

Nino Ciurleo (GARR)

Tutorial Netflow

dall'esportazione dei flussi alla elaborazione avanzata delle statistiche

- Perche' monitorare la rete con Netflow
- Protocolli per l'esportazione dei flussi Netflow
- Architettura del sistema di analisi dei flussi
- Suite Nfsen/Nfdump
- Esempi di utilizzo di Nfsen/Nfdump
- Monitoraggio della LAN con Netflow
- Estendere le funzionalita' di Nfsen/Nfdump

Perche' usare l'esportazione dei flussi?



- Limiti delle statistiche di traffico tradizionali (MRTG, Cacti, etc.)
- Possibilità offerte dall'analisi dei flussi

Limiti delle statistiche di traffico tradizionali



- Il monitoring tradizionale usa come sorgente d'informazione i contatori dei router. Esempio: numero di pacchetti e ottetti.
- Le informazioni collezionate sono relative alle interfacce e rappresentano il traffico aggregato in transito su di esse.
- Non e' possibile analizzare il traffico in base alle caratteristiche delle comunicazioni quali i protocolli utilizzati, le subnet, le porte di livello di trasporto, etc.

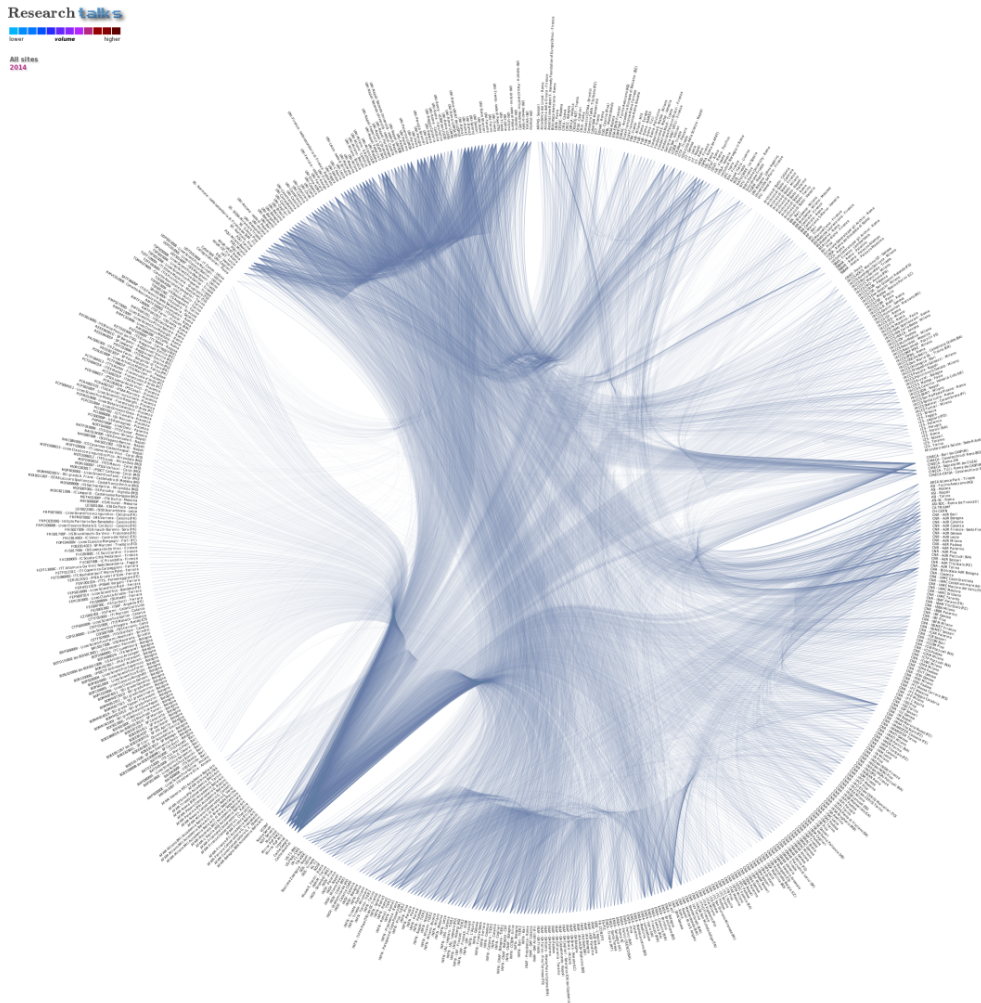
- L'analisi dei flussi prevede che ogni router invii le intestazioni dei pacchetti che lo attraversano ad un collettore.
- Informazioni del livello IP, di quello di trasporto (TCP/UDP): indirizzi IP, protocolli, porte
- In piu': AS, interfacce, contatori, etc.
- Le informazioni raccolte consentono di analizzare dettagliatamente l'attività di rete presente e passata.

Come lo usa GARR? Research talks



Matrice di traffico
tra sedi GARR

Matrice di tra sedi
GARR e AS

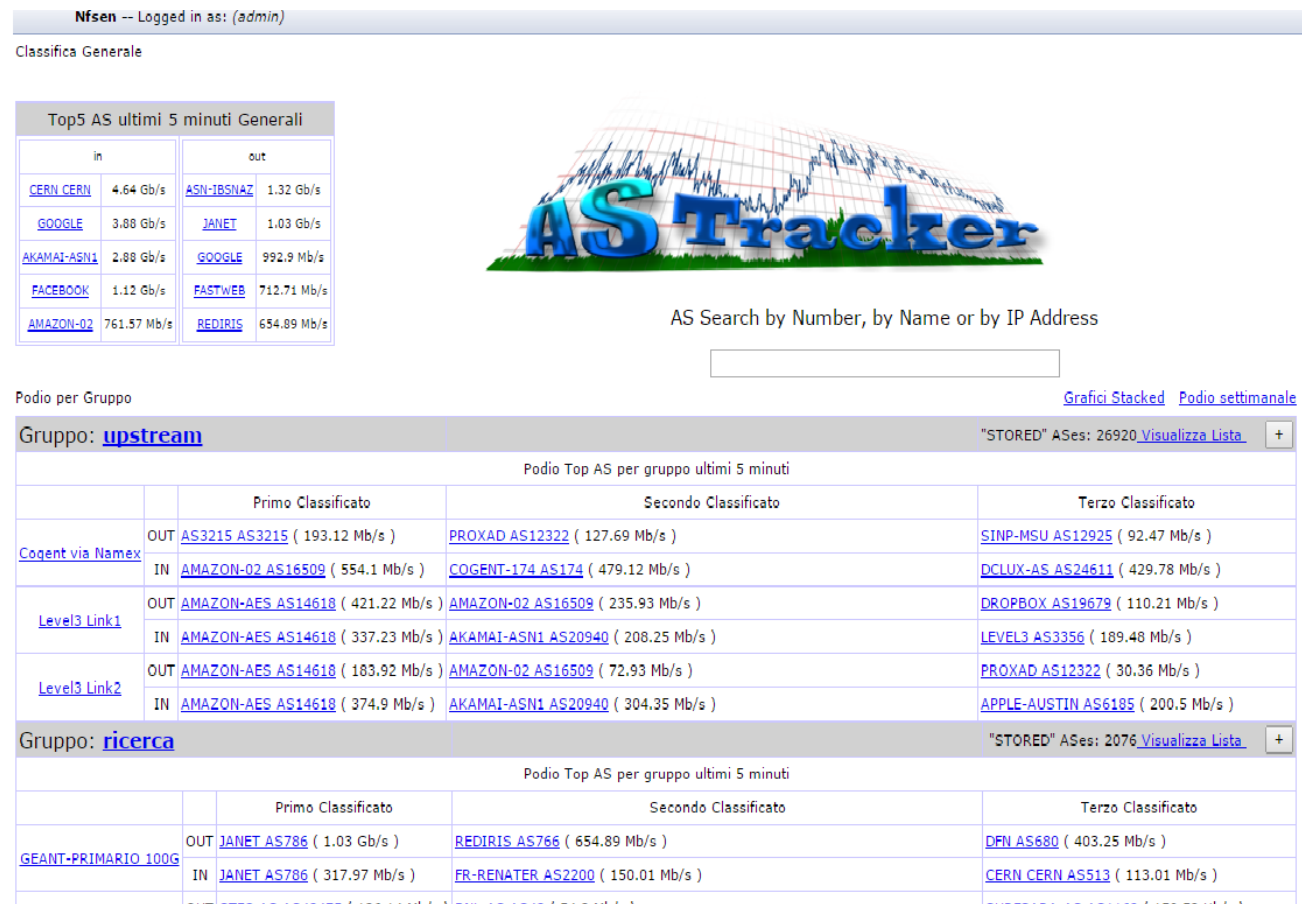


Come lo usa GARR? AsTracker

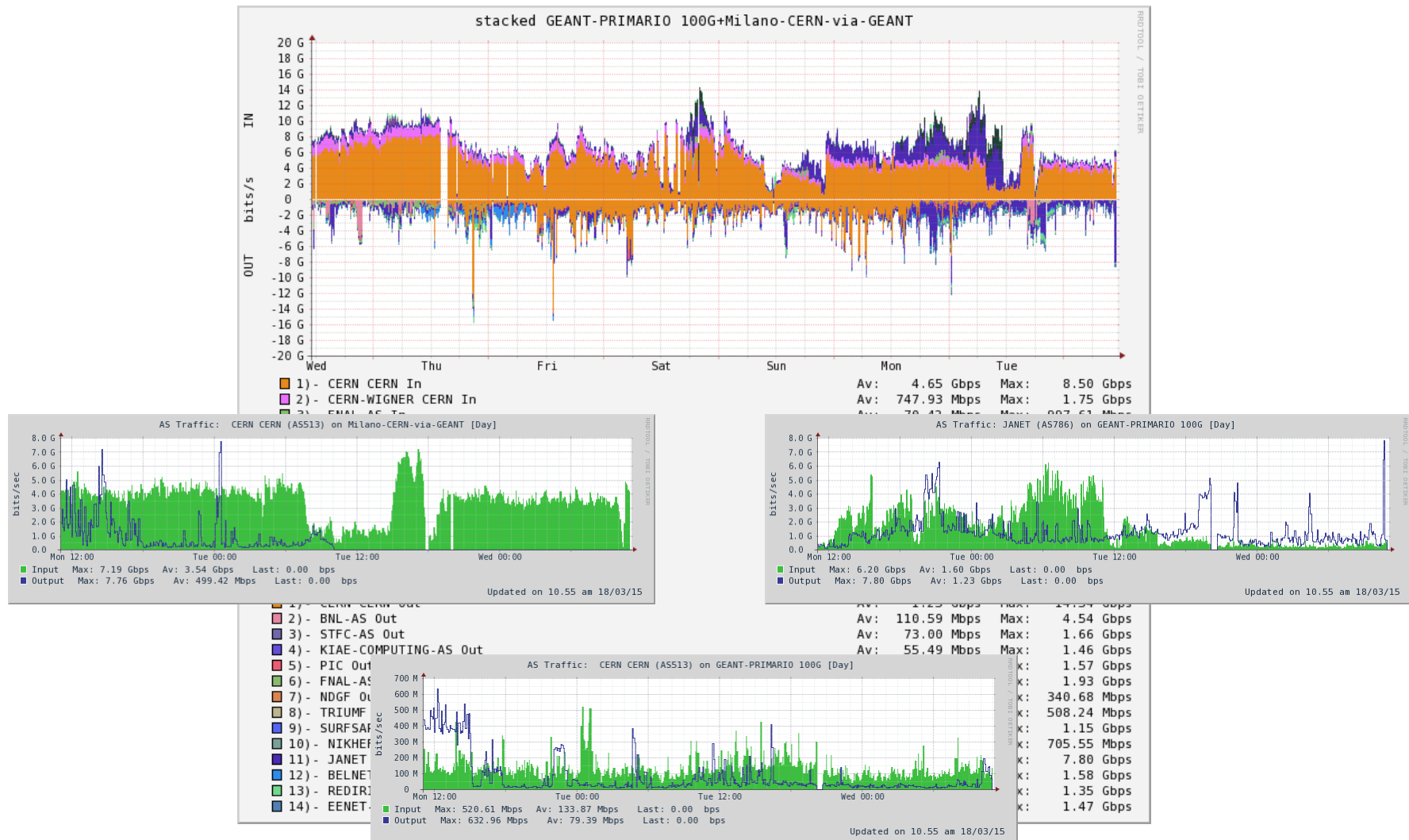
Statistiche di traffico
tra la rete GARR e
tutti gli altri
Autonomous System
di Internet

- Troubleshooting
politiche di peering
BGP

- Analisi categorie di
traffico (ricerca,
commerciale, peering
nazionali, etc)



