



JANOG36 Meeting in KITAKYUSHU

xFlow Tutorial

前半パート

Kazumasa Ikuta

kikuta@cisco.com

2015.7.15

xFlowチュートリアル

- xFlow(NetFlowやsFlow, IPFIX)はトラフィックの監視技術としては長い歴史があり、通信事業者に加え、今では企業や組織のLAN/WANでも一般的に利用されるようになっていきます。
- 一方で、メーカーや機器毎、リリース時期の実装によって差異があるなど、導入に踏み切れない、また導入しても使いこなせていないという声が少なからずあります。
- また、実はxFlowは枯れた存在でありつつも、今だに進化しているのです。
- 本チュートリアルでは、xFlowを運用に活用する上で必要な基礎知識、実際に使う上でのFAQや導入時のハマりどころを紹介していきます。単なるプロトコル説明にとどまらずに実践で使える知識とワザをお伝えします。

“フロー”って？

- *“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.”*
- フローとは、ネットワーク装置を通過する、共通の属性をもった一方通行の連続したパケットのことである
- フロー記録は、IPアドレスやパケットカウントやバイトカウント、タイムスタンプ、ToS、アプリケーションポート、入出カインターフェースなどを含むことがある
- *“Exported NetFlow data is used for a variety of purposes, including enterprise accounting and departmental chargebacks, ISP billing, data warehousing, network monitoring, capacity planning, application monitoring and profiling, user monitoring and profiling, security analysis, and data mining for marketing purposes.”*
- 出力されたNetFlowデータは、多様な用途に使われ、例えば以下のような例がある
- 企業でのアカウンティング、部門チャージ、ISPによる課金、データウェアハウジング、ネットワーク監視、容量計画、アプリケーション監視、アプリケーションプロファイリング、ユーザ監視、ユーザプロファイリング、セキュリティ分析、マーケティング目的のデータマイニングなど

チュートリアルの流れ： フローの世界へようこそ！

- はじめに
- NetFlow
- Tips



前半パート：生田

- sFlow
- コレクター
- FAQとTips
- まとめ



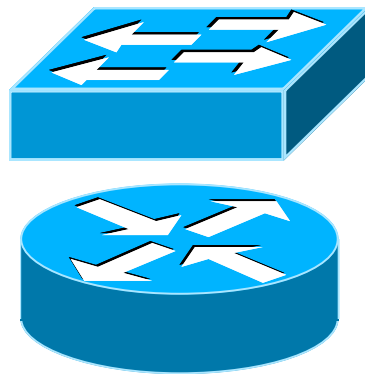
後半パート：田島

免責条項

- メーカー固有の情報はなるべく出てこないように心がけていますが、机上論にならないように設定例などは登場します
- 個別のコレクターに偏らないように心がけていますが、あらゆるコレクターに詳しいわけではないので、多少偏るかもしれません
- 技術的な仕様を、包括的に網羅するものではなく、なるべく現実的な利用に役立てて頂けるように心がけています
- 心がけ大事！
- 念のためですが、OpenFlowは含みません

はじめに

登場人物

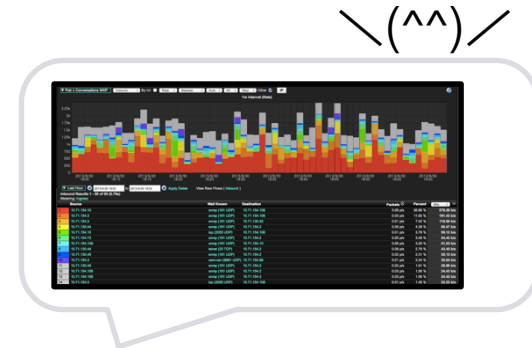


xFlow対応・ネットワーク装置
(エクスポーター)

Push
Protocol

ペイロード
(複数の
フロー情報)

パケット



xFlow対応サーバ
(コレクター)

xFlowユースケース

- 某社では、LAN/WAN、インターネット回線の監視を、SNMP中心に行っていた
- SNMPでは利用量は把握できるが、利用内訳が不明であり、ビジネスへの貢献度という意味でトラフィックをカテゴリー分けした説明ができなかった
- ネットワーク基盤の使われ方について、より細かな情報を調査し、理解する必要に迫られた



- ワームやDDoS攻撃検出の仕組み強化になった
- 輻輳の原因に関わるアプリケーションを識別し、投資価値の判断材料を得た
- 適切なQoSパラメータの根拠を得て、設定を有効化
- VPNTrafficおよびテレワーカーのトラフィックパターン分析に利用

なぜ今、改めて xFlowか？ ニーズ高まりと、敷居の低下

- ネットワーク利用可視化要件と説明責任
- 容量計画とコスト削減
- 異常検出とセキュリティ対策
- サポート装置の拡大、性能向上
- サポートアプリケーションの普及
- 初期設備投資の低下
- サポート機能の強化

通信事業者に加えて、企業WAN・LANも普及



NFDUMP, NfSen

FlowViewer

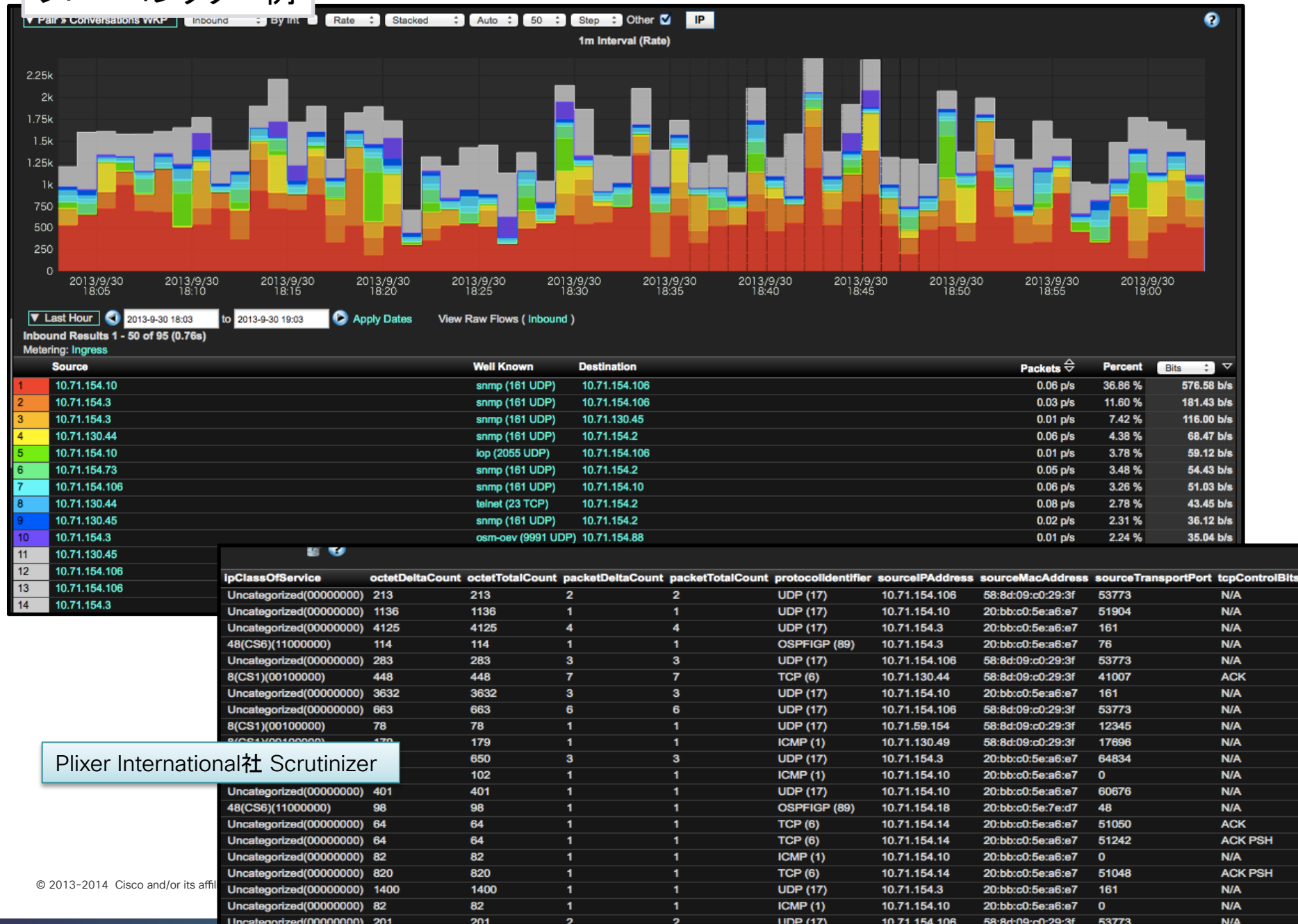


NetFlow関連の様々な実装

- Citrix NetScaler – AppFlow reports including Application, Connections, Request URL, Request host, and User Agent reporting
- Dell SonicWALL – DPI, Intrusions, Spyware, URLs, User reporting, VoIP monitoring, and more
- Juniper – Src/Dst IPv4/IPv6, Port, Vlan, AS..
- Open vSwitch
- Palo Alto Networks – Firewall event reporting, NAT reporting, DPI, and User reporting
- Riverbed Steelhead – numerous RTT reports, WAN Optimization reporting
- VMWare – VM (ESX hosts) information, dvswitch
- VyOS (Vyatta)
- Cisco ASA – specialized reporting for VPN users, ACL flow volumes, Network Address Translation reports
- Cisco Wireless Controllers – DPI (Application) reports, wireless hosts reporting, SSID lists
- Cisco Application Visibility and Control (AVC) – numerous custom reports for DPI, Application Performance, Performance Monitoring
- ...

