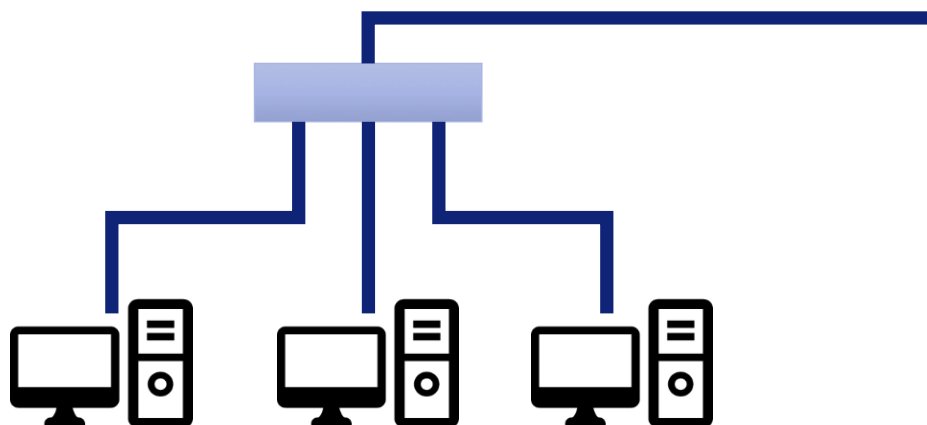
ネットワーク 解析ソフトの開発

- ・ ソフトウェアの概要
- ・ 各機能説明
- ・ ソフトウェアの今後の課題
- ・ 感想と考察

ネットワークアナライザとは？

Pcap-qt

- ネットワークに流れるパケットを解析するソフトウェア
- トラブルシューティングに使われる
- ネットワークの流れがわかる！



要件

- ・ Linuxで稼働する
- ・ ソフトウェアで提供する
- ・ データを見やすく表示する
- ・ DBへ暗号化し保存する
- ・ ソフトウェアの利用制限がある

仕様

CentOS7 (x86_64)