Come lo usa GARR? AsTracker

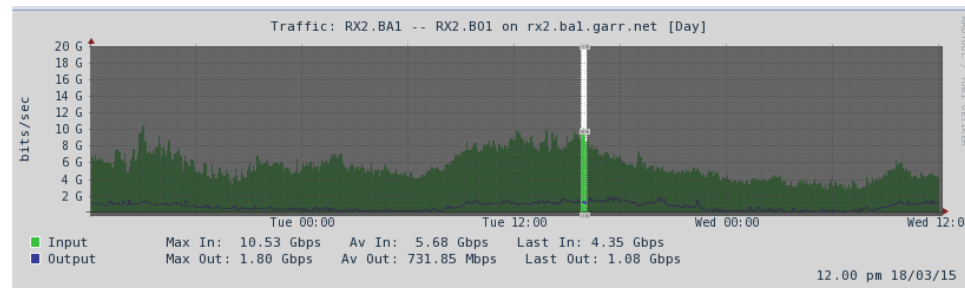


Come lo usa GARR? GINS



Analisi dettagliata
del traffico su tutti i
link della rete
(backbone, peering
e accesso):

- Incidenti di sicurezza
- Anomalie di traffico
- Troubleshooting



Select an area on the graph:

Start Date: Tue Mar 17 2015 15:27:06 GMT+0100 (W. Europe Standard Time)

End Date: Tue Mar 17 2015 15:43:04 GMT+0100 (W. Europe Standard Time)

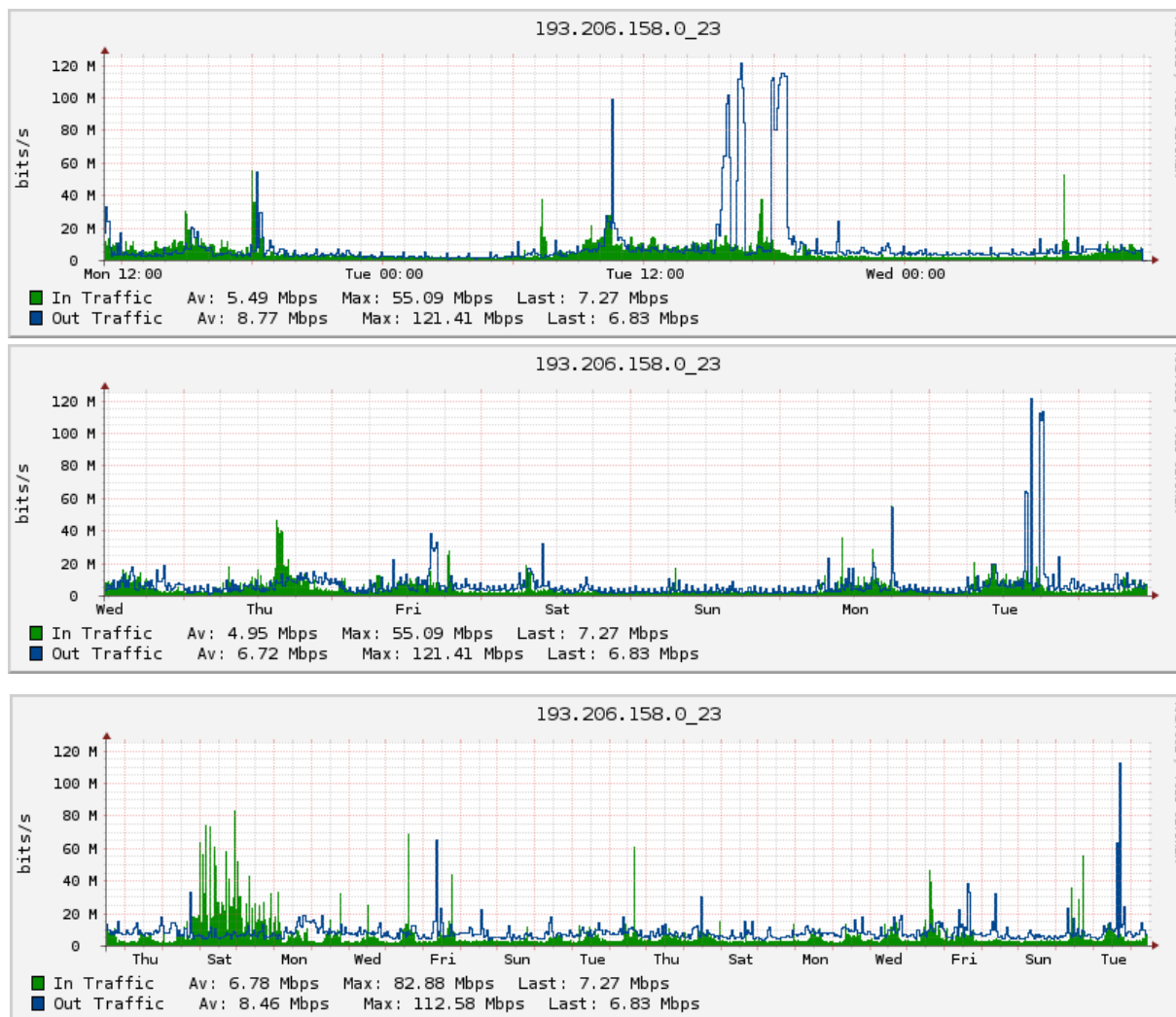
[get flow details](#) [get flow list](#) [get flow list no net](#) [get sites details](#)

Posizione	Sorgente	Destinazione	Traffico	% sul totale: 1.42 TB
1	INFN - CNAF - Bologna - TIER1	INFN - Bari - TIER2	500.12 GB	34.36%
2	FNAL-AS(AS3152)	INFN - Bari - TIER2	195.78 GB	13.45%
3	Google-Cache	UNI-Torino	26.8 GB	1.84%
4	Akamai-Cache	UNI-Napoli Federico II	18.87 GB	1.3%
5	Akamai-Cache	UNI-Torino	17.04 GB	1.17%
6	Google-Cache	UNI-Napoli Federico II	16.65 GB	1.14%
7	Akamai-Cache	UNI-Salerno	14.82 GB	1.02%
8	Akamai-Cache	UNI-Roma3	13.85 GB	0.95%
9	Google-Cache	POLI-Torino	13.39 GB	0.92%
10	Google-Cache	UNI-Salerno	12.3 GB	0.85%
11	INFN - CNAF - Bologna - TIER1	INFN - Napoli - TIER2	12.06 GB	0.83%
12	Akamai-Cache	UNI-Palermo	11.92 GB	0.82%
13	Akamai-Cache	POLI-Torino	11.18 GB	0.77%

Come lo usa GARR? Net stats

Statistiche di traffico
per network

Ogni rete gestita da
GARR ha statistiche
aggregate
indipendenti dai
circuiti su cui passa il
traffico



Come lo usa GARR? Rilevamento DOS



----- SUMMARY -----

90.147.102.0/24 INFN_-_Bari_-_TIER2 573.7 M

===== 90.147.102.0/24 =====

Src Network	Dst IP Addr:Port	Dst AS	bps	Bytes	Packets
90.147.102.0/24	162.218.53.18:80	19905	583.2 M	62.4 G	66.7 M
90.147.102.0/24	14.152.83.24:1688	58543	545.7 M	43.3 G	46.3 M
90.147.102.0/24	219.135.58.29:80	58543	574.5 M	33.5 G	35.8 M
90.147.102.0/24	14.152.82.125:81	58543	608.1 M	30.9 G	33.0 M
90.147.102.0/24	72.14.246.1:80	15169	607.0 M	26.0 G	27.8 M
90.147.102.0/24	101.71.48.190:80	4837	582.5 M	13.3 G	14.2 M
90.147.102.0/24	50.117.92.22:1688	18779	134.4 M	12.5 G	13.3 M
90.147.102.0/24	115.231.11.78:1688	4134	130.9 M	12.0 G	12.9 M

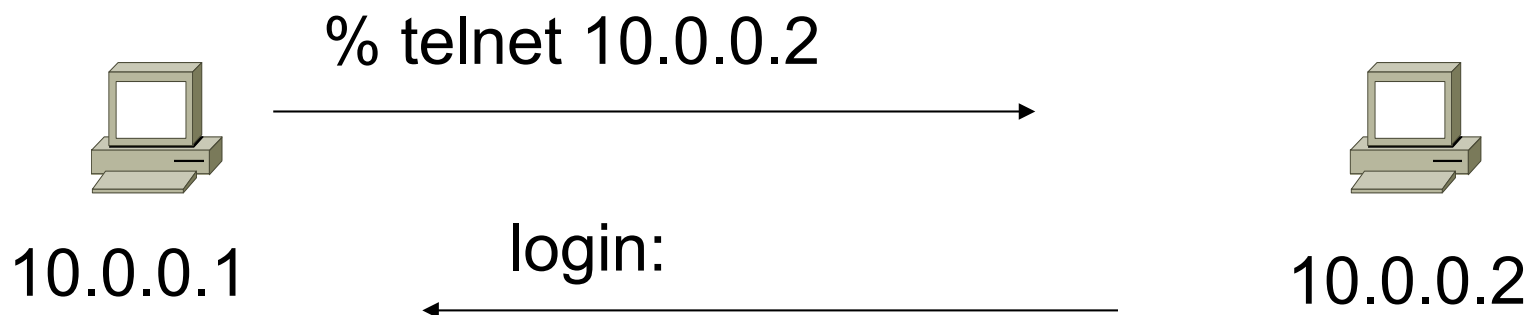
- Concetto di flusso
- Versioni del protocollo Netflow
 - Netflow 5
 - Netflow 9
 - IPFIX

Cos'è un flusso Netflow?



- Per flusso si intende ogni comunicazione unidirezionale identificata da 7 campi degli header IP e UDP/TCP (valido per Netflow 5)
- Vengono raggruppati nello stesso flusso i pacchetti che hanno in comune:
 - Indirizzo IP sorgente
 - Indirizzo IP destinazione
 - porte sorgente
 - porte destinazione
 - Protocollo
 - Interfaccia di ingresso
 - ToS

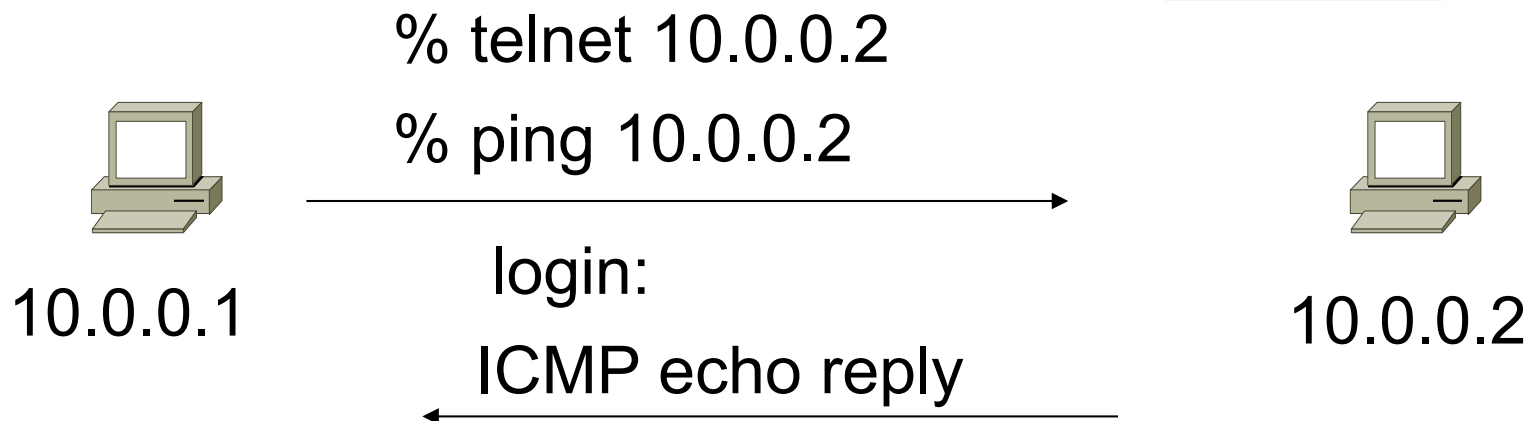
Esempi di flusso (1)



Active Flows

Flow	Source IP	Destination IP
1	10.0.0.1	10.0.0.2
2	10.0.0.2	10.0.0.1

Esempi di flusso (2)



Active Flows

Flow	Source IP	Destination IP	prot	srcPort	dstPort
1	10.0.0.1	10.0.0.2	TCP	32000	23
2	10.0.0.2	10.0.0.1	TCP	23	32000
3	10.0.0.1	10.0.0.2	ICMP	0	0
4	10.0.0.2	10.0.0.1	ICMP	0	0

Protocolli per l'esportazione dei flussi



- Netflow, inizialmente sviluppato da Cisco, e' diventato uno standard "de facto" ed implementato da gran parte dei costruttori di hardware (c/jflowd su Juniper, Cflowd su Alcatel, NetStream su Huawei).
- Versioni:
 - V5 la versione supportata da quasi tutti i vendor e ancora la piu' utilizzata
 - V7 per gli switch catalyst 5000
 - V8 possibilità di esportare flussi aggregati
 - V9 la versione piu' recente e flessibile (RFC 3954)
 - possibilità di definire template personalizzati
 - Trasporto di informazioni di livello2, IPV6, MPLS, BGP, protocol next_hop, etc
 - IPFIX (Internet Protocol Flow Information eXport) standardizzazione IETF (RFC 5101 e 5102) di NetFlow v9