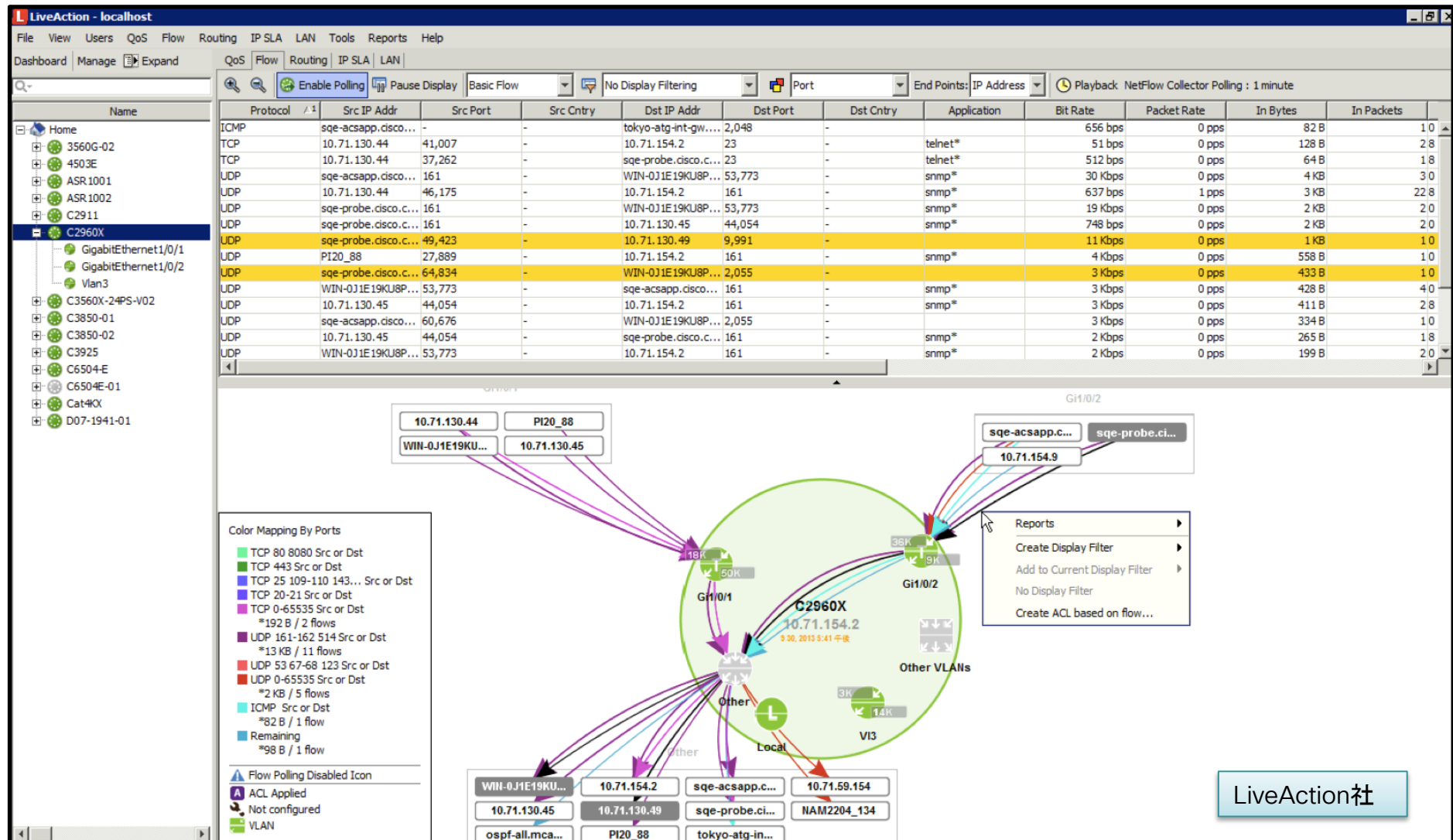
参考: <https://en.wikipedia.org/wiki/NetFlow>

フローコレクター例



Plixer International社 Scrutinizer

フローコレクター例



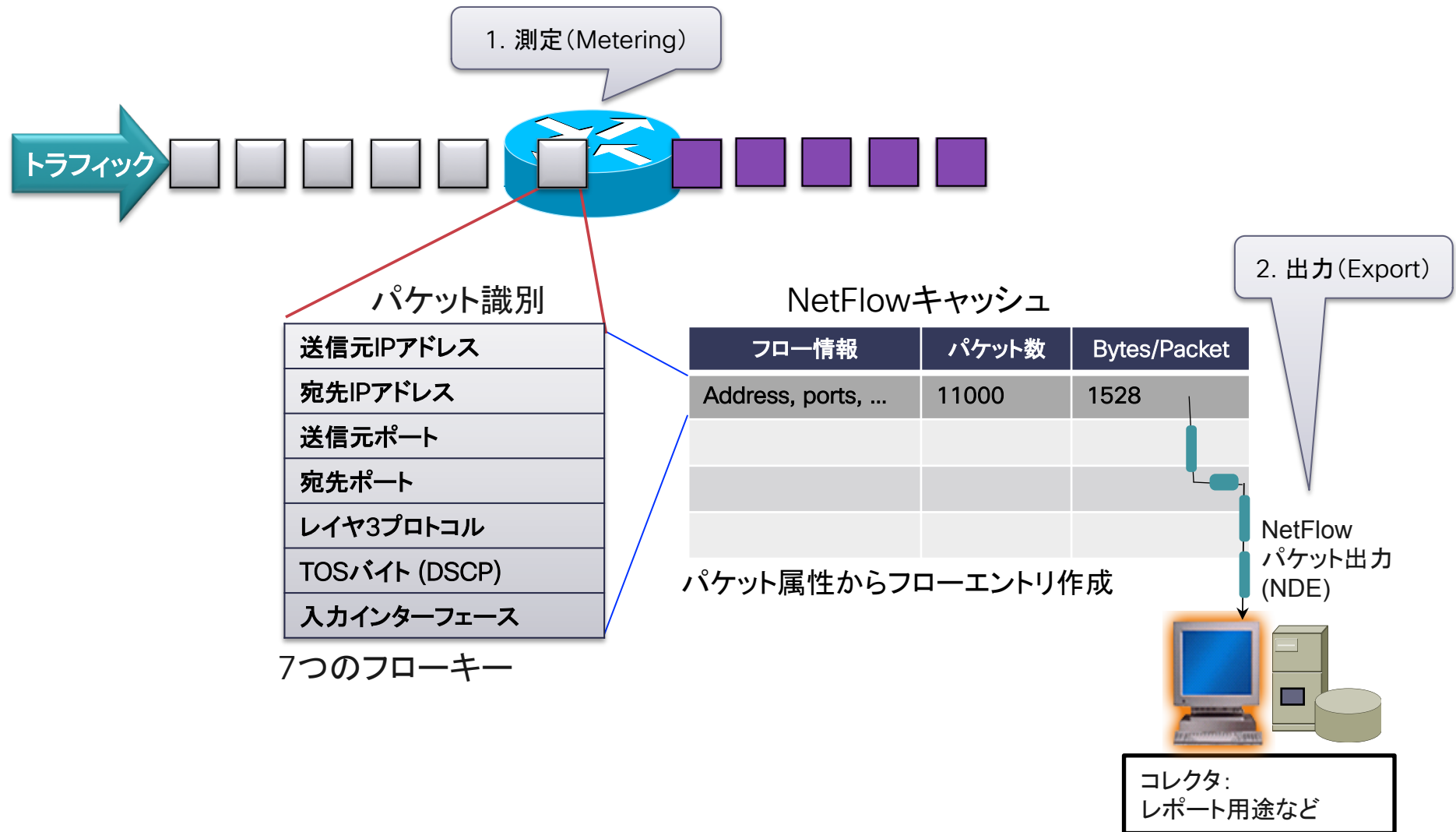
NetFlow

NetFlow

- シスコシステムズが開発、1996年にパテント取得
- トラフィック情報収集のデファクトスタンダード (version 5)
- 現在は、NetFlow version 5 と NetFlow version 9が主流
 - RFC 3954
Cisco Systems NetFlow Services Export Version 9 (Informational)
- 2013年にRFC7011-7015として標準化 (IPFIX)
 - RFC 7011
Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information (Standards Track)
 - 他、RFC7012, 7013, 7014, 7015など
 - (NetFlow version 10)
- 基本的には1:1のキャッシュ技術であり、パケットサンプリング (1:1000 など) はオプション設定
 - 一部機種はハードウェア制約によりサンプリングのみサポートのものもある
- ネットワークトラフィック監視、セキュリティ監視、ネットワークプランニング、トラフィック解析、IPレベルの課金などに利用される
- 全フロー情報を出力する特性を活かして、セキュリティ用途や、イベントログとしての応用が注目されている