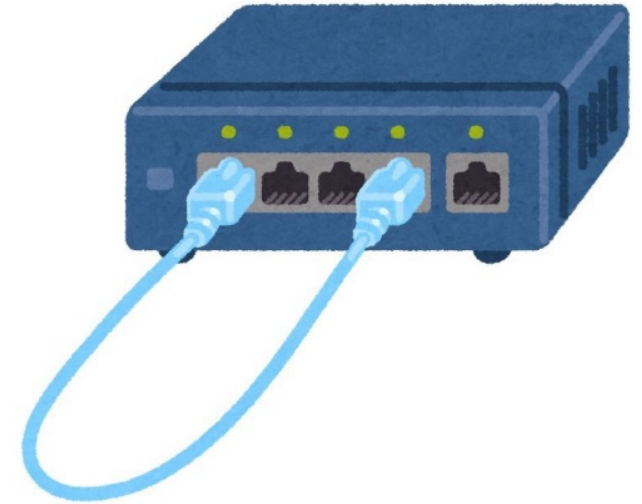
C++とQtによる開発

パケット一覧・グラフ・
リアルタイム可視化

ログを暗号化して保存

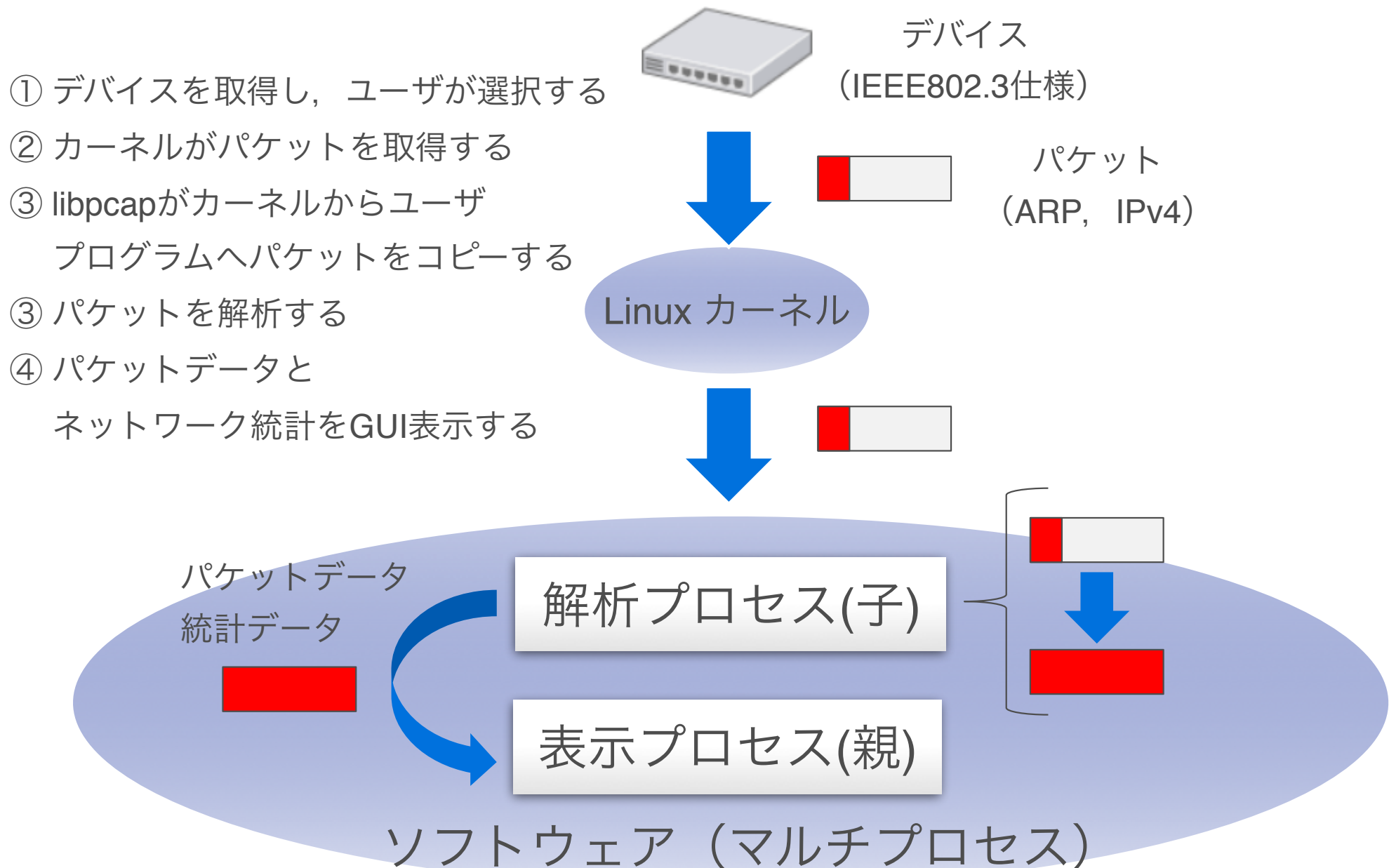
管理者権限で使用可能

- CentOS7 (64bit)
- Libpcap (1.5.3で動作確認)
- Qt (5.7/5.8で動作確認)
- MariaDB (10.1.3以降)
- データベース容量500MB (30日分のデータ格納可能)



システム全体構成

Pcap-qt



3つのプログラムを作成した
マルチプロセスで実行する

pcap-device

- ① デバイスを取得し、ユーザが選択する

pcap-capture

- ② カーネルがパケットを取得する
- ③ libpcapがカーネルからユーザプログラムへパケットをコピーする
- ③ パケットを解析する

pcap-qt

- ④ パケットデータとネットワーク統計をGUI表示する

The logo for Pcap-qt, featuring the text "Pcap-qt" in a white, sans-serif font, centered within a bright green rounded rectangle.