Netflow versione 5



- Supporta solamente IPv4
- Formato dei record Netflow 5:
 - IP sorgente e destinazione
 - porte sorgente e destinazione
 - interfaccia d'ingresso e di uscita
 - AS number sorgente e destinazione
 - TCP flags
 - ToS (DSCP)
 - Contatori di ottetti e pacchetti

Netflow versione 9



- Supporta IPv4, IPv6 ed MPLS
- informazioni trasportate sono:
 - IP sorgente e destinazione
 - porte sorgente e destinazione
 - interfaccia d'ingresso e di uscita
 - AS number sorgente e destinazione
 - Indirizzo IP "Next-Hop"
 - BGP "Next-hop"
 - TCP flags
 - ToS (DSCP)
 - Contatori di ottetti e pacchetti
 - Direzione del flusso
 - Indirizzo MAC sorgente e destinazione in ingresso
 - Indirizzo MAC sorgente e destinazione in uscita
 - Label VLAN
 - Label MPLS

- Standardizzazione di Netflow v9
- RFC7011, RFC5101, RFC5153
- Meccanismo Template flessibile

+-----+-----+-----+-----+-----+-----+-----+-----+							
	+-----+		+-----+		+-----+		
Message		Template		Data		Options	
Header		Set		Set	...	Template	
						Set	
	+-----+		+-----+		+-----+		
+-----+-----+-----+-----+-----+-----+-----+-----+							

+-----+-----+-----+-----+-----+-----+-----+-----+							
	+-----+		+-----+		+-----+		
Message		Data		Data		Data	
Header		Set	...	Set	...	Set	
	+-----+		+-----+		+-----+		
+-----+-----+-----+-----+-----+-----+-----+-----+							

+-----+-----+-----+-----+-----+-----+-----+-----+							
	+-----+		+-----+		+-----+		
Message		Template		Template		Options	
Header		Set	...	Set	...	Template	
						Set	
	+-----+		+-----+		+-----+		
+-----+-----+-----+-----+-----+-----+-----+-----+							

■ Esempio Template

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Set ID = 2          |          Length = 28 octets          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|      Template ID 256        |      Field Count = 5                 |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0|    sourceIPv4Address = 8    |      Field Length = 4               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0| destinationIPv4Address = 12 |      Field Length = 4               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0|  ipNextHopIPv4Address = 15  |      Field Length = 4               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0|    packetDeltaCount = 2     |      Field Length = 4               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0|    octetDeltaCount = 1      |      Field Length = 4               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

- Esempio Data set

Src IP Addr.	Dst IP Addr.	Next-Hop Addr.	Packet Number	Octets Number
192.0.2.12	192.0.2.254	192.0.2.1	5009	5344385
192.0.2.27	192.0.2.23	192.0.2.2	748	388934
192.0.2.56	192.0.2.65	192.0.2.3	5	6534

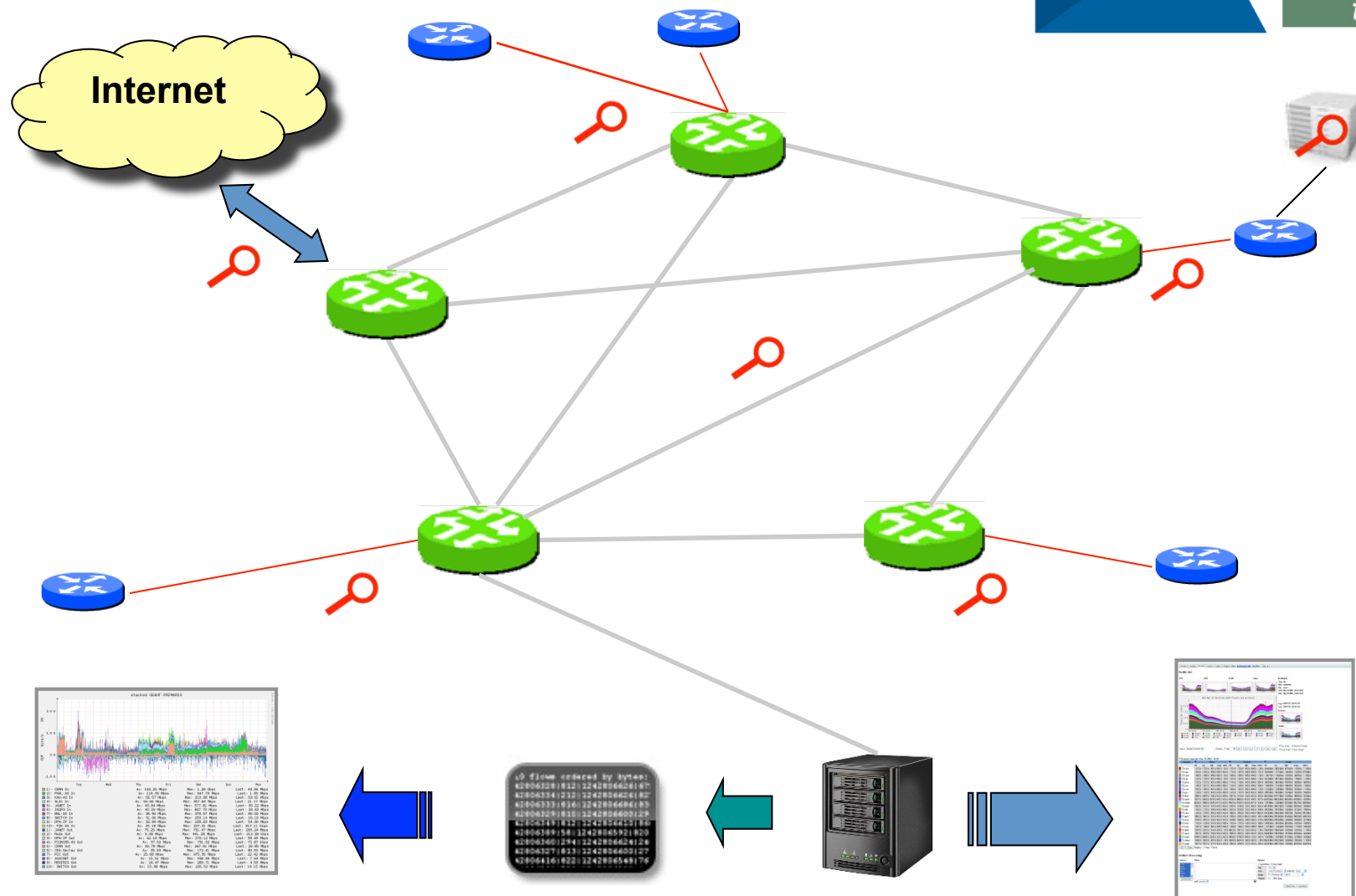
[illegible]

- Esportazione dei flussi

- Collezione  ed analisi 



Architettura



Analisi elaborata

Elaborazione

Collezione

Analisi diretta

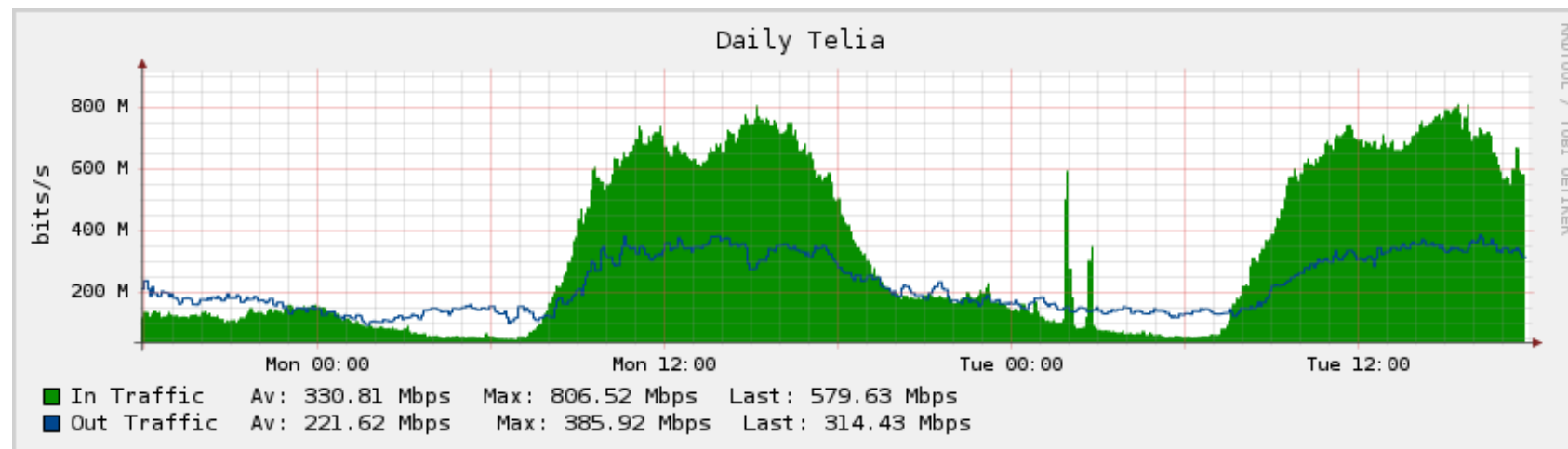
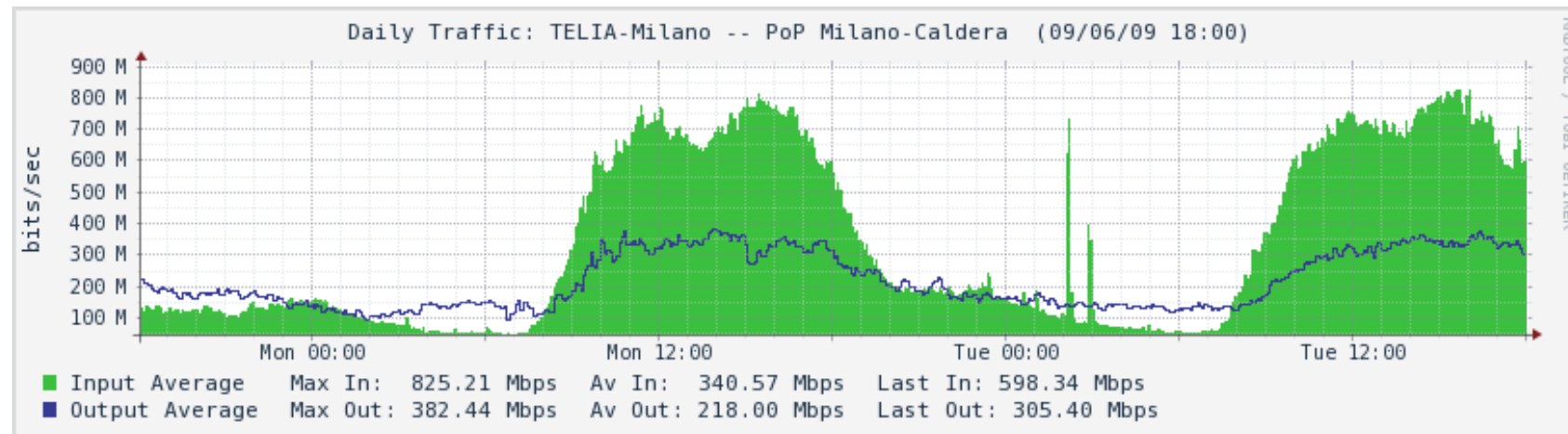
- Netflow e' stato concepito, in origine, per far esportare i flussi dagli apparati di rete, quali router e switch.
- Esistono software per server che sono in grado di emulare il comportamento dei router
- Parametri da configurare:
 - Interfacce di cui si vogliono esportare i flussi
 - Indirizzo e porta del collector
 - Versione di Netflow
 - Timeout per l'esportazione (active e inactive timeout)
 - Eventuale tasso di campionamento
 - Nel caso di Netflow v9 va definito il template dei dati da esportare

Sampling (1/2)



- La cattura dei flussi Netflow può risultare gravosa per il router, portando la CPU in saturazione. Per evitare problemi dovuti al sovraccaricamento del processore del router, è stato introdotto un meccanismo di campionamento: invece di catturare tutti i pacchetti in transito, il router prende un pacchetto ogni n . Tale parametro è configurabile a piacere.
- Quando viene usato il campionamento, il traffico visualizzato nei sistemi di analisi dei flussi è approssimato.
- Ma per la legge dei grandi numeri i conti tornano...