NetFlowアーキテクチャ



Version 5 フローフォーマット

フロー・キー: フローが識別される条件
ノン・キー: フローごとに収集される情報

フローキー と ノン・キー フィールド

From/to

利用量

- パケットカウント
- バイトカウント

- 送信元IPアドレス
- 宛先IPアドレス

時間

- 開始 sysUpTime
- 終了 sysUpTime

- 送信元 TCP/UDP ポート番号
- 宛先 TCP/UDP ポート番号

ポート

- 入力 ifIndex
- 出力 ifIndex

- ネクストホップアドレス
- 送信元 AS 番号
- 宛先 AS 番号
- 送信元プレフィックスマスク
- 宛先プレフィックスマスク

L4アプリケーション

QoS

- Type of Service
- TCP フラグ
- プロトコル番号

ルーティングと
ピアリング

Version 5 NDE NetFlow Data Export

Pkt	Time(s)	Size	Source	Destination	Protocol	Info
1	0.000	1174	192.168.99.23	192.168.61.96	CFLOW	total: 23 (v5) flows
2	12.000	1414	192.168.99.23	192.168.61.96	CFLOW	total: 28 (v5) flows
3	24.000	1222	192.168.99.23	192.168.61.96	CFLOW	total: 24 (v5) flows

+ UDP	User Datagram Protocol, Src Port: 64878 (64878), Dst Port: 3000 (3000)
- CFLOW	Cisco NetFlow/IPFIX
CFLOW	Version: 5
CFLOW	Count: 23
CFLOW	SysUptime: 1143711644
CFLOW	Timestamp: Apr 8, 2010 17:12:07.197163876
CFLOW	CurrentSecs: 1270746727
CFLOW	CurrentNSecs: 197163876
CFLOW	FlowSequence: 1005584
CFLOW	EngineType: 0
CFLOW	EngineId: 0
CFLOW	00..... = SamplingMode: No-sampling-mode-configured (0)
CFLOW	..00 0000 0000 0000 = SampleRate: 0
CFLOW	pdu 1/23
CFLOW	SrcAddr: 192.168.40.2 (192.168.40.2)
CFLOW	DstAddr: 192.168.99.21 (192.168.99.21)
CFLOW	NextHop: 192.168.99.21 (192.168.99.21)
CFLOW	InputInt: 19
CFLOW	OutputInt: 1
CFLOW	Packets: 10
CFLOW	Octets: 600
CFLOW	[Duration: 0.148000000 seconds]
CFLOW	StartTime: 1143689.548000000 seconds
CFLOW	EndTime: 1143689.696000000 seconds
CFLOW	SrcPort: 10000
CFLOW	DstPort: 55557
CFLOW	padding
CFLOW	TCP Flags: 0x10
CFLOW	Protocol: 17
CFLOW	IP ToS: 0x00
CFLOW	SrcAS: 0
CFLOW	DstAS: 0
CFLOW	SrcMask: 24 (prefix: 192.168.40.0/24)
CFLOW	DstMask: 24 (prefix: 192.168.99.0/24)
CFLOW	padding
CFLOW	pdu 2/23
CFLOW	SrcAddr: 192.168.99.21 (192.168.99.21)

NDEのヘッダ

個々のフロー情報
(フローレコード)

NetFlowキャッシュの生成～出力

1.NetFlow cacheの生成、更新

SrcIrf	SrcIPadd	DstIrf	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.0.23.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	00A1	/24	180	00A1	/24	15	10.0.23.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.0.23.2	1040	24.5	14

2.期限切れ (タイマー)

- ・ Inactive Timer 終了 (デフォルト15秒)
- ・ Active Timer 終了 (デフォルト30分)
- ・ NetFlowキャッシュが一杯 (古いフローから期限切れ)
- ・ RST または FIN TCP フラグ

SrcIrf	SrcIPadd	DstIrf	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1800	4

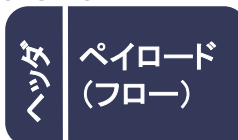
3.集約の有無

4.出力バージョン

集約なしフロー出力バージョン: version 5 or 9

5.転送プロトコル

出力
パケット



Yes

Protocol-Port 集約の例

Protocol	Pkts	SrcPort	DstPort	Bytes/Pkt
11	11000	00A2	00A2	1528

集約フロー出力バージョン: Version 8 or 9

NetFlow v5 コンフィギュレーション例

```
Router(config)# interface <slot/port/subinterface>
Router(config-if)# ip flow ingress
Router(config-if)# ip flow egress

Router(config)# ip flow-cache entries <number>
Router(config)# ip flow-cache timeout active <minutes>
Router(config)# ip flow-cache timeout inactive <seconds>

Router(config)# ip flow-export version 5 peer-as
Router(config)# ip flow-export destination 10.10.10.10 2055
Router(config)# ip flow-export source loopback 0
```

NetFlowキャッシュ: IOS showコマンド例

```
1812W-AG-K05-01#sh ip cache flow
```

```
IP packet size distribution (106188864 total packets):
```

```
1-32 64 96 128 160 192 224 256 288 320 352 384 416 448 480
.000 .104 .006 .000 .000 .000 .888 .000 .000 .000 .000 .000 .000 .000 .000
```

```
512 544 576 1024 1536 2048 2560 3072 3584 4096 4608
.000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
```

パケット長ごと割合

```
IP Flow Switching Cache, 278544 bytes
```

```
30 active, 4066 inactive, 2310263 added
```

```
49835813-ager polls, 0-flow alloc failures
```

```
Active flows timeout in 1 minutes
```

```
Inactive flows timeout in 10 seconds
```

```
IP Sub Flow Cache, 34056 bytes
```

```
60 active, 964 inactive, 3316053 added, 2310263 added to flow
```

```
0 alloc failures, 0 force free
```

```
1 chunk, 1 chunk added
```

```
last clearing of statistics never
```

フロー数

Protocol	Total Flows	Flows /Sec	Packets /Flow	Bytes /Pkt	Packets /Sec	Active(Sec) /Flow	Idle(Sec) /Flow
TCP-WWW	29733	0.0	6	156	0.1	4.4	3.2
UDP-other	2267280	1.9	46	184	92.5	0.9	13.3
ICMP	6225	0.0	1	82	0.0	0.0	13.3
IP-other	6995	0.0	16	81	0.0	140.5	0.9
Total:	2310233	2.0	45	184	92.8	1.3	13.1

アプリケーション流量

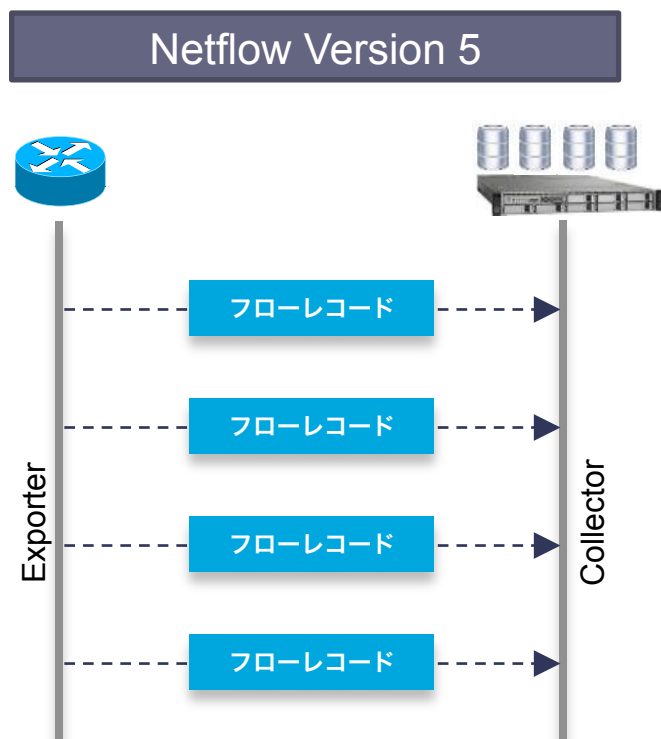
SrcIf	SrcIPAddress	DstIf	DstIPAddress	Pr	SrcP	DstP	Pkts
VI40	192.168.40.2	Fa0*	192.168.99.21	11	07AF	DAC4	1
Fa0	192.168.99.21	VI40	192.168.40.2	11	DAC4	07AF	1
Fa0	192.168.99.21	VI40	192.168.40.2	11	DAC4	2710	10
VI40	192.168.40.2	Fa0*	192.168.99.21	11	2710	DAC4	10
Fa0	192.168.61.254	VI40	192.168.40.2	11	270F	F5B3	1000
Fa0	192.168.99.21	VI40	192.168.40.2	11	DB72	2710	10
VI40	192.168.40.2	Fa0*	192.168.99.21	11	2710	DB72	10
Fa0	192.168.61.254	VI40	192.168.40.2	11	270F	D5F7	297
VI40	192.168.40.2	Fa0*	192.168.99.21	11	07AF	DB72	1
Fa0	192.168.99.21	VI40	192.168.40.2	11	DB72	07AF	1
Fa0	192.168.61.254	VI40	192.168.40.2	11	07AF	D5F7	1