Pcap-qt

パケットの速度とフレームワークQtを意識した
スピード感のあるロゴの採用

キャプチャ可能なデータリンク

- IEEE802.3仕様

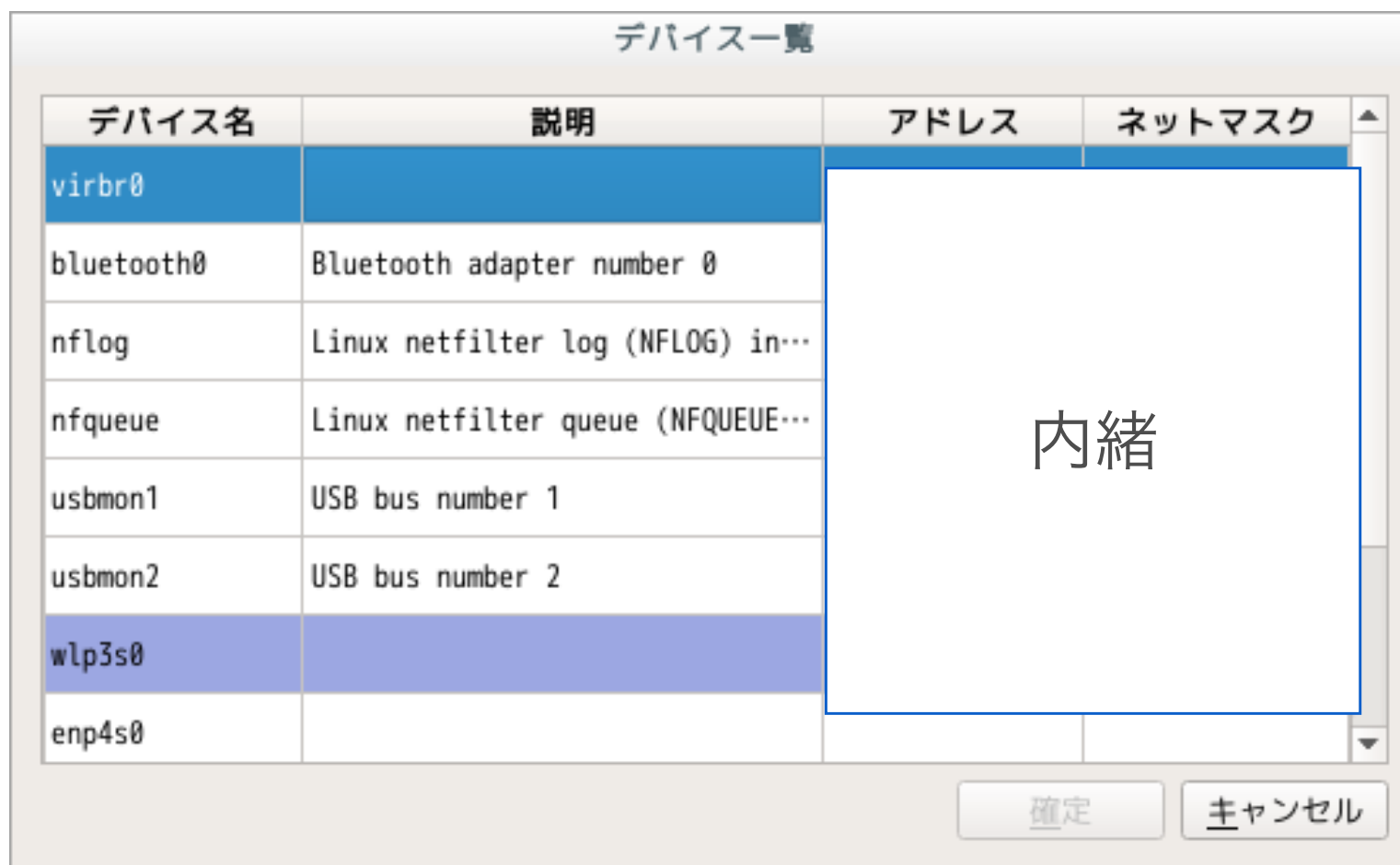
解析可能なプロトコル

- ARP
- IPv4
- TCP
- UDP
- ICMP
- DNS

識別可能なプロトコル

- HTTP(S)
- FTP
- Telnet
- SSH
- DHCP
- NTP

パケットキャプチャするデバイスを選択する



pcap-deviceから取得

メインフレーム（パケット一覧）

Pcap-qt

解析したパケットを表示する
エラーの有無・プロトコルごとに色分け

パケットキャプチャ						
表示 キャプチャ 分析 ヘルプ						
⏮ ⏪ ⏩ ⏭ 🔍 🌙 📊 📈 🗄 ?						
番号	取得時刻	送信元IPアドレス	宛先IPアドレス	パケット長	プロトコル	備考
31	44			42	ARP	
32	44			337	UDP	5] データグラム長:303
33	44			365	ICMP	先ホストとの通信が...
34	44			76	DNS	わせ(QUERY)] 再帰可...
35	44			257	DNS	わせ(QUERY)] 再帰希...
36	44			74	TCP] シーケンス番号:24...
37	44			74	TCP] シーケンス番号:18...
38	44			66	TCP] シーケンス番号:24...
39	44			256	TCP] シーケンス番号:24...
40	44			66	TCP] シーケンス番号:18...