Sampling (2/2)



Cisco vs Juniper router (1/2)



■ Esempi di configurazione (v5):

■ Cisco

■ Configurazione interfacce

interface FastEthernet0/0

ip route cache flow

■ Esportazione dei flussi

ip flow-export

ip flow-export version 5

ip flow-export destination 172.16.0.1 20000

ip flow-cache timeout active 5

ip flow-cache timeout inactive 1000

Cisco vs Juniper router (2/2)



- Juniper
 - Definizione del filtro:
set firewall family inet filter NETFLOW-SAMPLE term default then sample
set firewall family inet filter NETFLOW-SAMPLE term default then accept
 - Configurazione interfacce:
set interfaces fe-0/1/0 unit 0 family inet filter input NETFLOW-SAMPLE
set interfaces fe-0/1/0 unit 0 family inet filter output NETFLOW-SAMPLE
 - Configurazione collezionatore
set forwarding-options sampling input family inte rate 1000
set forwarding-options sampling output cflowd 172.16.0.1 port 20000
set forwarding-options sampling output cflowd version 5
set forwarding-options sampling output cflowd autonomous-system-type [peer|origin]

Collezione e analisi



- Esistono applicativi open-source e commerciali per collezionare ed analizzare i flussi esportati. Quelli open piu' conosciuti sono: flow-tools, stager, ntop, Nerd e la suite nfsen/nfdump
- La scelta, nel caso di GARR, e' ricaduta sulla suite Nfsen/Nfdump.
 - E' un progetto open-source sviluppato dalla NREN svizzera SWITCH (Peter Haag e' l'autore).
 - Tool completo
 - Supporto Netflow 9 (IPv6, MPLS)
 - Tool da linea di comando per la collezione e l'analisi dei flussi (nfcapd, nfdump)
 - altre utility per la gestione dei flussi (ri-esportazione dei flussi, cancellazione dei flussi piu' vecchi, conversione da altri formati, etc.)
 - Possibilità di anonimizzare i flussi
 - un sistema di plugin per l'estensione delle funzionalità
 - Interfaccia grafica
 - Software in continuo sviluppo (mailing lista attiva!)

Suite Nfdump/NfSen



- Installazione e configurazione
- Nfcapd
 - Collezione e salvataggio dei dati
- Nfdump
 - Analisi dei dati via shell
- Nfsen
 - Analisi dei dati tramite interfaccia web
 - Profili
 - Allarmi
 - Plugin

Installazione e configurazione



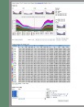
- Nfdump
 - pacchetti binari delle distribuzioni linux
es. `apt-get install nfdump`
 - sorgenti
es. `./configure --enable-nfprofile && make && make install`
- Nfsen
 - Prerequisiti:
 - PHP
 - Perl
 - RRDtools
 - Nfdump con `--enable-nfprofile`
 - Installazione da sorgenti
 - Copiare `nfsen-dist.conf` in `nfsen.conf`
 - Modificare `nfsen.conf` definendo i propri exporter in `%sources` e i path
 - Lanciare `./install.pl </path/nfsen.conf>`

Nfcapd



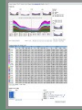
- Demone che riceve i pacchetti Netflow e li scrive periodicamente (per default ogni 5 minuti) su disco sotto forma di file (dati binari indicizzati)
- Un'istanza di nfcapd per ogni router collezionato oppure un'istanza per piu' router
- Permette di lanciare un comando shell ogni volta che ha finito di scrivere un nuovo set di dati (con `-x <comando>`).
- *Sintassi: `nfcapd -w -D -s <sampling rate> -T <v9 extension tags> -l <source_name> -p <porta> -u <user> -g <group> -B <buffer size> -S <dir-hierarchy level> -l /<path>/source_name`*
- *Esempio: `nfcapd -w -D -s 1000 -T all -l router1 -p 12001 -B 200000 -S 1 -l /data/nfsen/profiles-data/live/router1`*

Nfdump (1/2)



- Serve per consultare ed elaborare i dati salvati su disco sotto forma di file binari da nfcapd
- Caratteristiche:
 - Molto efficiente (dati binari indicizzati, scritto in C)
 - Supporta i formati Netflow 5,9 e IPFIX
 - Supporta sflow (sfcapd)
 - Sintassi semplice ed intuitiva (possibilità di definire filtri stile "tcpdump")
 - Consultazione molto flessibile:
 - Analisi dei singoli flussi registrati
 - Generazione dei top talkers (bytes, flows, packets)
 - Aggregazione dei dati (temporale e di sorgenti diverse).

Nfdump (2/2)



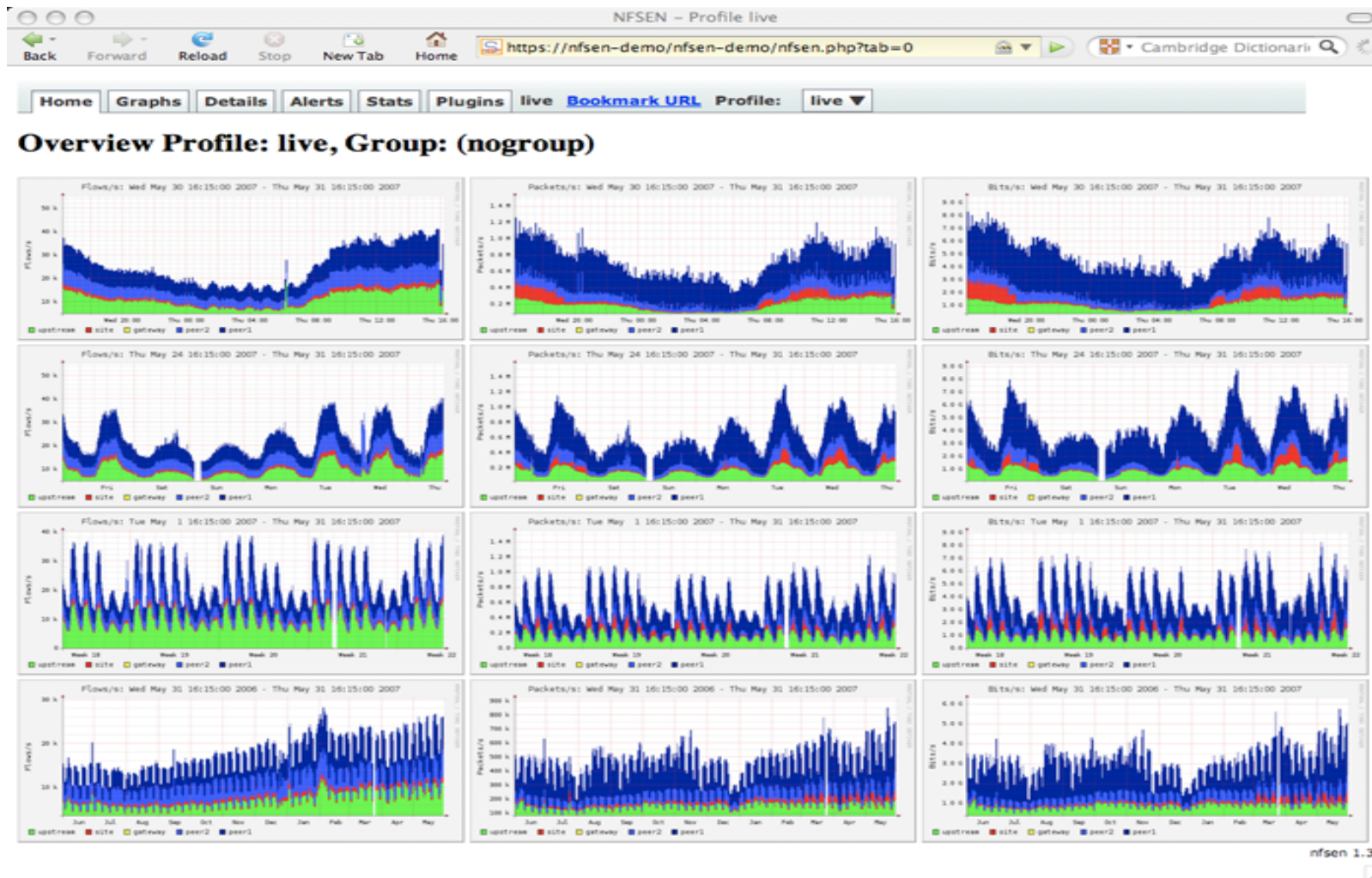
- Caratteristiche (continua ...):
 - Crea statistiche su aggregati di flussi (record) secondo un insieme di criteri scelti dall'utente (src/dst IP, src/dst port, src/dst AS, protocol, ToS, interface)
 - Permette la personalizzazione del formato di output, prevede un output machine readable per lo scripting ed integrazioni con alcuni linguaggi di programmazione
 - Consente l'anonimizzazione degli indirizzi IP tramite la libreria CryptoPan
 - E' utilizzato in background da nfsen



- Interfaccia grafica web di nfdump
- Caratteristiche:
 - utilizza `nfdump` come back-end
 - fornisce una visione complessiva e dettagliata allo stesso tempo dello stato della rete
 - grafici specifici per profili (per host, per porta etc.)
 - analisi di un particolare lasso di tempo
 - sistema di post-processing automatico e alerting
 - flessibile con l'utilizzo dei plugin
 - utilizzo intuitivo
 - funzionalità di IP lookup
 - “si pulisce da solo”

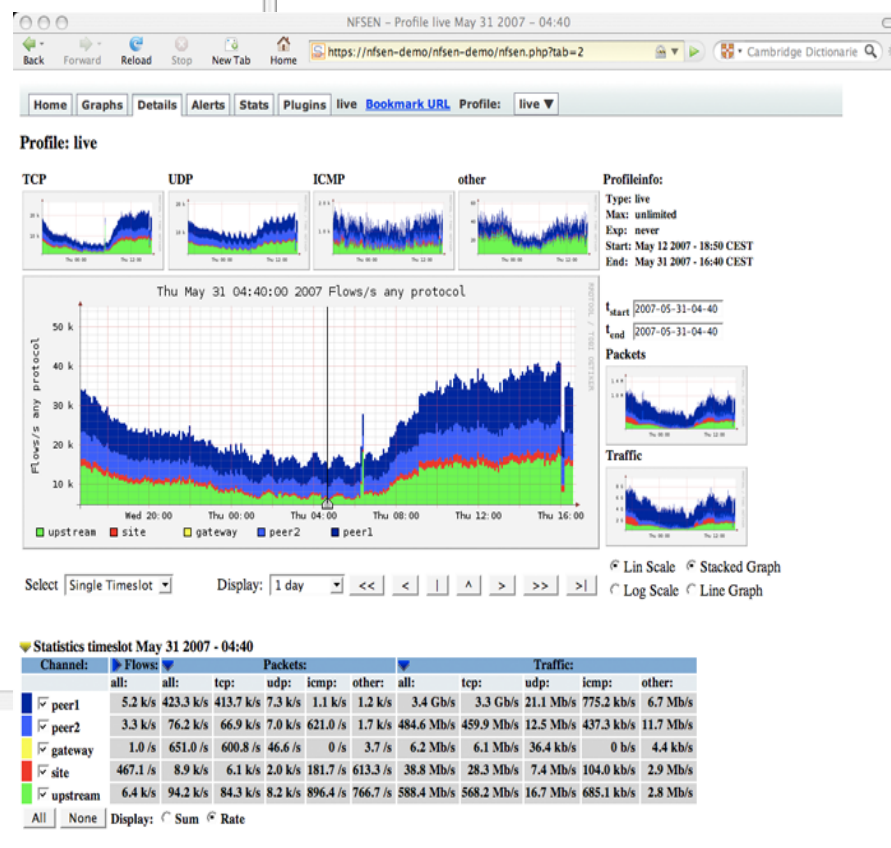
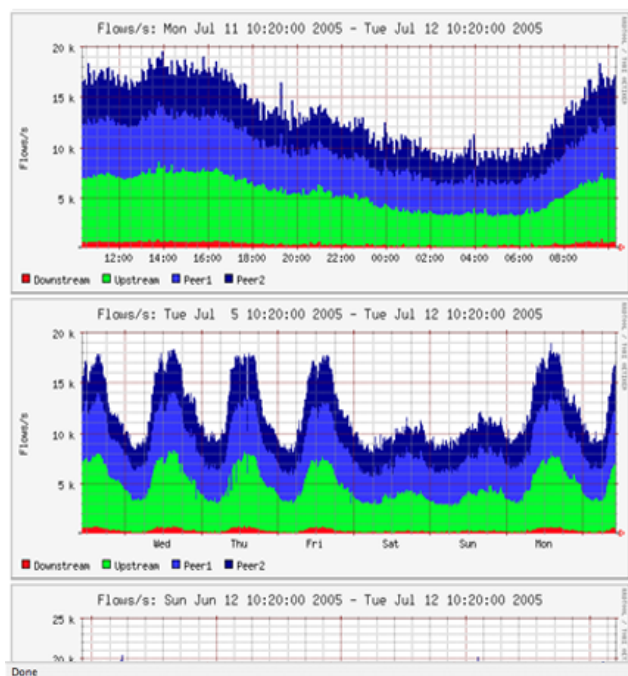
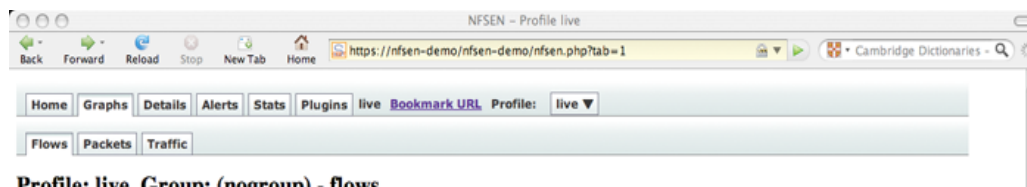
Interfaccia grafica di NfSen (1/3)

Home Tab



Interfaccia grafica di NfSen (2/3)

Graphs e Details tab



Interfaccia grafica di NfSen (3/3)