カンバセーション

```
<snip>
```

NetFlow version 9/IPFIX

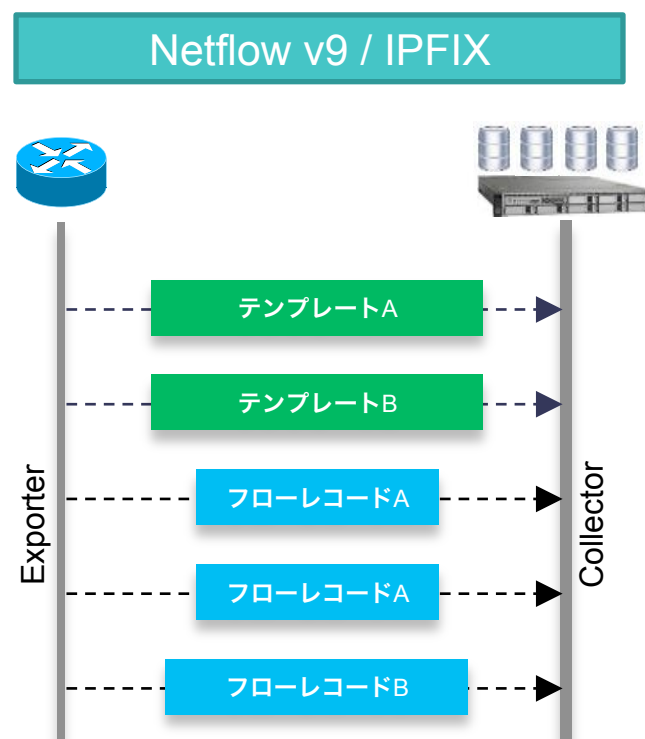
固定されたフロー出力



- 18の固定フィールド

送信元・宛先IPアドレス、送信元・宛先ポート番号、入力インターフェース、TOS、出力インターフェース、パケット・バイト数など

動的で拡張性があるフロー出力



- 測定内容はエクスポーターで自由に実装できる
- フロー形式はフローレコードとは別にコレクタに伝達される(テンプレート)
- 既存の仕組みの中であらゆるデータを出力できる
- オプションフローフォーマット、オプションフローレコードによりフロー以外の関連情報も伝達できる

NetFlow version 9/IPFIX

- RFC3954 “Cisco Systems® NetFlow Services Export Version 9” (Informational)
- バージョン9 のエクスポート・フォーマットは、**テンプレート**を使用しており、IP パケットのフローを、柔軟性が高く拡張性に富んだ方法で観測することができるようになっている。(RFC3954)
- テンプレートとは、その**構造と意味について記述されたフィールドの並び**である。(RFC3954)
- テンプレート
 - テンプレート ... データレコードの構造をコレクタに伝える(テンプレートID、Type、Length)
 - オプションテンプレート ... データレコード以外の情報をコレクタに伝える
- データレコード
 - フローデータ ... テンプレートIDとデータ値
 - オプションフローデータ
- テンプレートベースのメリット
 - 新フィールドのサポートが容易 ... 出力フォーマット構造の変更不要
 - コレクタ側では新フィールドを理解せずにフローレコードの”解釈”が可能
- NetFlow version9をもとに、IPFIXとして標準化 (NetFlow version 10)
 - RFC 7011, 7012, 7013, 7014, 7015など
 - セキュリティ、輻輳回避、データ整合性など、様々な拡張

NetFlow version 9 テンプレートとデータ例

Template Flowset

massaito-ASR1K_NFv9_FNF_NBAR.pcap - Wireshark

Filter: cflow

No.	Time	Source	Destination	Protocol	Info
235	38.186019	192.168.0.180	192.168.0.100	CFLOW	total: 26
238	38.395327	192.168.0.180	192.168.0.100	CFLOW	total: 1
239	39.189903	192.168.0.180	192.168.0.100	CFLOW	total: 49
240	39.189927	192.168.0.180	192.168.0.100	CFLOW	total: 49

Timestamp: Jan 5, 2011 10:08:37.000000000 (0.000000000)

FlowSequence: 1049

SourceId: 256

FlowSet 1

- Template FlowSet: 0
- FlowSet Length: 40
- Template (id = 260, Count = 8)
- Template Id: 260
- Field Count: 8
- Field (0/8)
 - Type: IP_SRC_ADDR (8)
 - Length: 4
- Field (1/8)
 - Type: IP_DST_ADDR (12)
 - Length: 4
- Field (2/8)
 - Type: PROTOCOL (4)
 - Length: 1
- Field (3/8)
 - Type: APPLICATION_ID (95)
 - Length: 4
- Field (4/8)
 - Type: BYTES (1)
 - Length: 4
- Field (5/8)
 - Type: PKTS (2)
 - Length: 4
- Field (6/8)
 - Type: FIRST_SWITCHED (22)
 - Length: 4
- Field (7/8)
 - Type: LAST_SWITCHED (21)
 - Length: 4

Data Flowset

massaito-ASR1K_NFv9_FNF_NBAR.pcap - Wireshark

Filter: cflow

No.	Time	Source	Destination	Protocol	Info
235	38.186019	192.168.0.180	192.168.0.100	CFLOW	total: 26 (v9) records
238	38.395327	192.168.0.180	192.168.0.100	CFLOW	total: 1 (v9) record
239	39.189903	192.168.0.180	192.168.0.100	CFLOW	total: 49 (v9) records
240	39.189927	192.168.0.180	192.168.0.100	CFLOW	total: 49 (v9) records

Frame 239: 1490 bytes on wire (11920 bits), 1490 bytes captured (11920 bits)

Ethernet II, Src: Cisco_82:6b:00 (d0:d0:fd)

Internet Protocol, Src: 192.168.0.180 (192.168.0.180), Dst: 192.168.0.100 (192.168.0.100)

User Datagram Protocol, Src Port: 64089 (64089), Dst Port: 80 (80)

Cisco NetFlow/IPFIX

Version: 9

Count: 49

Sysuptime: 12307586

Timestamp: Jan 5, 2011 10:08:38.000000000 (0.000000000)

FlowSequence: 1050

SourceId: 256

FlowSet 1

Data FlowSet (Template Id): 260

FlowSet Length: 1428

Flow 1

- SrcAddr: 10.1.3.12 (10.1.3.12)
- DstAddr: 10.2.100.2 (10.2.100.2)
- Protocol: 6
- ApplicationID: 1
- Octets: 84
- Packets: 2
- [Duration: 0.000000000 seconds]

Flow 2

- SrcAddr: 10.1.3.12 (10.1.3.12)
- DstAddr: 10.2.100.2 (10.2.100.2)
- Protocol: 6
- ApplicationID: 3
- Octets: 1348
- Packets: 5
- [Duration: 0.000000000 seconds]

Flow 3

- SrcAddr: 10.1.4.12 (10.1.4.12)
- DstAddr: 10.2.100.3 (10.2.100.3)
- Protocol: 6
- ApplicationID: 4
- Octets: 1348
- Packets: 5
- [Duration: 0.000000000 seconds]

Option Template

massaito-ASR1K_NFv9_FNF_NBAR.pcap - Wireshark

Filter: cflow

No.	Time	Source	Destination	Protocol	Info
30345	3369.36156	192.168.0.180	192.168.0.100	CFLOW	total: 16 (v9) records
30347	3370.36172	192.168.0.180	192.168.0.100	CFLOW	total: 12 (v9) records
30353	3373.22027	192.168.0.180	192.168.0.100	CFLOW	total: 1 (v9) record

Timestamp: Jan 5, 2011 11:04:17.000000000 (0.000000000)

FlowSequence: 6089

SourceId: 0

FlowSet 1

- Options FlowSet: 1
- FlowSet Length: 26
- Template Id: 256
- Option Scope Length: 4
- Option Length: 12
- Scope Type: System (1)
- Scope Field Length: 4
- Type: APPLICATION_ID (95)
- Length: 4
- Type: APPLICATION_NAME (96)
- Length: 24
- Type: APPLICATION_DESC (94)
- Length: 55