pcap-captureから取得

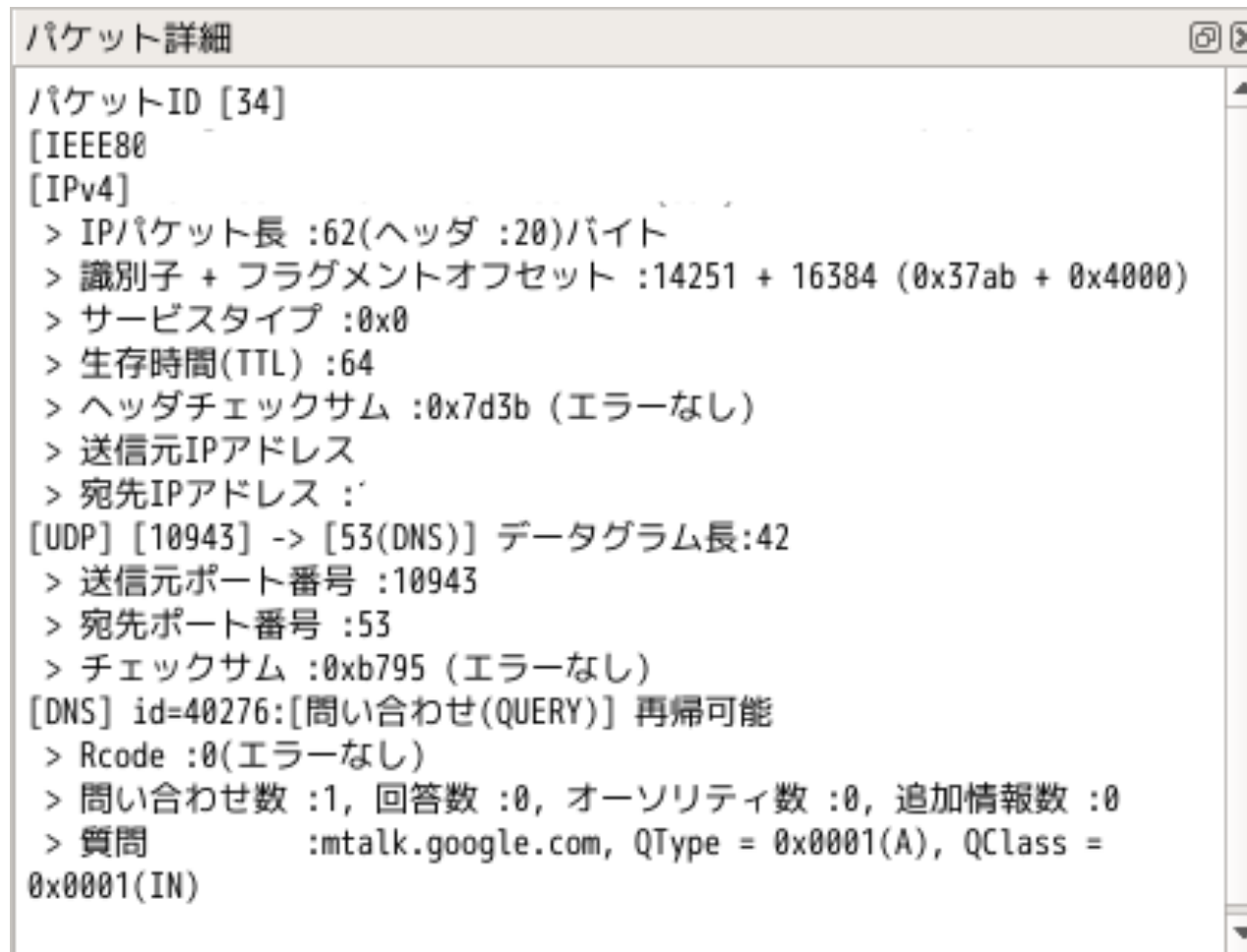
パケットの情報を詳しく表示する

IEEE802.3

IPv4

UDP

DNS



DNSメッセージ

実際にネットワークを流れている
データを16進数で表示する



DNSメッセージ

パケットのフィルタリング

Pcap-qt

簡単なUIで、キャプチャするパケットを絞り込む

フィルタ設定

ホスト ポート番号 プロトコル

ホスト名

・コンマ','または空白' 'で複数指定
(例) 192.168.2.111,201.10.1.19

オプション

☒ 送信元/宛先ホスト

☐ 送信元ホスト

☐ 宛先ホスト

☐ ホストフィルタを設定しない

フィルタ設定

ホスト ポート番号 プロトコル

ポート番号

・コンマ','または空白' 'で複数指定
・ハイフン'-'でポート番号の範囲指定
(例) 50,137-139,465

☒ アプリケーションプロトコル

☒ DNS ☒ HTTP(S)

☒ FTP(S) ☒ NTP

☒ DHCP ☒ Telnet

☒ SSH

オプション

☒ 送信元/宛先ポート番号

☐ 送信元ポート番号

☐ 宛先ポート番号

☐ ポート番号フィルタを設定しない

フィルタ設定

ホスト ポート番号 プロトコル

プロトコル

☒ ARP

☒ TCP

☒ UDP

☒ ICMP

気になるパケットデータを検索する

パケットキャプチャ

表示 主タブチャ 分析 ヘルプ

⏮ ⏪ ⏩ ⏭ 🔍 🔄 📊 📄 ?

番号	取得時刻	送信元IPアドレス	宛先IPアドレス	パケット長	プロトコル	備考
696	50			81	HTTPS	[443(HTTPS)] -> [58526] データグラ...
697	50			109	ICMP	宛先到達不能, ポートに到達できない
698	50			42	ARP	
699	53			81	HTTPS	[443(HTTPS)] -> [58526] データグラ...
700	53			109	ICMP	宛先到達不能, ポートに到達できない
701	60			81	HTTPS	[443(HTTPS)] -> [58526] データグラ...
702	60			109	ICMP	宛先到達不能, ポートに到達できない
703	73			81	HTTPS	[443(HTTPS)] -> [58526] データグラ...
704	73			109	ICMP	宛先到達不能, ポートに到達できない
705	73			90	NTP	[43149] -> [123(NTP)] データグラム...