Dettaglio dei flussi

NFSEN - Profile live May 31 2007 - 04:40

Back Forward Reload Stop New Tab Home <https://nfsen-demo/nfsen-demo/nfsen.php#processing> Cambridge Dictionary

	peer2	gateway	site	upstream
3.3 k/s	76.2 k/s	66.9 k/s	7.0 k/s	621.0 /s
1.0 /s	651.0 /s	600.8 /s	46.6 /s	0 /s
467.1 /s	8.9 k/s	6.1 k/s	2.0 k/s	181.7 /s
6.4 k/s	94.2 k/s	84.3 k/s	8.2 k/s	896.4 /s

Display: ☐ Sum ☐ Rate

Netflow Processing

Source: peer1, peer2, gateway, site, upstream
Filter:
All Sources and <none>

Options:
☐ List Flows ☒ Stat TopN
Top: 10
Stat: Flow Records order by flows
Aggregate: ☒ proto, ☒ srcPort, ☒ dstPort
Limit: ☐ Packets > 0
Output: line ☐ / IPv6 long
Clear Form process

```
** nfdump -M /netflow0/nfsen-demo/profile-data/live/peer1:peer2:gateway:site:upstream -T -r 2007/05/31/04/nfcapd.200705310440
nfdump filter:
any
Aggregated flows 2797250
Top 10 flows ordered by flows:
Date flow start      Duration Proto      Src IP Addr:Port      Dst IP Addr:Port      Packets      Bytes      Flows
2007-05-31 04:39:54.045 299.034 UDP      116.147.95.88:1110    188.142.64.162:27014    68          5508       68
2007-05-31 04:39:56.282 298.174 UDP      116.147.249.27:1478   188.142.64.163:27014    67          5427       67
2007-05-31 04:39:57.530 298.206 UDP      117.196.44.62:1031    188.142.64.166:27014    67          5427       67
2007-05-31 04:39:57.819 298.112 UDP      117.196.75.134:1146   188.142.64.167:27014    67          5427       67
2007-05-31 04:39:53.787 297.216 UDP      61.191.235.132:4121    60.9.138.37:4121       62          3720       62
2007-05-31 04:39:55.354 300.833 UDP      60.9.138.37:2121      118.25.93.95:2121       61          3660       61
2007-05-31 04:39:58.936 298.977 UDP      60.9.138.36:2121      119.182.123.166:2121    61          3660       61
2007-05-31 04:39:54.329 303.585 UDP      120.150.194.76:2121    60.9.138.37:2121       61          3660       61
2007-05-31 04:39:53.916 300.734 UDP      60.9.138.37:2121      125.167.25.128:2121    61          3660       61
2007-05-31 04:39:57.946 300.353 UDP      60.9.138.36:2121      121.135.4.186:2121     61          3660       61

IP addresses anonymized
Summary: total flows: 4616424, total bytes: 156.6 G, total packets: 172.6 M, avg bps: 644.8 M, avg pps: 90946, avg bpp: 929
Time window: 2007-05-31 04:11:49 - 2007-05-31 04:44:58
Total flows processed: 4616424, skipped: 0, Bytes read: 240064932
Sys: 6.184s flows/second: 746464.4 Wall: 6.185s flows/second: 746361.3
```

nfSen 1.3

- Demo:
 - Navigazione
 - Dettagli
 - spostamento cursore
 - scelta intervallo di tempo singolo o “lungo”
 - demo per processare filtri e dati
 - cambio protocollo/unita' di misura
 - vedere qualche picco
 - aggregazioni varie
 - finestra “lookup”

- Costituiscono una vista specifica di dati netflow ottenuta attraverso l'applicazione di un filtro di nfdump
- Vista GRAFICA di un filtro
- Si possono creare nel passato
- Si possono creare in modo “continuo”
- Puo' contenere un numero qualsiasi di sorgenti
- Puo' essere generato da un qualsiasi filtro

Profili: esempi

- Tutti gli host che si connettono a server noti di botnet
- il filtro e':
 - 'ip in [ip1, ip2, ..., ipN]'
 dove ip1, ip2, ..., ipN sono ip di botnet



Statistics timeslot Jun 02 2009 - 05:30

Channel:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:
rcfi	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s
rtmi2	0.0/s	0/s	0/s	0.0/s	0/s	0.0/s	0/s	0/s	0.0/s	0/s	2.0 b/s	0 b/s	0 b/s	2.0 b/s	0 b/s	0/s	0/s	0 b/s	0 b/s	0 b/s
rtbo1	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0 b/s	0 b/s	0 b/s	0 b/s	0 b/s	0/s	0/s	0 b/s	0 b/s	0 b/s
rtfi	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0 b/s	0 b/s	0 b/s	0 b/s	0 b/s	0/s	0/s	0 b/s	0 b/s	0 b/s
rtpi1	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0/s	0 b/s	0 b/s	0 b/s	0 b/s	0 b/s	0/s	0/s	0 b/s	0 b/s	0 b/s
rtna1	0.0/s	0.0/s	0/s	0/s	0/s	0.0/s	0.0/s	0/s	0/s	0/s	1.6 b/s	1.6 b/s	0 b/s	0 b/s	0 b/s	0/s	0/s	0 b/s	0 b/s	0 b/s

All None Display: Sum Rate

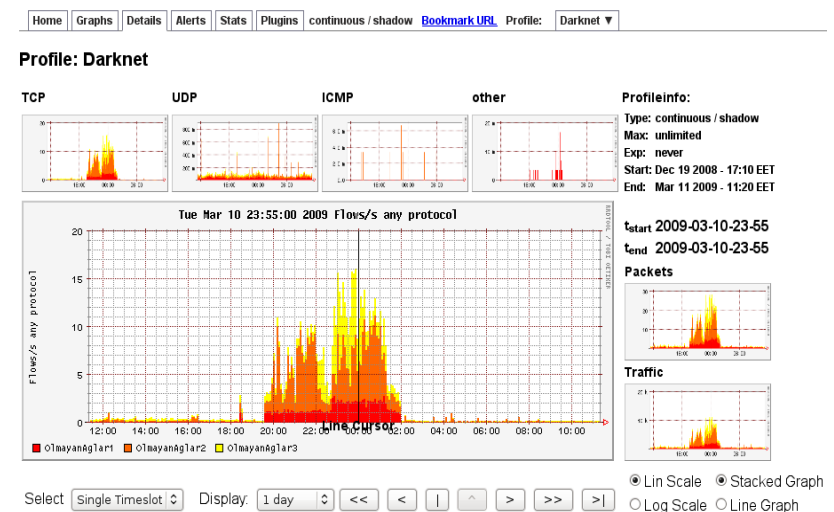
Netflow Processing

Source: Filter: Options:

Find: nfsen Next Previous Highlight all Match case

Done

- Monitoraggio di una DarkNet
- il filtro e':
 - 'net rete.dark.net/24'



Hide stat table Statistics timeslot Mar 10 2009 - 23:55

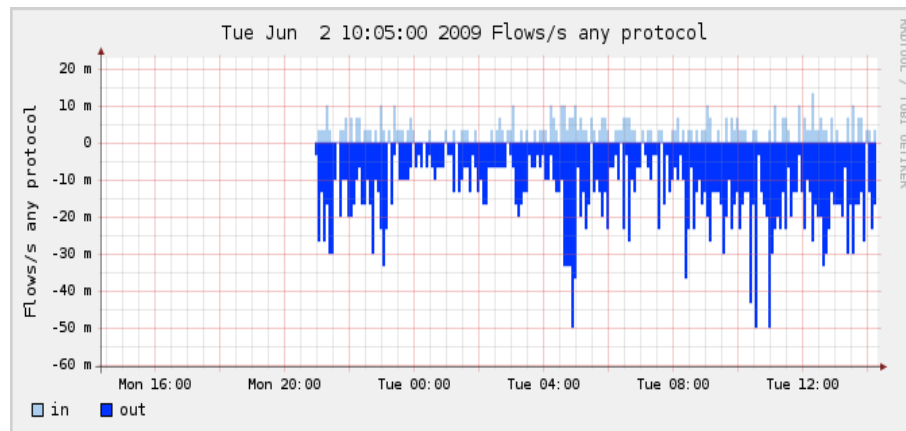
Channel:	collapse				Flows:				collapse				Packets:				collapse				Traffic:			
	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:				

Netflow Processing

Done

Profili: creazione

- Si possono definire canali con segno positivo e negativo per visualizzare il traffico in entrata e uscita contemporaneamente sullo stesso grafico.
- In questo caso monitoriamo server di posta suddividendo il traffico fra quello che esce (sopra) e quello che entra sulla porta 25/TCP



Description:			
Type:	Continuous / shadow		
Start:	2008-04-01-00-00		
End:	2009-06-02-13-20		
Last Update:	2009-06-02-13-20		
Size:	0 B		
Max. Size:	unlimited		
Expire:	never		
Status:	OK		
Channel List: +			
▼ in			
Colour:	#abcdef	Sign:	+
Order:	1		
Filter:	port 25 and (dst net 193.205.230.0/24 or dst net 192.167.9.0/24)		
Sources:	rcfi rtbo1 rtfi rtmi2		
▼ out			
Colour:	#0033FF	Sign:	-
Order:	1		
Filter:	port 25 and (src net 193.205.230.0/24 or src net 192.167.9.0/24)		

- **Continuous:** e' un profilo che, una volta creato, viene aggiornato ogni 5 minuti col nuovo traffico che soddisfa il filtro impostato
- **History:** e' un profilo che si ferma in un punto del passato e non viene piu' aggiornato
- **Real:** e' un profilo i cui dati filtrati vengono immagazzinati su una parte diversa del disco rispetto al profilo di default
- **Shadow:** e' un profilo i cui dati vengono filtrati, vengono generati i grafici ma non vengono salvati, il calcolo viene fatto al volo quando richiesto

Alert

Home	Graphs	Details	Alerts	Stats	Plugins	live	Bookmark URL	Profile:	live ▼
Alerts overview: +									
No.	Status	Name	Last Triggered						
1	armed	supero_media_1h	Tue Jun 2 05:10:00 2009						
2	armed	Bottnett1_alert	Tue Jun 2 12:55:00 2009						
3	armed	NOC_alert	Tue Jun 2 04:10:00 2009						
4	inactive	DoS_UDP_generico2	never						
5	inactive	dos2_alert	Fri Feb 13 12:40:00 2009						
6	armed	DoS_UDP_generico	Tue Jun 2 04:55:00 2009						

- Ogni 5 minuti viene controllata la condizione di alerting impostata
- Estrema flessibilita' nelle impostazione di soglie e filtri
- Sistema automatico di "reazione" una volta che la condizione di alert e' verificata
- Sistema automatico di "notifica" una volta che la condizione di alert e' verificata

Alert: esempi di condizioni



- Manda un alert quando il numero di flussi nei 5 minuti $e' > 2$
- Manda un alert quando il rate dei flussi del router X supera i 100 flussi/secondo
- Invia una mail al superamento della soglia dei 1000 pacchetti/secondo sul router Y
- Esegui un programma esterno che calcola l'ip coinvolto e il suo relativo traffico, ogni volta che il numero di byte per protocollo TCP, sulla porta 80, che viene dalla sorgente X, supera il 30% della media calcolata su un periodo di 6 ore (leggermente piu' complicato)

Alert: configurazione

Stato

Filtro

Condizioni

Trigger

Azioni

Alerts details: NOC_alert		
Trigger	Status	Last Triggered
armed	☒ enabled	2009-06-02-04:10
Filter applied to 'live' profile:		
rtna1 rtpi1 rtfi rtbo1	proto UDP	
Conditions based on total flow summary:		
0	Total flows >	50
	1 hour average value	%
1	Flows/s >	20
	24 hour average value	-
Conditions based on individual Top 1 statistics:		
Conditions based on plugin:		
Trigger:		
Each time after 1x condition = true, and block next trigger for 0 cycles		
Action:		
☐ No action		
☐ Send alert email		
To:		
Subject: Alert triggered		
☒ Call plugin: NOC_alert		
Alert Infos:		
Last cycle: 2009-06-02-16:20		
Mon Jun 1 16:20:00 2009 - Tue Jun 2 16:20:00 2009 Flows/s		

Alert: infos

Alert Infos:

Last cycle: 2009-06-05-14:30



	Last	Avg 10m	Avg 30m	Avg 1h	Avg 6h	Avg 12h	Avg 24h
☒ Flows	17.1 k	17.0 k	16.7 k	16.3 k	17.1 k	11.9 k	12.1 k
	56.9 /s	56.6 /s	55.5 /s	54.3 /s	56.9 /s	39.6 /s	40.3 /s
☐ Packets	32.2 k	32.5 k	30.6 k	31.9 k	49.3 k	42.5 k	40.3 k
	107.5 /s	108.4 /s	102.0 /s	106.5 /s	164.4 /s	141.8 /s	134.2 /s
☐ Bytes	2.1 MB	2.1 MB	2.0 MB	2.0 MB	2.9 MB	2.6 MB	2.5 MB
	55.3 kb/s	56.2 kb/s	52.7 kb/s	52.8 kb/s	76.7 kb/s	69.4 kb/s	67.5 kb/s