FlowSet 2

Data FlowSet (Template Id): 256

FlowSet Length: 961

Flow 1

- ScopeSystem: 192.168.0.180 (192.168.0.180)
- ApplicationID: 1223
- ApplicationName: sixtofour-ipv6-tunneled
- ApplicationDesc: 6 To 4 ipv6 tunneled

Flow 2

- ScopeSystem: 192.168.0.180 (192.168.0.180)
- ApplicationID: 244
- ApplicationName: custom-10
- ApplicationDesc: custom protocol custom-10

利用例) TemplateのApplication IDにNBARで認識したApplication IDが記録される
Application IDとApplication名のマッピングはOption Templateにより通知される

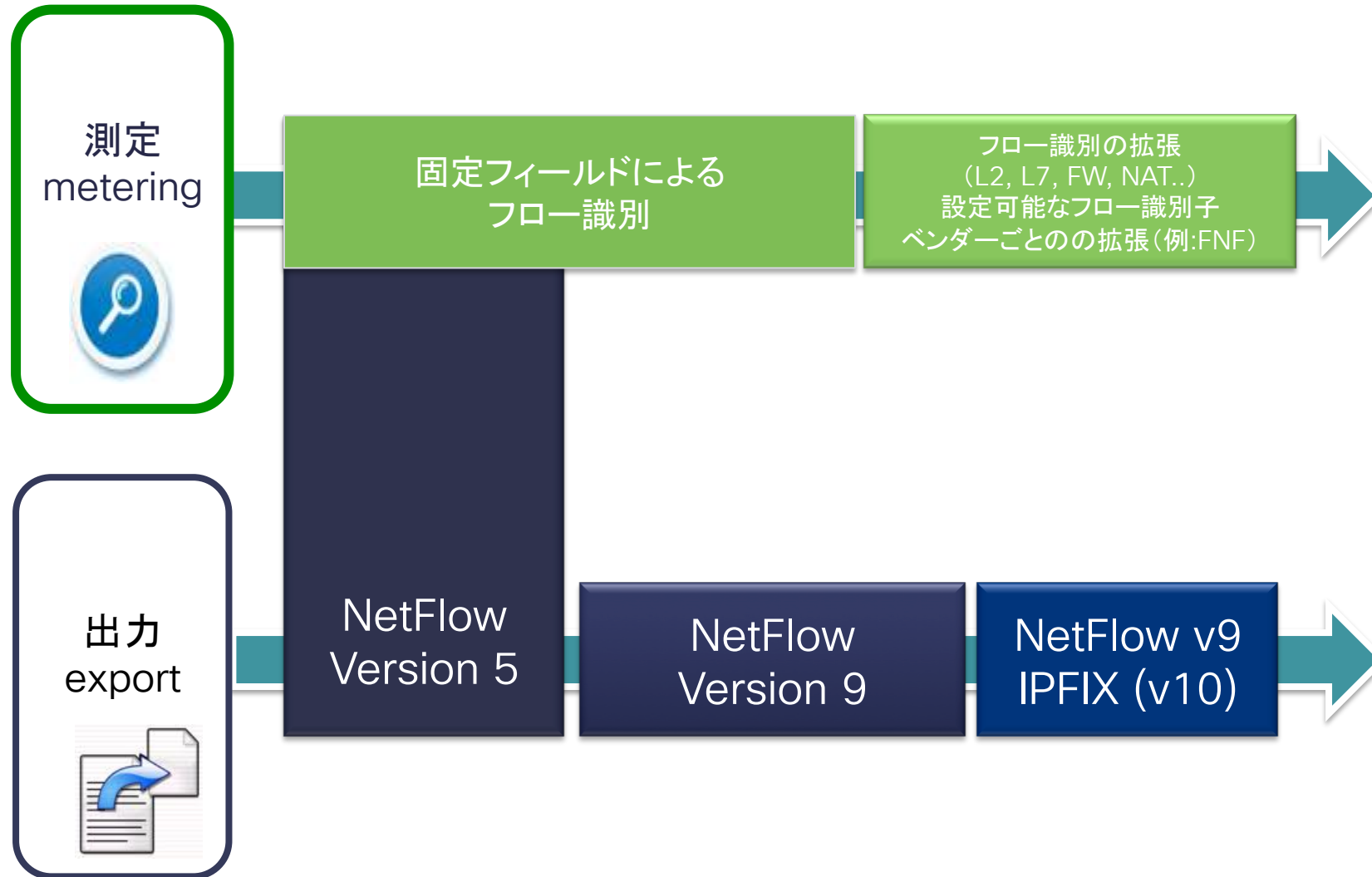
オプションテンプレートの実装例

```
C1941(config-flow-exporter)#option ?
  application-attributes  Application Attributes Table Option
  application-table       Application Table Option
  c3pl-class-table        C3PL class cce-id table
  c3pl-policy-table       C3PL policy cce-id table
  exporter-stats          Exporter Statistics Option
  interface-table         Interface SNMP-index-to-name Table Option
  metadata-version-table  Metadata Version Table Option
  sampler-table           Export Sampler Option
  vrf-table               VRF ID-to-name Table Option
```

```
Cat3850(config-flow-exporter)#option ?
  application-table       Application Table Option
  exporter-stats          Exporter Statistics Option
  interface-table         Interface SNMP-index-to-name Table Option
  metadata-version-table  Metadata Version Table Option
  sampler-table           Export Sampler Option
  usermac-table           Wireless usermac-to-username Table Option
```