検索

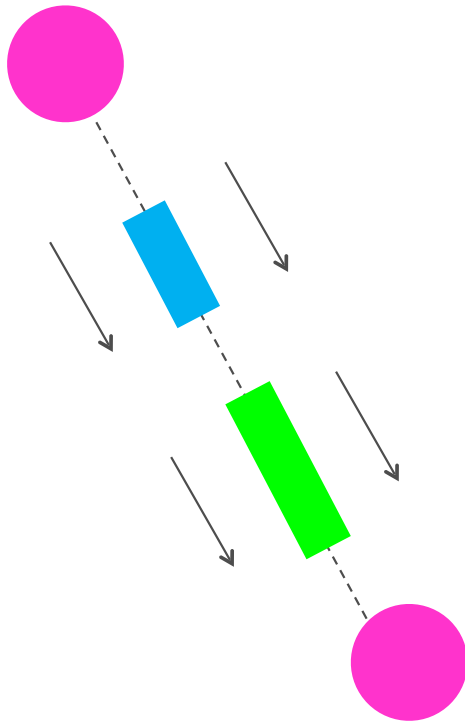
パケット詳細

パケットID [697]
[IEEE802.3] 検索文字列 ICMP 検索 < > 2/5 終了

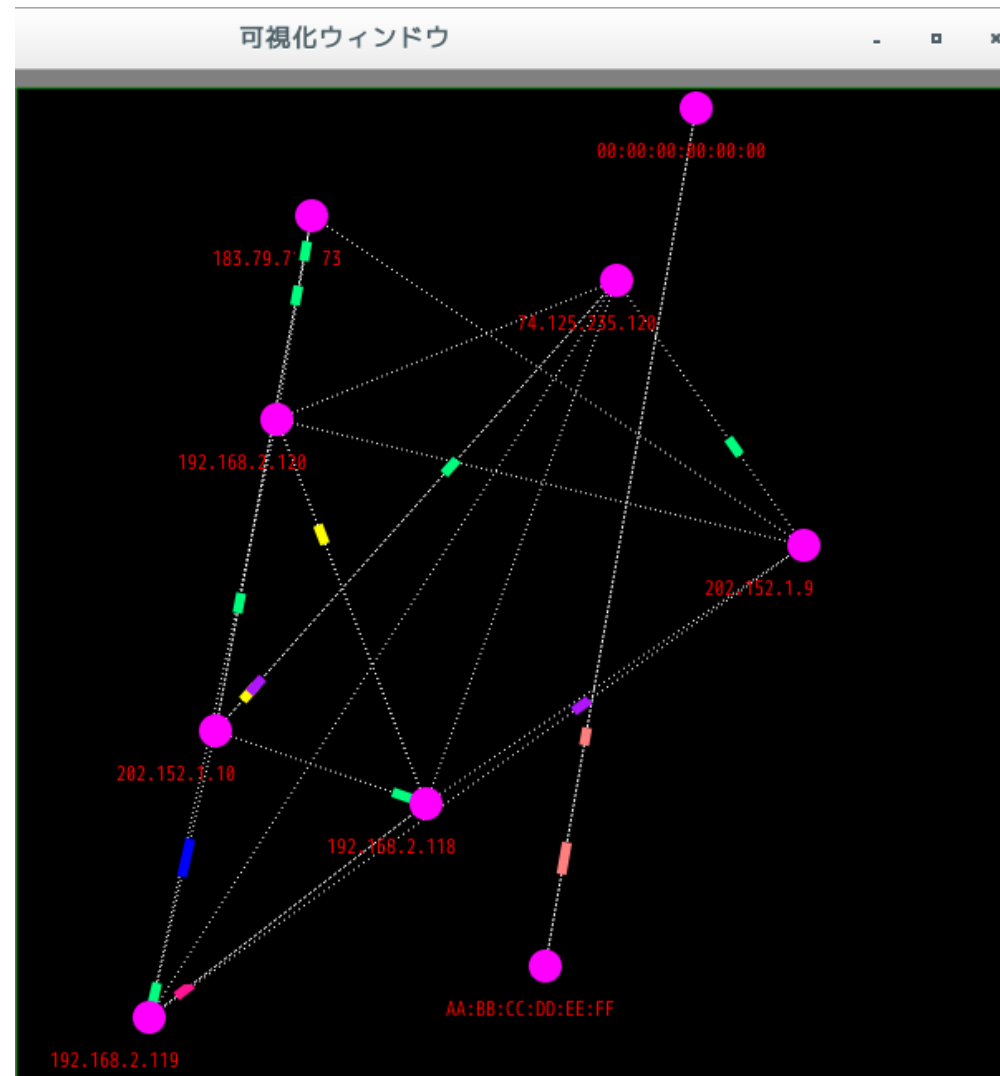
[IPv4]
> IPパケット長 : 95(ヘッダ : 20)バイト
> 識別子 + フラグメントオフセット : 59790 + 0 (0xe98e + 0x0)
> サービスタイプ : 0xc0
> 生存時間(TTL) : 64
> ヘッダチェックサム : 0xb46f (エラーなし)
[ICMP]宛先到達不能, ポートに到達できない
> ICMPタイプ : 3(宛先到達不能)
> ICMPコード : 3(ポートに到達できない)
チェックサム : 0x1dd9

0010
0020
0030
0040
0050
0060

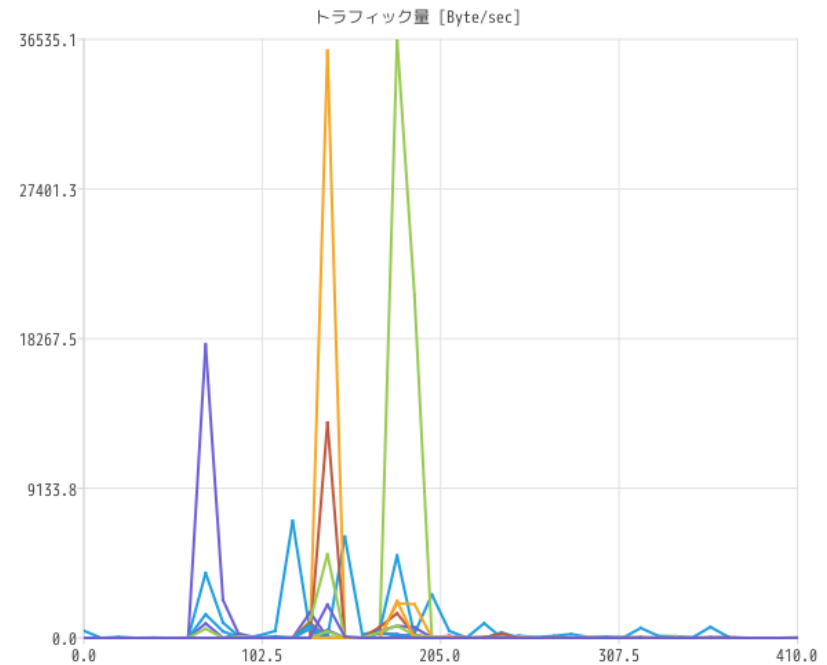
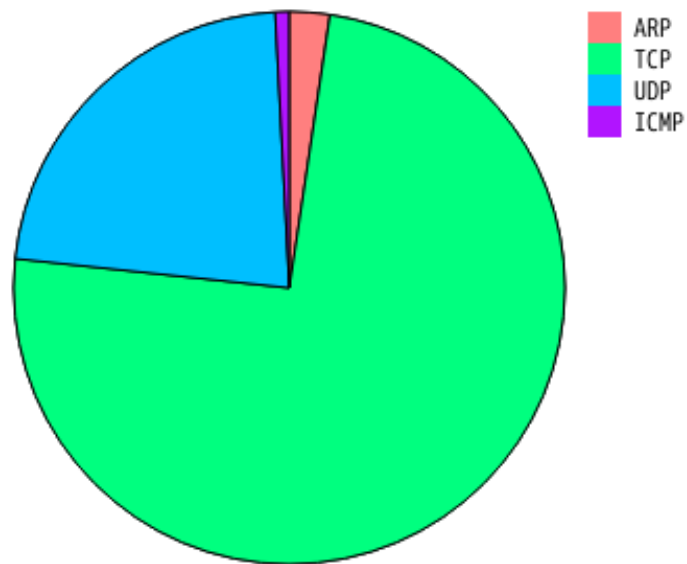
アニメーションでネットワークの流れを理解する