Conditions:	0	1	Final:
State:	False	False	False

- Servono a:
 - estendere le funzionalità di Nfsen/Nfdump
 - automatizzare una serie di operazioni complesse
- Possono funzionare insieme agli alert
 - Generazione di una condizione di alert
 - Azione a seguito di un alert
- Quando usarli?
 - Monitoraggio e notifica di eventi particolari sulla rete
 - Tracciare botnet e mandare notifiche
 - Tracciare scanning, DoS, probe non visibili dai grafici
 - Analisi di traffico: porte maggiormente utilizzate o distribuzione di protocolli
 - Elaborazione di statistiche avanzate

- Esistono due tipi di plugin:
 - Back-end plugin:
 - Sono moduli perl che vengono “agganciati” ad Nfsen
 - Fanno cose altrimenti non possibili con la suite
 - Sono richiamati automaticamente ogni 5 minuti
 - Front-end plugin:
 - Sono moduli PHP “agganciati” ad NfSen
 - Fanno parte delle pagine web di NfSen
 - Anche questi si aggiornano ogni 5 minuti

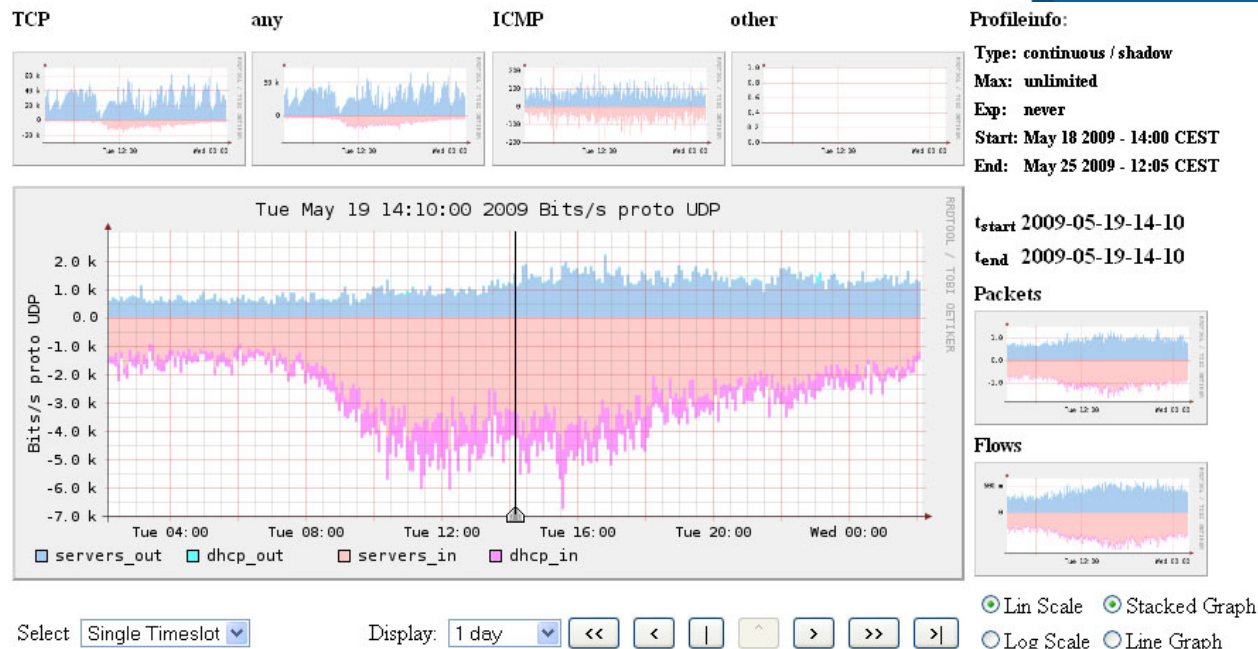
Esempi di uso di Nfsen/Nfdump



- Monitoraggio della rete
 - Analisi per subnet, per protocollo, per porta, per AS
 - Subnet
 - Servizi (Mail, DNS, WEB)
 - Protocollo
 - Porte (plugin PortTracker)
 - Numeri AS (plugin AsTracker)
 - TopTalkers
 - Aggregazione predefinita
 - Aggregazione personalizzata
- Sicurezza
 - Analisi degli incidenti di sicurezza
 - Tracciamento degli host
 - Identificazione di traffico “malevolo”

- E' possibile condurre analisi per ogni singolo campo previsto (indirizzi IP, protocollo, porte, AS, interfacce, etc)
 - Tramite lo strumento “profili” si possono configurare statistiche specifiche relative a criteri desiderati.
 - Ogni profilo, per default, fornisce le statistiche distinte per protocollo.
- Esempi:
 - Suddivisione delle rete in categorie(es. Workstation, DMZ, dhcp hosts)
 - Bogon Network, ICMP su Upstream Provider
 - Mail Server, Name Server, Web Server
 - ASes

Esempio: Traffico suddiviso per subnet



Statistics timeslot May 19 2009 - 14:10

Channel:	Flows:					Packets:					Traffic:				
	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:
☒ dhcp_out	0.0 /s	0.0 /s	0.0 /s	0 /s	0 /s	0.0 /s	0.0 /s	0.0 /s	0 /s	0 /s	12.1 b/s	10.6 b/s	1.6 b/s	0 b/s	0 b/s
☒ servers_out	3.7 /s	3.0 /s	0.6 /s	0.1 /s	0 /s	5.0 /s	3.8 /s	1.0 /s	0.1 /s	0 /s	30.2 kb/s	29.0 kb/s	1.1 kb/s	108.9 b/s	0 b/s
☒ servers_in	2.9 /s	2.2 /s	0.6 /s	0.1 /s	0 /s	4.0 /s	2.7 /s	1.2 /s	0.1 /s	0 /s	13.2 kb/s	9.7 kb/s	3.4 kb/s	81.7 b/s	0 b/s
☒ dhcp_in	0.1 /s	0.0 /s	0.0 /s	0 /s	0 /s	0.1 /s	0.0 /s	0.1 /s	0 /s	0 /s	951.5 b/s	103.1 b/s	848.4 b/s	0 b/s	0 b/s
All None Display: ☐ Sum ☒ Rate															
☒ dhcp_in	0.1 /s	0.0 /s	0.0 /s	0 /s	0 /s	0.1 /s	0.0 /s	0.1 /s	0 /s	0 /s	951.5 b/s	103.1 b/s	848.4 b/s	0 b/s	0 b/s
All None Display: ☐ Sum ☒ Rate															

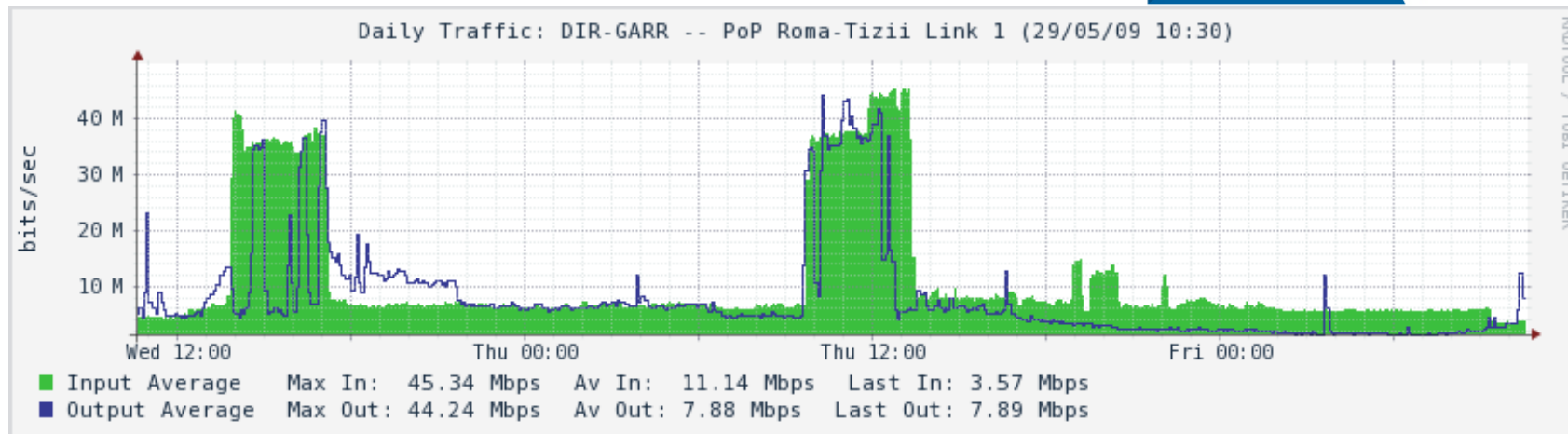
Esempio: filtri impostati

▼ servers_in			
Colour:	#FCCCCC	Sign:	-
		Order:	1
Filter:	out if 113 and dst net 193.206.158.0/24		
Sources:	rtrm2		

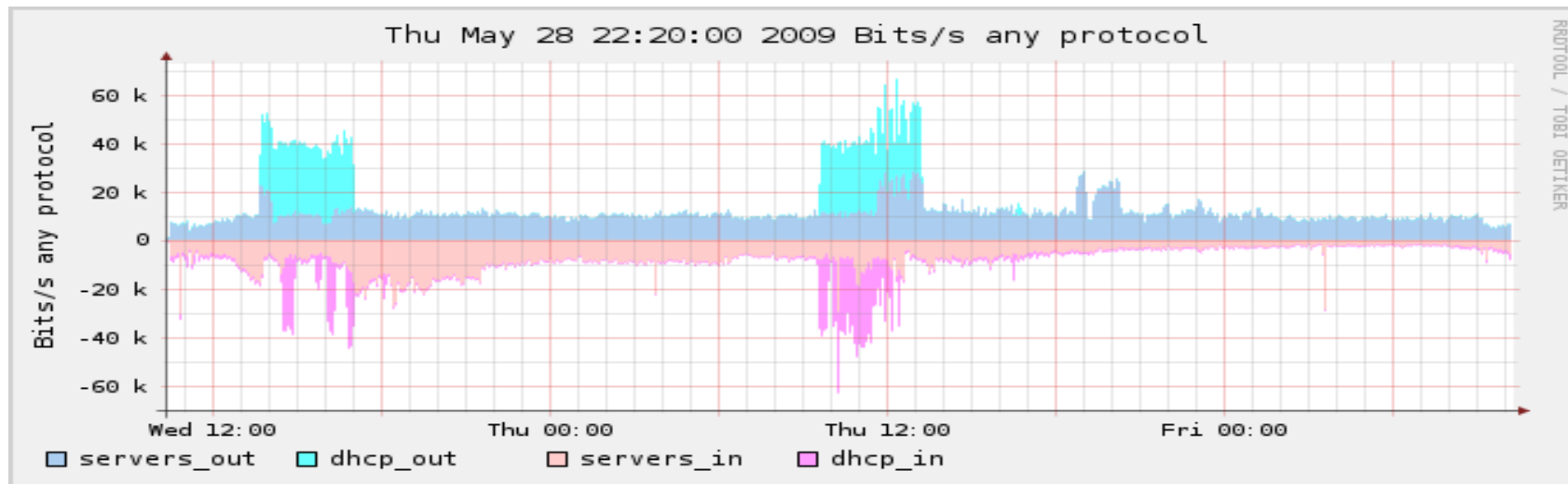
▼ dhcp_in			
Colour:	#FF99FF	Sign:	-
		Order:	2
Filter:	out if 113 and dst net 193.206.159.0/24		
Sources:	rtrm2		

Monitoring Tradizionale vs Analisi dei flussi

Monitoring Tradizionale



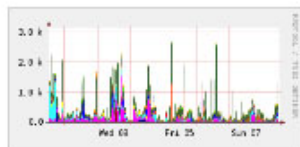
Analisi dei flussi (filtro per subnet)



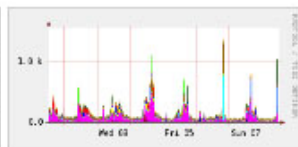
Bogon Network

Profile: Bogon_Nets

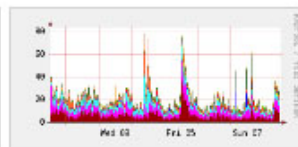
TCP



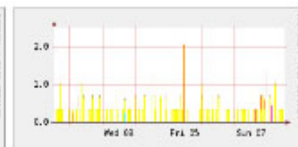
UDP



ICMP



other



Profileinfo:

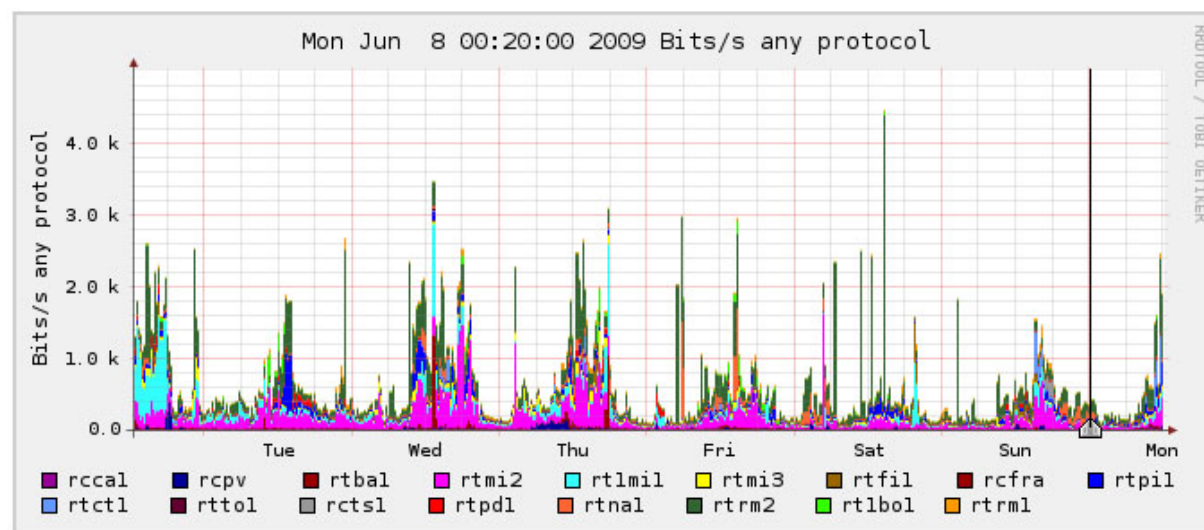
Type: continuous / shadow

Max: unlimited

Exp: never

Start: Sep 30 2008 - 00:00 CEST

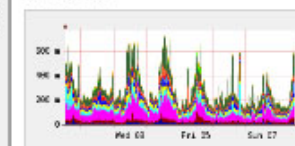
End: Jun 08 2009 - 12:20 CEST



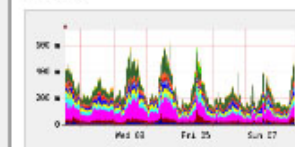
t_start 2009-06-08-00-20

t_end 2009-06-08-00-20

Packets



Flows



Select Single Timeslot

Display: 1 week

☒ Lin Scale ☒ Stacked Graph
☐ Log Scale ☐ Line Graph

ICMP su Upstream Provider

Profile: Telia ICMP in out

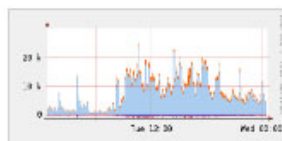


Web Server

Home Graphs Details Alerts Stats Plugins continuous / shadow [Bookmark URL](#) Profile: RM3_web ▼

Profile: RM3_web

TCP



UDP



ICMP



other



Profileinfo:

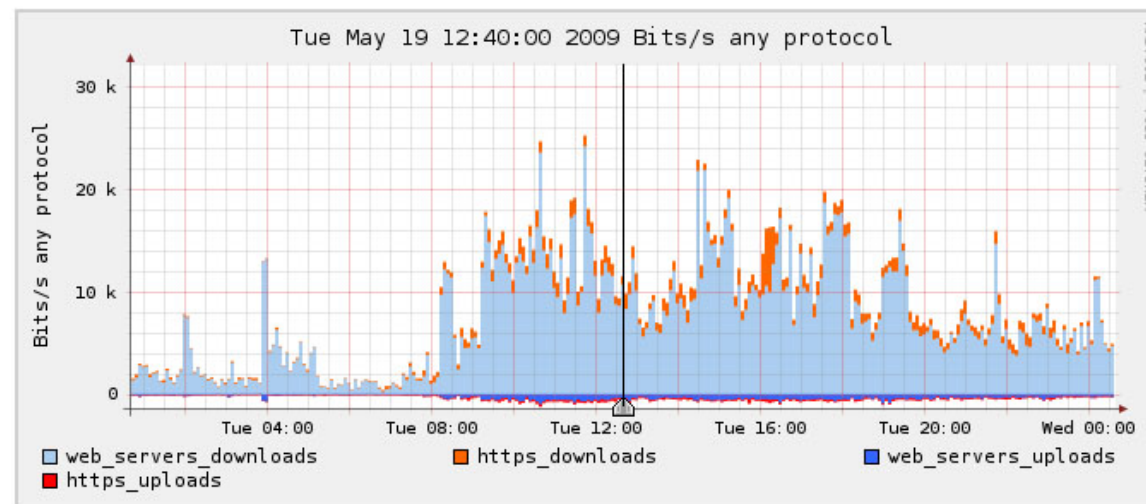
Type: continuous / shadow

Max: unlimited

Exp: never

Start: May 17 2009 - 14:00 CEST

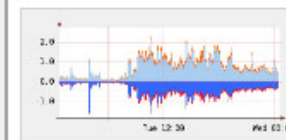
End: May 21 2009 - 14:40 CEST



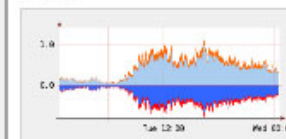
tstart 2009-05-19-12-40

tend 2009-05-19-12-40

Packets



Flows



Select Single Timeslot ▼

Display: 1 day << < | ^ > >> >|

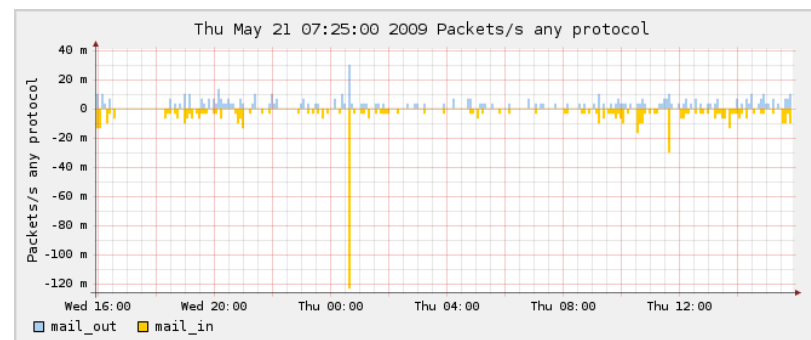
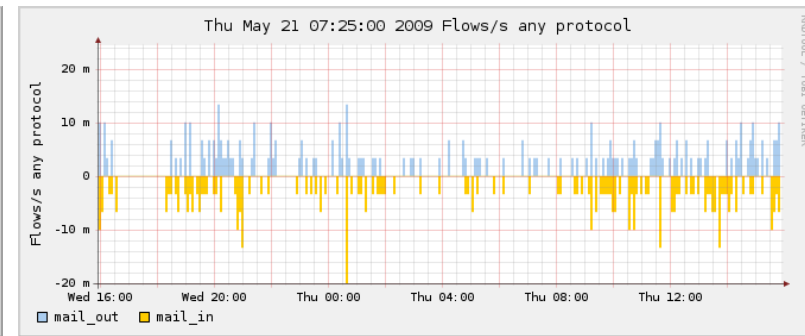
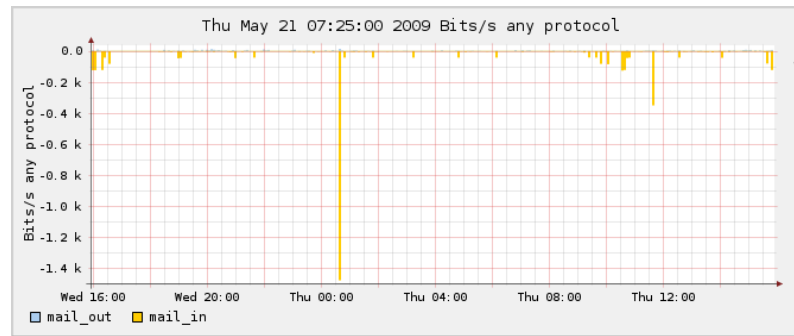
☒ Lin Scale ☒ Stacked Graph
☐ Log Scale ☐ Line Graph

Web Server: Filtri impostati

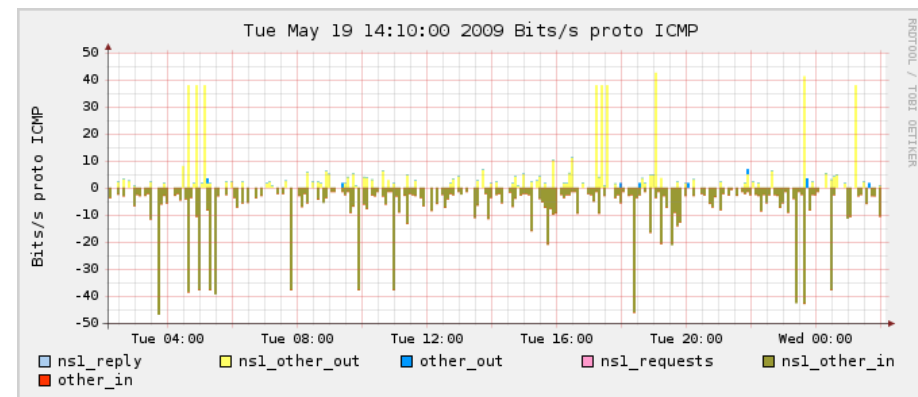
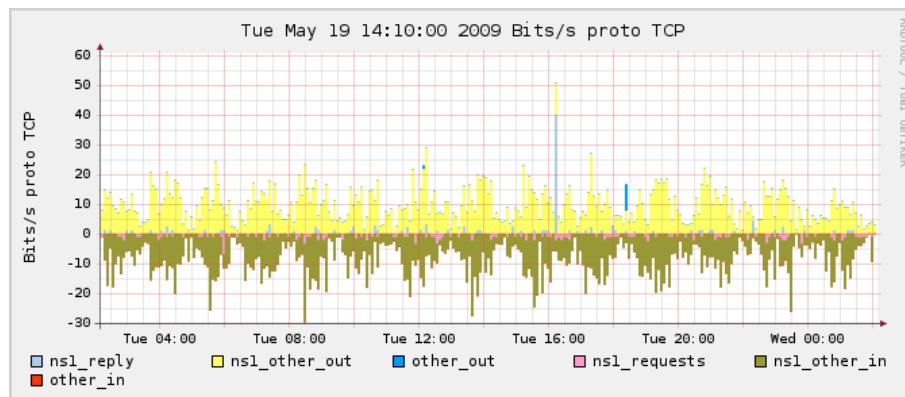
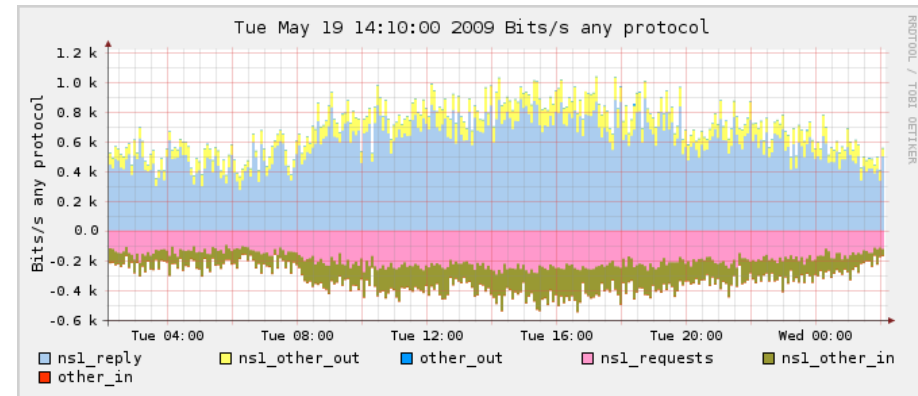
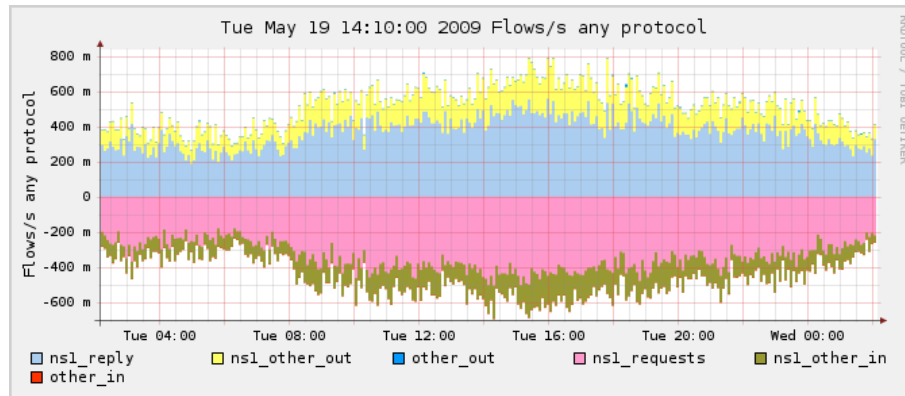
▼ web_servers_downloads			
Colour:	#abcdef	Sign:	+ Order: 1
Filter:	in if 305 and proto tcp and src port 80		
Sources:	rtrm2		

▼ web_servers_uploads			
Colour:	#3366FF	Sign:	- Order: 1
Filter:	out if 305 and proto tcp and dst port 80		
Sources:	rtrm2		

Mail Server



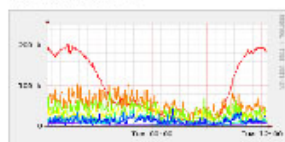
Name Server



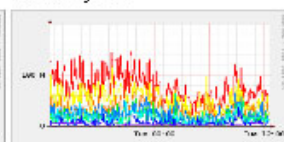
Porte (plugin PortTracker)