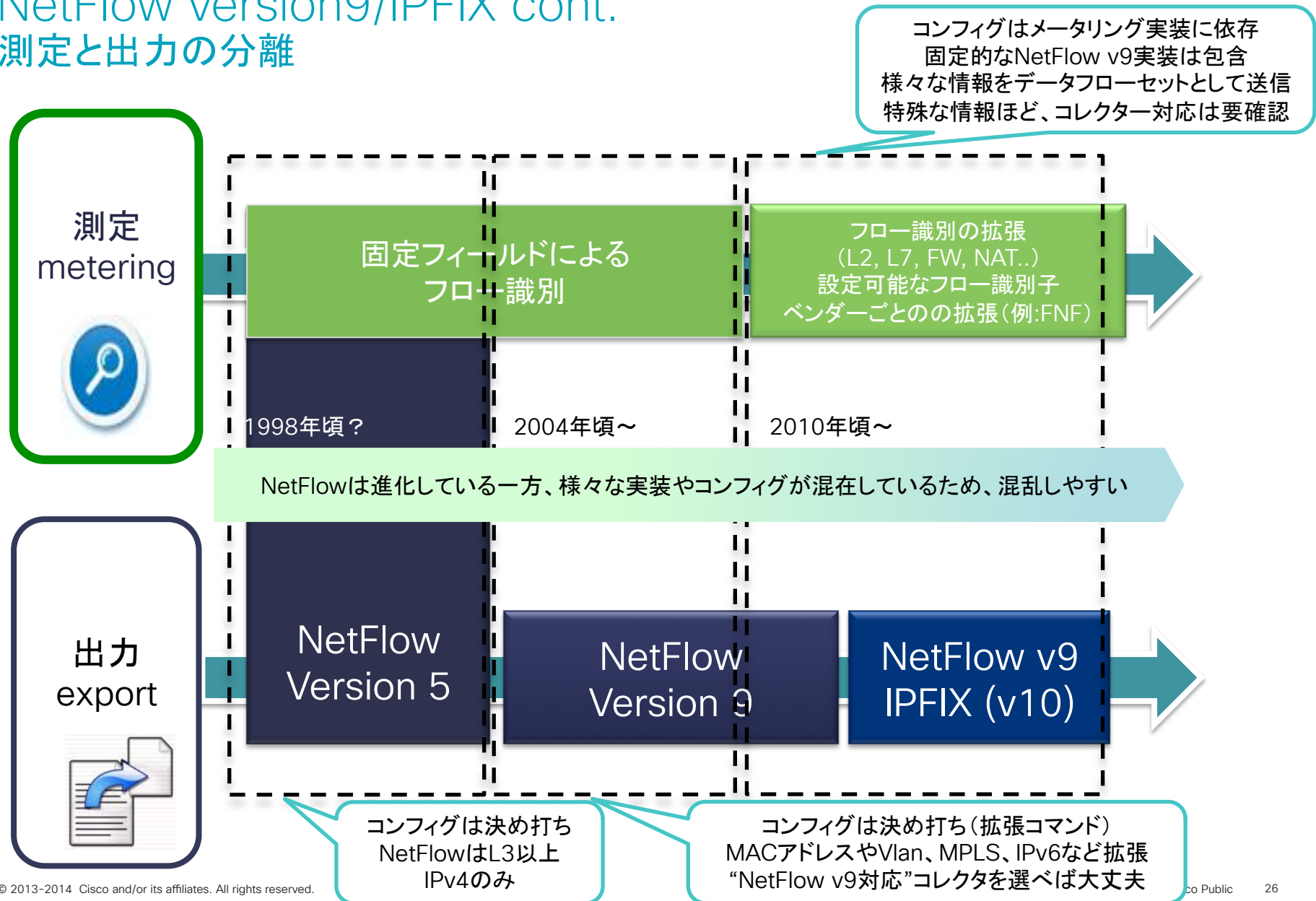
NetFlow version9/IPFIX

測定と出力の分離

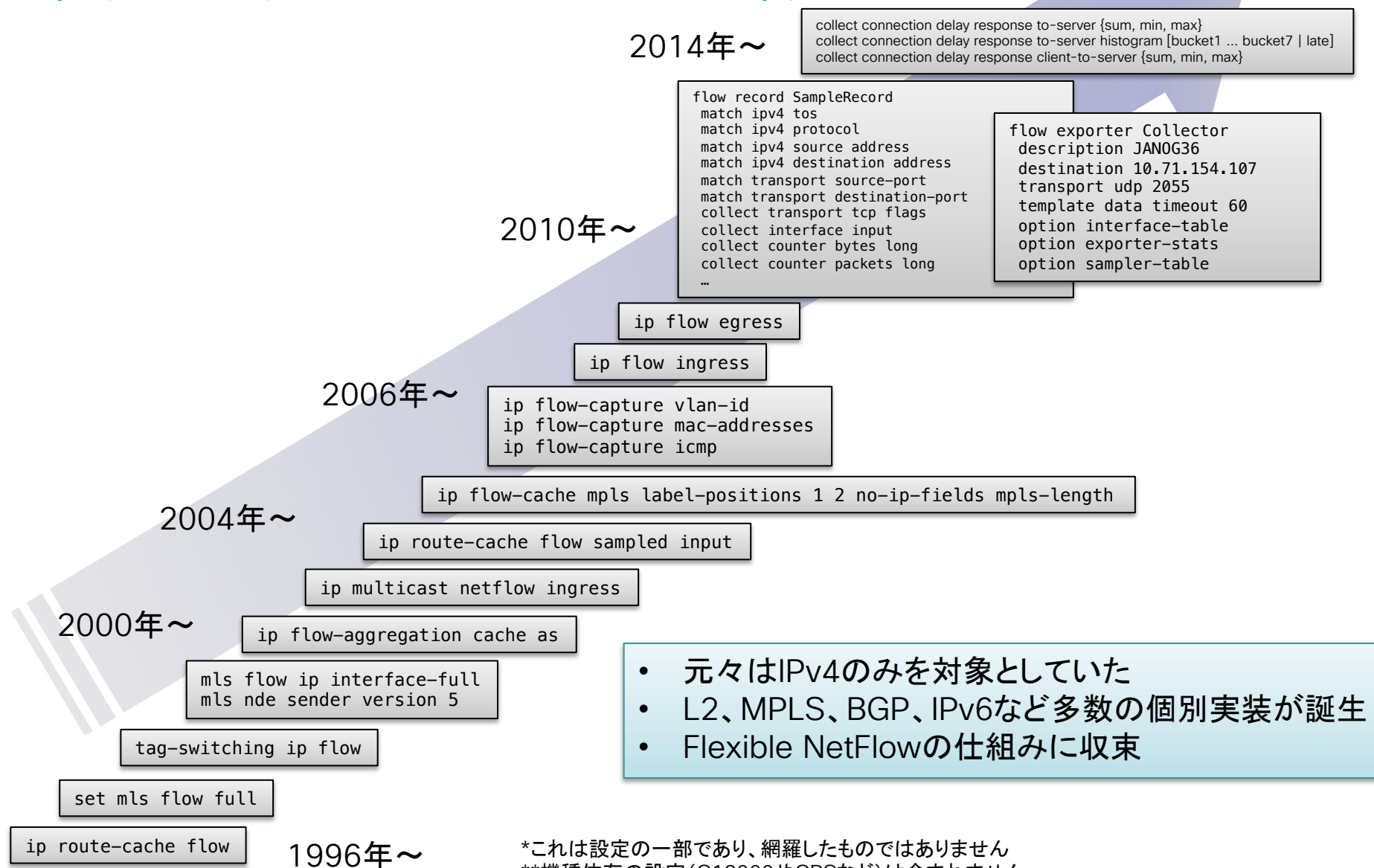


NetFlow version9/IPFIX cont.

測定と出力の分離



世代にみるCisco NetFlow コンフィグレーション



- 元々はIPv4のみを対象としていた
- L2、MPLS、BGP、IPv6など多数の個別実装が誕生
- Flexible NetFlowの仕組みに収束

*これは設定の一部であり、網羅したものではありません

**機種依存の設定(C12000やCRSなど)は含まれません

***同じコマンドでも装置によってサポート時期が前後する場合があります

世代にみるCisco NetFlow

インターフェースごとじゃなくて、計測対象をクラスマップで絞れるようになったんだよ

えっ、NBARやCBQoS-MIBまでフローで代替するの？

2014年～

```
collect connection delay response to-server {sum, min, max}
collect connection delay response to-server histogram [bucket1 ... bucket7 | late]
collect connection delay response client-to-server {sum, min, max}
```

まだIPFIXよりNetFlow v9の方が多いかな？

Record, Exporter, Monitorの仕組み、めっちゃ綺麗やん

```
flow record SampleRecord
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
collect transport tcp flags
collect interface input
collect counter bytes long
collect counter packets long
...
```

```
flow exporter Collector
description JAN0636
destination 10.71.154.107
transport udp 9999
template data timeout 60
option ip-src-label
option exporter-stats
option sampler-table
```

NATロギングやDPIロギングにも使われているんだよ

2010年～

12.2SXとか混迷の時代だったなあ...

ip flow egress

ip flow ingress

まだ ip flow ingress ってコマンドは、納得しないなあ

2006年～

```
ip flow-capture vlan-id
ip flow-capture mac-addresses
ip flow-capture icmp
```

egress に設定するとか、なんかキモい

```
label-positions 1 2 no-ip-fields mpls-length
```

小規模ルータでは使い物にならなかったなあ

```
ip route-cache flow sampled input
```

2004年～
CatalystはVlan取れるけど、TCPフラグはまだですか！？

```
ip multicast netflow ingress
```

Catalystはインターフェースごとに設定書けないよね？

2000年～

```
ip flow-aggregation cache as
```

```
mls flow ip interface-full
mls nde sender version 5
```

```
tag-switching ip flow
```

```
set mls flow full
```

setコマンドあったなあ

- 元々はIPv4のみを対象としていた
- L2、MPLS、BGP、IPv6など多数の個別実装が誕生
- Flexible NetFlowの仕組みに収束

古くはルーティングのキャッシュだったんだよ

1996年～

```
ip route-cache flow
```

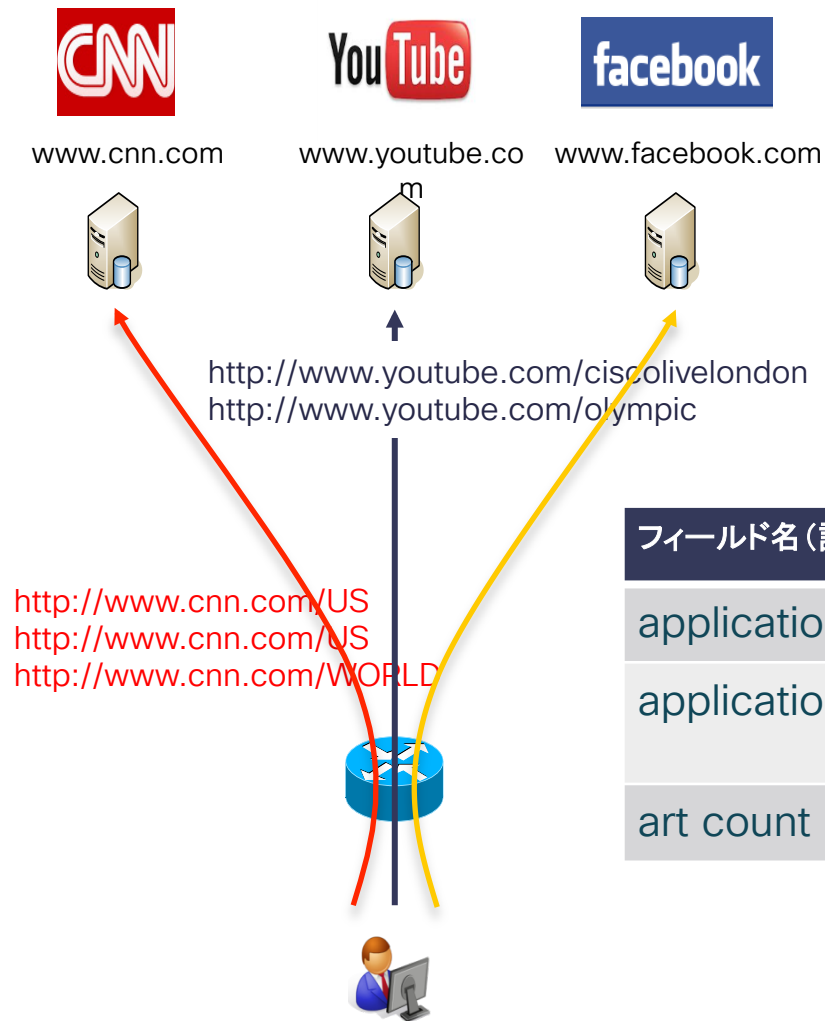
*これは設定の一部であり、網羅したものではありません
 **機種依存の設定(C12000やCRSなど)は含まれません
 ***同じコマンドでも装置によってサポート時期が前後する場合があります