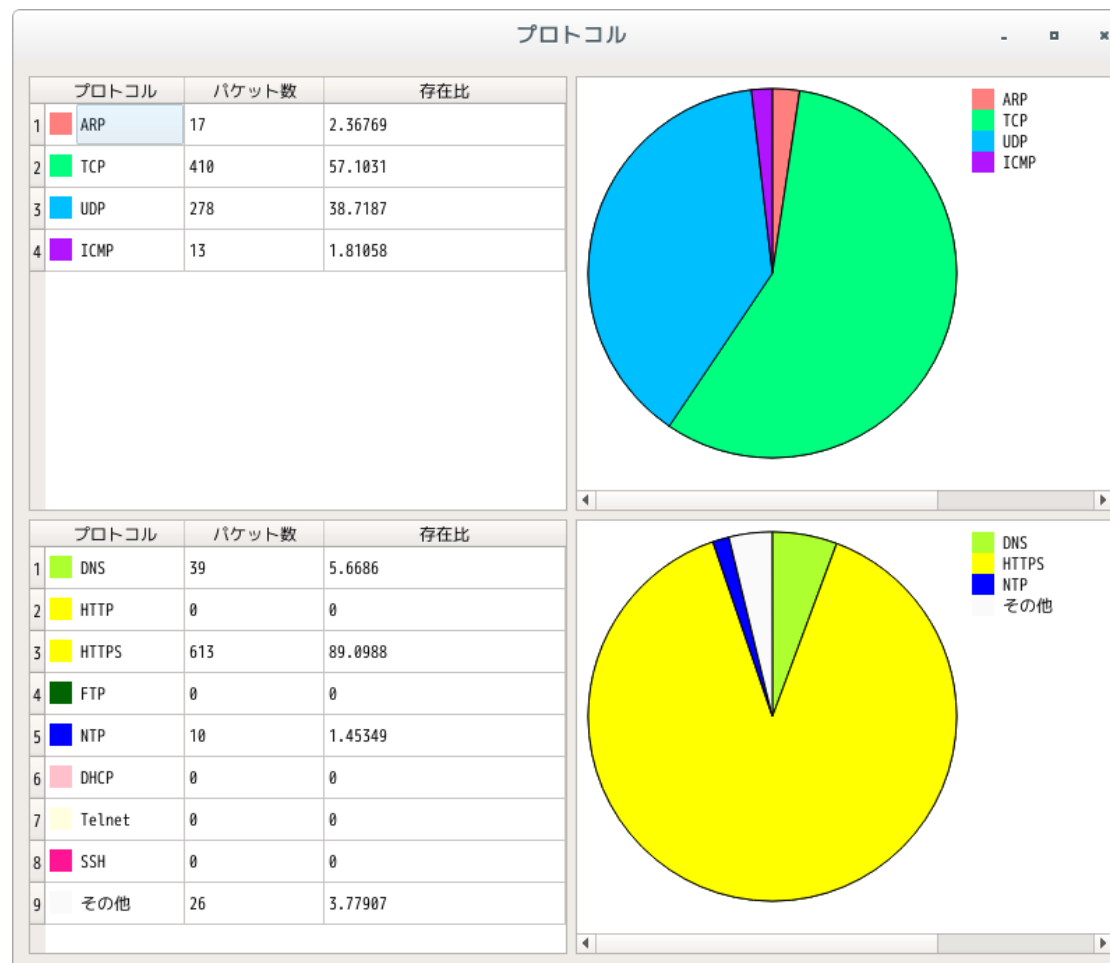
パケットが動く！



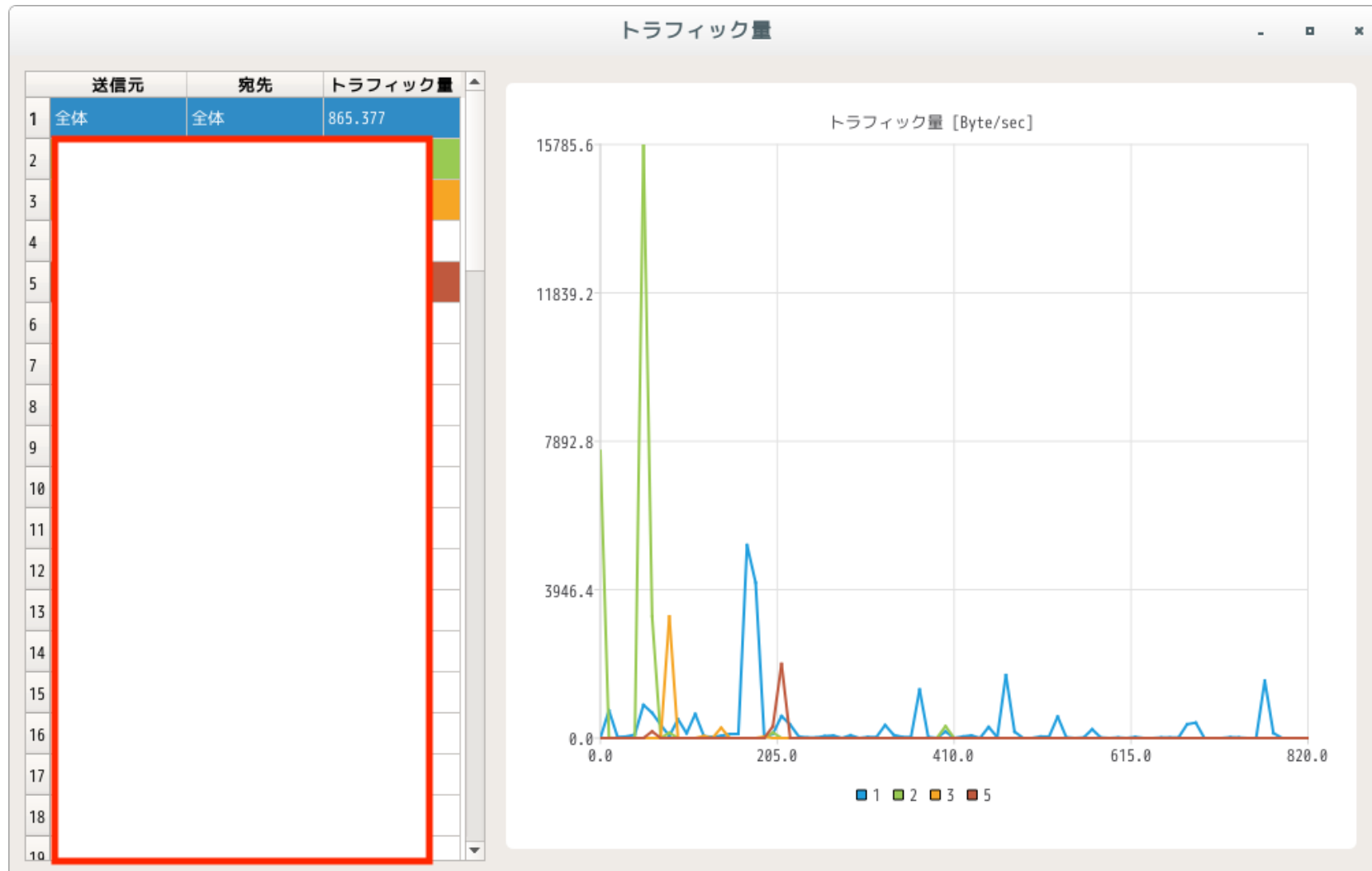
- プロトコル統計
- トラフィック量統計



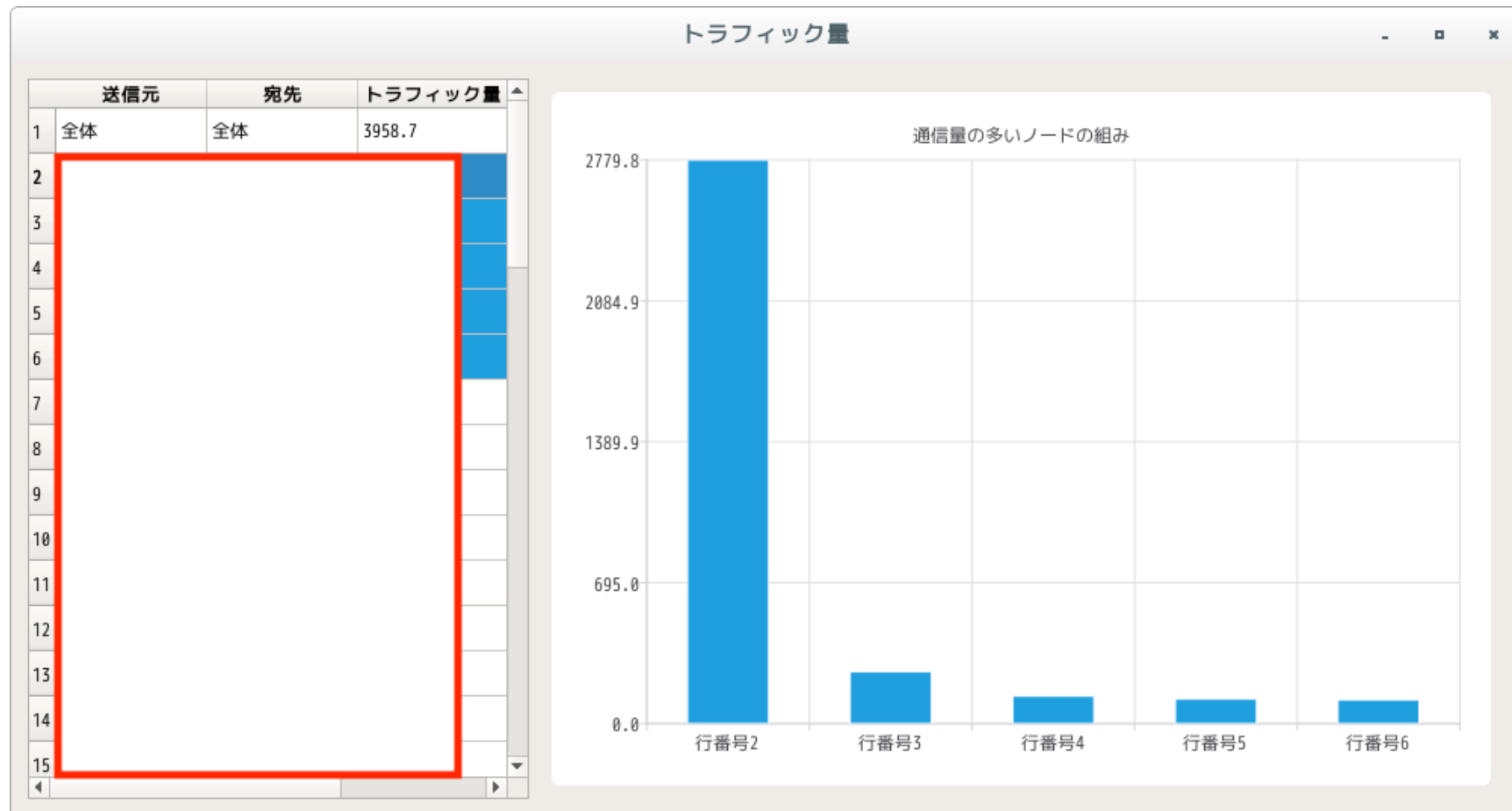