IPFIX テンプレート例

```
▷ Internet Protocol Version 4, Src: 10.0.0.254 (10.0.0.254), Dst: 10.100.1.128 (10.100.1.128)
▷ User Datagram Protocol, Src Port: 58390 (58390), Dst Port: iop (2055)
▼ Cisco NetFlow/IPFIX
  Version: 10
  Length: 380
  ▷ Timestamp: Apr 23, 2013 18:05:54.000000000 JST
  FlowSequence: 16354
  Observation Domain Id: 0
  ▼ Set 1
    FlowSet Id: Data Template ((V10 [IPFIX])) (2)
    FlowSet Length: 364
    ▼ Template (Id = 258, Count = 48)
      Template Id: 258
      Field Count: 48
      ▼ Field (1/48): IP_SRC_ADDR
        0... .. = Pen provided: No
        .000 0000 0000 1000 = Type: IP_SRC_ADDR (8)
        Length: 4
        ▷ Field (2/48): IP_DST_ADDR
        ▷ Field (3/48): L4_DST_PORT
        ▷ Field (4/48): PROTOCOL
        ▷ Field (5/48): 9252 [pen: ciscoSystems]
        ▷ Field (6/48): 9357 [pen: ciscoSystems]
        ▷ Field (7/48): 12235 [pen: ciscoSystems]
        ▷ Field (8/48): APPLICATION_ID
```

IPFIX 利用例

URLホストとヒットカウント



- アクセス頻度の高いWebサイトは？
- サイト内のURLは？

フィールド名 (設定例)	フィールドID	値
application http host	45003	www.cnn.com
application http uri statistics	42125	US\02 WORLD\01*
art count new connections	42050	3

*\0はデリミタ、NetFlowコレクターがParse

```

Flow 1
Flow 2
  SrcAddr: 10.71.154.14 (10.71.154.14)
  DstAddr: 72.163.10.14 (72.163.10.14)
  DstPort: 80
  Protocol: 6
  Enterprise Private entry: (ciscoSystems) Type 9252: Value (hex bytes): 00
  [Enterprise Private entry: (ciscoSystems) Type 9357: Value (hex bytes): 74 61 67 00 00 03 (Variable Length)]
    String_len_short: 255
    String_len_short: 6
  [Enterprise Private entry: (ciscoSystems) Type 12235: Value (hex bytes): 03 00 00 50 34 02 6e 65 77 73 2d 74 61 67 73 2e ... (Variable Length)]
    String_len_short: 255
    String_len_short: 25
  DSCP: 0
  ApplicationID: NBAR Application ID: 3:80 (type:id)
  Enterprise Private entry: (ciscoSystems) Type 9303: Value (hex bytes): 00 00 01 c8
  Enterprise Private entry: (ciscoSystems) Type 9305: Value (hex bytes): 00 00 00 98
  Enterprise Private entry: (ciscoSystems) Type 9304: Value (hex bytes): 00 00 00 98
  Enterprise Private entry: (ciscoSystems) Type 9306: Value (hex bytes): 00 00 00 09
  Enterprise Private entry: (ciscoSystems) Type 9308: Value (hex bytes): 00 00 00 03
  Enterprise Private entry: (ciscoSystems) Type 9307: Value (hex bytes): 00 00 00 03
  Enterprise Private entry: (ciscoSystems) Type 9313: Value (hex bytes): 00 00 02 38
  Enterprise Private entry: (ciscoSystems) Type 9315: Value (hex bytes): 00 00 00 b8
  Enterprise Private entry: (ciscoSystems) Type 9314: Value (hex bytes): 00 00 00 c4
  Enterprise Private entry: (ciscoSystems) Type 9316: Value (hex bytes): 00 00 00 78
  Enterprise Private entry: (ciscoSystems) Type 9318: Value (hex bytes): 00 00 00 24
  Enterprise Private entry: (ciscoSystems) Type 9317: Value (hex bytes): 00 00 00 30
  Enterprise Private entry: (ciscoSystems) Type 9319: Value (hex bytes): 00 00 01 c0
  Enterprise Private entry: (ciscoSystems) Type 9321: Value (hex bytes): 00 00 00 94
  Enterprise Private entry: (ciscoSystems) Type 9320: Value (hex bytes): 00 00 00 98
  Enterprise Private entry: (ciscoSystems) Type 9309: Value (hex bytes): 00 00 02 40
  Enterprise Private entry: (ciscoSystems) Type 9311: Value (hex bytes): 00 00 00 bc
  Enterprise Private entry: (ciscoSystems) Type 9310: Value (hex bytes): 00 00 00 c8
  Enterprise Private entry: (ciscoSystems) Type 9273: Value (hex bytes): 00 00 00 03
  Enterprise Private entry: (ciscoSystems) Type 9275: Value (hex bytes): 00 00 00 03
  Enterprise Private entry: (ciscoSystems) Type 9274: Value (hex bytes): 00 00 00 00
  Enterprise Private entry: (ciscoSystems) Type 9272: Value (hex bytes): 00 00 00 00
  Enterprise Private entry: (ciscoSystems) Type 9268: Value (hex bytes): 00 00 00 00
  Enterprise Private entry: (ciscoSystems) Type 9282: Value (hex bytes): 00 00 00 00
  Enterprise Private entry: (ciscoSystems) Type 9282: Value (hex bytes): 00 00 00 00
0130 ff 00 06 74 61 67 00 00 03 ff 00 19 03 00 00 50 ...tag... ..P
0140 34 02 6e 65 77 73 2d 74 61 67 73 2e 63 69 63 63 4.news-t ags.cisc
0150 6f 2e 63 6f 6d 00 03 00 00 50 00 00 01 c8 00 00 0.com... .P.....
0160 00 98 00 00 00 98 00 00 00 09 00 00 03 00 00 .....
0170 00 03 00 00 02 38 00 00 00 b8 00 00 00 c4 00 00 .....8..

```

可変長のため、IPFIXが必須
(NetFlow v9は不可)

```

Enterprise Private entry: (ciscoSystems) Type 9282: Value (hex bytes): 00 00 00 03
Enterprise Private entry: (ciscoSystems) Type 9292: Value (hex bytes): 00 00 00 03
Enterprise Private entry: (ciscoSystems) Type 9300: Value (hex bytes): 00 00 00 00
Enterprise Private entry: (ciscoSystems) Type 9293: Value (hex bytes): 00 00 00 00
Enterprise Private entry: (ciscoSystems) Type 9294: Value (hex bytes): 00 00 00 00
Enterprise Private entry: (ciscoSystems) Type 9295: Value (hex bytes): 00 00 00 00
Enterprise Private entry: (ciscoSystems) Type 9296: Value (hex bytes): 00 00 00 00
Enterprise Private entry: (ciscoSystems) Type 9297: Value (hex bytes): 00 00 00 00
Enterprise Private entry: (ciscoSystems) Type 9298: Value (hex bytes): 00 00 00 03
Enterprise Private entry: (ciscoSystems) Type 9299: Value (hex bytes): 00 00 00 00
InputInt: 3
OutputInt: 2
Post Octets: 1848
Post Packets: 15
Enterprise Private entry: (ciscoSystems) Type 9266: Value (hex bytes): 00 00 00 00 00 00 00 00 of
Responder Octets: 1224
Enterprise Private entry: (ciscoSystems) Type 9265: Value (hex bytes): 00 00 00 00 00 00 00 00 of
Initiator Octets: 5577
Octets: 6453
Packets: 21
Flow 3
ff 00 06 74 61 67 00 00 03 ff 00 19 03 00 00 50 ...tag... ..P
34 02 6e 65 77 73 2d 74 61 67 73 2e 63 69 63 63 4.news-t ags.cisc
6f 2e 63 6f 6d 00 03 00 00 50 00 00 01 c8 00 00 0.com... .P.....
00 98 00 00 00 98 00 00 00 09 00 00 03 00 00 .....
00 03 00 00 02 38 00 00 00 b8 00 00 00 c4 00 00 .....8..

```

Tips ... 時間があれば

Tips. 今風なやり方でVlanごとのトラフィック量を測る

設定バリエーション例

```
Catalyst3850 (config-flow-record)#match ?
  application  Application fields
  datalink     Datalink (layer2) fields
  flow         Flow identifying fields
  interface    Interface fields
  ipv4         IPv4 fields
  ipv6         IPv6 fields
  transport    Transport layer fields
  wireless     Wireless fields

Cat3850-01(config-flow-record)#match datalink ?
  dot1q        dot1q field
  ethertype     The Ethertype of the packet
  mac          MAC fields
  vlan         The VLAN the packet is on
```

設定例

```
flow record vlanRec
match datalink dot1q vlan input
match interface input
collect counter bytes
collect counter packets
collect timestamp sys-uptime first
collect timestamp sys-uptime last
!
flow monitor JANOG36
record vlanRec
!
interface GigabitEthernet1/10
switchport mode trunk
vlan-range 1-2000
  datalink flow monitor JANOG36 input
```

※exporterは設定していません

出力例

```
4503E#sh flow mon JANOG36 cache format table
Cache type: Permanent
Cache size: 4096
Current entries: 2
High Watermark: 2
Flows added: 2
Updates sent ( 1800 secs) 0
=====
DATA LINK DOT1Q VLAN INPUT INTF INPUT time first time last bytes long perm pkts long perm
=====
10 Gi2/3 15:54:47.562 16:24:41.562 8864352 8706
1 Gi2/3 16:04:07.562 16:23:51.566 243 3
```

Tips. 用途に応じて、複数の宛先にNDEを出力する(1/3)

QoSモニタリング

```
flow record QOS_Rec
match ipv4 tos
match interface input
match interface output
collect counter bytes
collect counter packets
```

IPv6フロー

```
ipv6 cef
!
flow record IPv6_Rec
description Used for basic IPv6 traffic analysis
match ipv6 destination address
match ipv6 source address
match ipv6 protocol
match transport source-port
match transport destination-port
collect counter bytes
collect counter packets
collect timestamp sys-uptime first
collect timestamp sys-uptime last
```

TCPフラグ & TTLモニター

```
flow record TCP_Rec
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport tcp flags
match ipv4 ttl
collect counter bytes
collect counter packets
collect timestamp sys-uptime first
collect timestamp sys-uptime last
```

VRFごと

```
flow record VRF_Rec
match routing vrf input
match ipv4 source address
match ipv4 destination address
collect interface input
collect interface output
collect counter bytes
collect counter packets
collect timestamp sys-uptime first
collect timestamp sys-uptime last
```

従来の7タプル

```
flow record TNF_Rec
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
match interface input
collect counter bytes
collect counter bytes
collect counter packets
collect timestamp sys-uptime first
collect timestamp sys-uptime last
```

Tips. 用途に応じて、複数の宛先にNDEを出力する(2/3)

コレクター1

```
flow exporter my-exporter1
description GeneralCollector
destination 10.101.41.24
source GigabitEthernet0/1
transport udp 2055
```

コレクター2

```
flow exporter my-exporter2
description SecurityCollector
destination 10.101.41.25
source GigabitEthernet0/1
transport udp 9996
option interface-table
```

コレクター3

```
flow exporter my-exporter3
description IPv6_Collector
destination 10.101.41.26
source GigabitEthernet0/1
transport udp 3000
option interface-table
option exporter-stats
option application-table
```

コレクター4

```
flow exporter my-exporter4
description TestCollector
destination 10.101.41.27
source GigabitEthernet0/1
transport udp 2055
option interface-table
option exporter-stats
option application-table
```

各レコードとエクスポートを 紐づけたモニターを作成

```
flow monitor TNF_Mon
record TNF_Rec
exporter my-exporter1
exporter my-exporter4
cache timeout active 1
!
interface GigabitEthernet0/1
ip address 10.0.0.254 255.255.255.0
ip flow monitor TNF_Mon input
```

```
flow monitor Security_Mon
record TCP_Rec
exporter my-exporter2
cache timeout active 1
!
interface GigabitEthernet0/24
ip address 10.71.15.254 255.255.255.0
ip flow monitor Security_Mon input
```

```
flow monitor IPv6_Mon
record IPv6_Rec
exporter my-exporter3
cache timeout active 1
!
interface GigabitEthernet0/24
ip address 10.180.1.25 255.255.255.0
ip flow monitor Security_Mon input
```

Tips. 用途に応じて、複数の宛先にNDEを出力する(3/3)

装置内で、いろいろなモニタリングが楽しめる！

ToSモニター

INTF INPUT	INTF OUTPUT	IP TOS	bytes long perm	pkts long perm
=====	=====	=====	=====	=====
Gi0/1	Gi0/2	0x00	1002890100	1365382
Gi0/2	Gi0/1	0x00	263579575	1457850
Gi0/2	Gi0/1	0xC0	546068	8873
Gi0/2	Gi0/2	0x00	19145460	246939
Gi0/2	Gi0/2	0xC0	9348	123
Gi0/1	Gi0/1	0x00	383384	5279

TCPフラグモニター

IPv4 SRC ADDR	IPv4 DST ADDR	TCP FLAGS	IP PROT	bytes	pkts	time first	time last
=====	=====	=====	=====	=====	=====	=====	=====
219.75.253.70	10.1.100.33	0x00	17	76	1	08:38:49.278	08:38:49.278
64.68.120.146	10.101.10.74	0x18	6	480	2	08:38:51.766	08:38:50.062
202.234.232.6	10.101.9.251	0x00	17	210	2	08:38:50.262	08:38:50.274
118.243.82.177	10.1.100.33	0x00	17	76	1	08:38:51.274	08:38:51.274
219.75.253.70	10.1.100.33	0x00	17	76	1	08:38:51.278	08:38:51.278
10.100.1.90	10.101.41.11	0x02	6	48	1	08:38:53.666	08:38:53.666
10.100.1.90	10.101.41.11	0x10	6	40	1	08:38:53.670	08:38:53.670
10.100.1.90	10.101.41.11	0x18	6	767	1	08:38:53.674	08:38:53.674
10.100.1.90	10.101.41.11	0x11	6	40	1	08:38:53.686	08:38:53.686
192.221.72.51	10.1.100.175	0x00	17	314	1	08:38:53.942	08:38:53.942

0x02 ... Syn
0x10 ... Ack
0x11 ... FIN, Ack
0x12 ... Syn, Ack
0x18 ... PUSH, Ack

アプリケーションモニター

IPv4 SRC ADDR	IPv4 DST ADDR	INTF INPUT	INTF OUTPUT	IP PROT	APP NAME	bytes	pkts	time first	time last
=====	=====	=====	=====	=====	=====	=====	=====	=====	=====
125.206.198.59	10.1.100.33	Gi0/1	Gi0/2	17	nbar ntp	76	1	08:26:48.206	08:26:48.206
10.100.1.92	10.101.40.252	Gi0/1	Gi0/2	6	nbar xwindows	64	1	08:26:51.938	08:26:51.938
10.100.1.90	10.101.41.11	Gi0/1	Gi0/2	6	nbar unknown	216	5	08:26:52.014	08:26:54.494
10.100.1.90	10.101.41.11	Gi0/1	Gi0/2	6	nbar http	1578	6	08:26:52.018	08:26:54.802
64.68.120.146	10.101.10.74	Gi0/1	Gi0/2	6	nbar secure-http	560	4	08:26:54.530	08:26:54.830
174.129.220.171	10.101.90.99	Gi0/1	Gi0/2	6	nbar http	65	1	08:26:53.238	08:26:53.238
10.100.1.92	10.101.40.252	Gi0/1	Gi0/2	6	nbar vdlive	64	1	08:29:02.938	08:29:02.938
10.100.1.92	10.101.40.252	Gi0/1	Gi0/2	6	nbar h323	384	6	08:29:02.942	08:29:03.430
10.100.1.92	10.101.40.252	Gi0/1	Gi0/2	6	nbar skinny	192	3	08:29:02.950	08:29:01.130
10.100.1.92	10.101.40.252	Gi0/1	Gi0/2	6	nbar nfs	64	1	08:29:02.970	08:29:02.970
10.100.1.92	10.101.40.252	Gi0/1	Gi0/2	6	nbar corba-iiop	64	1	08:29:02.978	08:29:02.978
10.100.1.92	10.101.40.252	Gi0/1	Gi0/2	6	nbar appletc	64	1	08:29:02.994	08:29:02.994

- コンフィグを作り込んでみたら面白そう
- コレクター側はエクスポートから、いろいろ飛んでくるから、なかなか大変
- オープンソースのツールで受信、データは自分用に加工したり..
- 装置内でカウンター代わりに使ってみたり..

それでは、後半パートへ...

Thank you.