各プロトコルの割合を円グラフで表示する



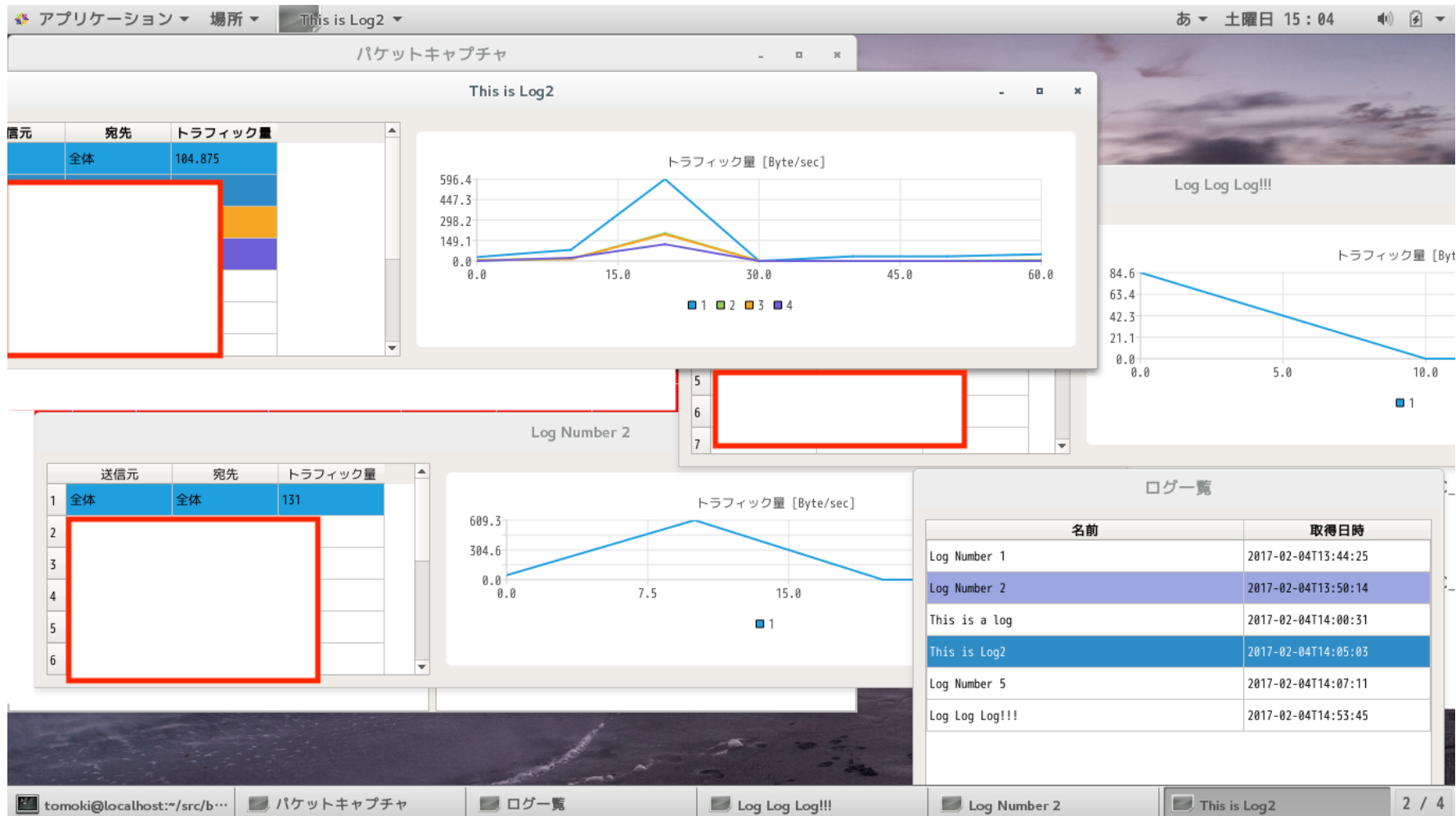
ノード間のトラフィック量を折れ線グラフで表示する



通信量の多い通信を棒グラフで表示する



過去30件までのネットワーク統計をデータベースに保存可能
データベース内で暗号化する



- ・メモリ占有による重い動作を改善する
- ・データベースをより効率的に設計する
- ・扱えるプロトコル/データリンクの数を増やす
- ・可視化ウィンドウを高機能にする
- ・折れ線グラフのスケールを動的に変化させる



俺たちの戦いはこれからだ

Pcap-qt

おしまい