Port Tracker

TCP Packets



TCP Bytes



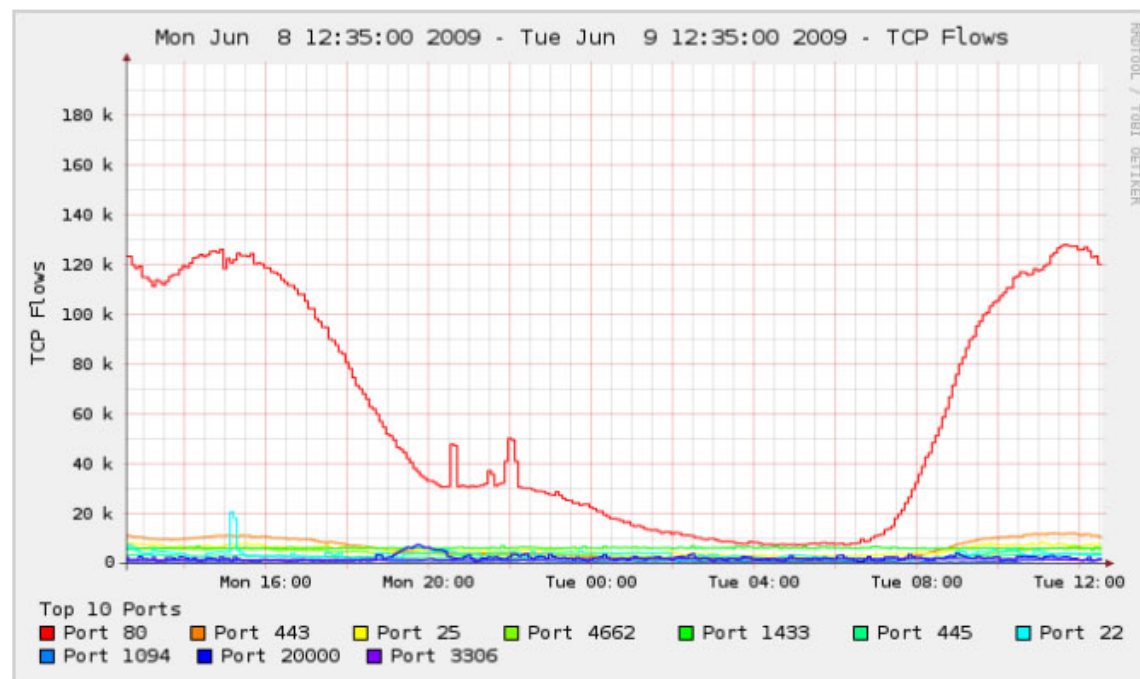
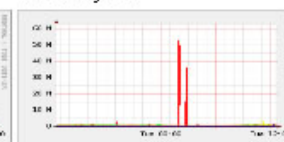
UDP Flows



UDP Packets



UDP Bytes



Show Top 10 Ports

☒ now ☐ 24 hours

Track Ports:

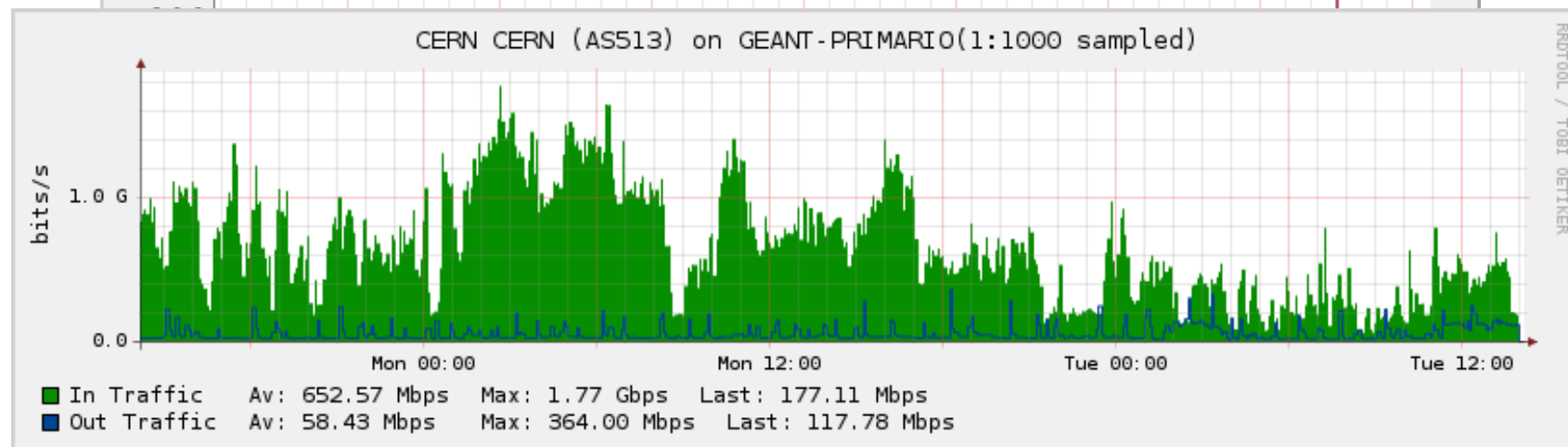
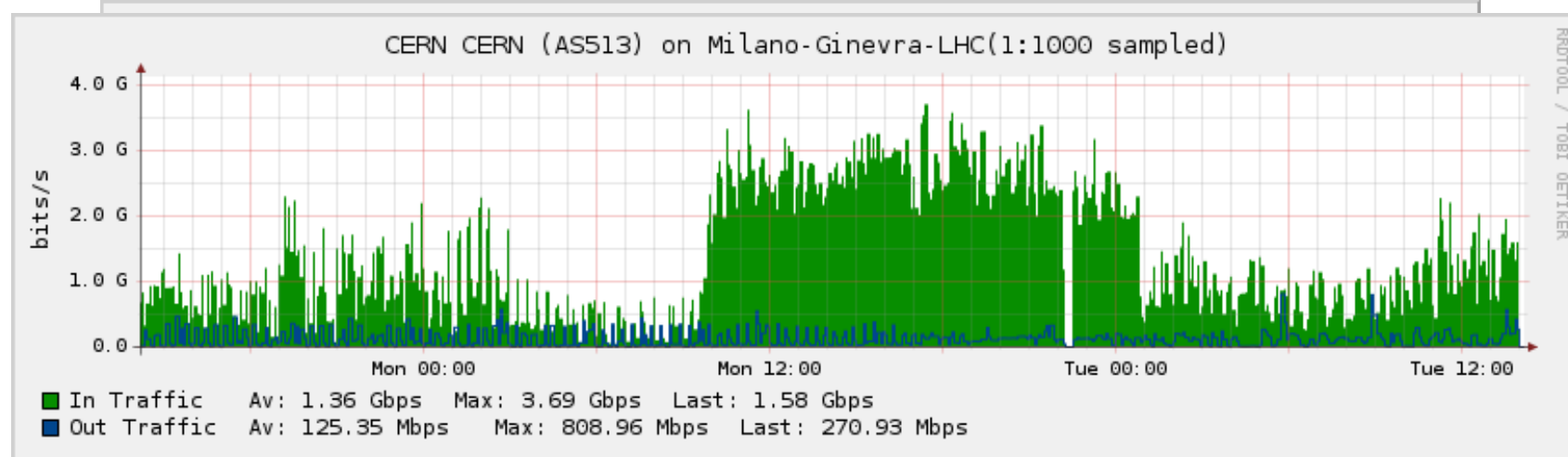
Skip Ports:

Display 1 day

Y-axis: ☒ Linear ☐ Log

Type: ☐ Stacked ☒ Line

Numeri AS (plugin AsTracker)



7) - ULTRALIGHT Out	Av: 58.38 Mbps	Max: 560.50 Mbps	Last: 44.85 kbps
8) - IN2P3 Out	Av: 37.21 Mbps	Max: 297.78 Mbps	Last: 2.94 Mbps
9) - BNL-AS Out	Av: 22.48 Mbps	Max: 486.40 Mbps	Last: 6.08 Mbps
10) - TRIUMF Out	Av: 21.19 Mbps	Max: 342.27 Mbps	Last: 4.77 Mbps

IP sorgenti che hanno spedito piu' bytes in uscita sull'interfaccia 88 di rtrm2:

Netflow Processing

Source:
 Filter:
 Options:
☐ List Flows ☒ Stat TopN
 Top: 10
 Stat: SRC IP Address order by bytes
 Limit: ☐ Packets > 0 -
 Output: ☐ / IPv6 long
 Clear Form process

** nfdump -M /data/nfsen/profiles-data/live/rtrm2 -T -r 2009/06/09/nfcapd.200906090150 -n 10 -s srcip/bytes

Nfdump filter:

out if 88

Top 10 Src IP Addr ordered by bytes:

Date first seen	Duration	Proto	Src IP Addr	Flows	Packets	Bytes	pps	bps	bpp
2009-06-09 01:59:33.076	299.668	any	194.206.113.164	783	7790	10.7 M	25	300382	1444
2009-06-09 01:59:33.075	299.669	any	168.170.108.199	196	1010	1.4 M	3	38555	1429
2009-06-09 02:00:00.276	272.203	any	128.81.67.96	45	860	931808	3	27385	1083
2009-06-09 01:59:33.263	299.480	any	168.170.80.52	277	684	916303	2	24477	1339
2009-06-09 01:59:33.073	299.401	any	183.233.148.108	274	409	503907	1	13464	1232
2009-06-09 01:59:33.297	299.443	any	194.205.41.37	37	199	291483	0	7787	1464
2009-06-09 01:59:33.596	296.011	any	183.233.112.25	107	188	241836	0	6535	1286
2009-06-09 01:59:35.302	296.923	any	168.170.220.175	115	184	234215	0	6310	1272
2009-06-09 01:59:35.260	297.489	any	194.205.152.107	28	155	220376	0	5926	1421
2009-06-09 01:59:37.731	293.831	any	178.50.109.75	62	156	214522	0	5840	1375

Summary: total flows: 13583, total bytes: 24.3 M, total packets: 27850, avg bps: 680244, avg pps: 92, avg bpp: 914

Time window: 2009-06-09 01:59:32 - 2009-06-09 02:04:32

Total flows processed: 52740, Records skipped: 0, Bytes read: 2742528

Sys: 0.016s flows/second: 3296250.0 Wall: 0.012s flows/second: 4091543.8

IP che hanno avuto il maggior throughput:

nfdev:~# *nfdump -r /data/nfsen/profiles-data/live/rtrm2/2009/05/20/nfcapd.200905201000 -s ip/bps*

Top 10 IP Addr ordered by bps:

Date first seen	Duration	Proto	IP Addr	Flows	Packets	Bytes	pps	bps	bpp
2009-05-20 10:00:30.610	0.001	any	150.51.86.103	2	4	4540	3999	34.6 M	1135
2009-05-20 09:59:09.513	0.001	any	75.250.7.176	1	2	3000	1999	22.9 M	1500
2009-05-20 10:02:29.722	0.001	any	246.172.20.59	2	2	1540	1999	11.7 M	770
2009-05-20 10:03:29.710	0.001	any	226.82.105.197	2	2	1540	1999	11.7 M	770
2009-05-20 10:03:00.206	0.001	any	140.67.231.196	2	2	1460	1999	11.1 M	730
2009-05-20 10:03:00.206	0.001	any	139.119.70.252	2	2	1460	1999	11.1 M	730
2009-05-20 09:59:08.572	0.001	any	86.241.106.137	2	2	1322	1999	10.1 M	661
2009-05-20 10:03:19.627	0.003	any	225.136.245.127	3	3	3052	999	7.8 M	1017
2009-05-20 10:01:52.910	0.003	any	86.234.231.165	1	2	2984	666	7.6 M	1492
2009-05-20 10:01:52.910	0.003	any	224.190.13.154	1	2	2984	666	7.6 M	1492

Summary: total flows: 161250, total bytes: 185.8 M, total packets: 265150, avg bps: 4.1 M, avg pps: 736, avg bpp: 734

Time window: 2009-05-20 09:58:46 - 2009-05-20 10:04:46

Total flows processed: 161250, Records skipped: 0, Bytes read: 8385120

Sys: 0.108s flows/second: 1492972.6 Wall: 0.099s flows/second: 1620733.3

IP che hanno scambiato piu' bytes:

```
nfdev:~$ nfdump -r /data/nfsen/profiles-data/live/rtrm2/2009/05/20/nfcapd.200905201000 -s ip/bytes
```

Top 10 IP Addr ordered by bytes:

Date first seen	Duration	Proto	IP Addr	Flows	Packets	Bytes	pps	bps	bpp
2009-05-20 09:58:46.982	359.855	any	225.207.115.198	5200	16856	15.2 M	46	354798	946
2009-05-20 09:58:47.116	359.780	any	225.207.115.194	3845	12558	11.4 M	34	266840	955
2009-05-20 09:58:47.795	359.039	any	225.207.119.162	1784	3661	3.0 M	10	69643	853
2009-05-20 09:58:48.694	357.934	any	146.236.52.35	125	2464	2.5 M	6	59696	1083
2009-05-20 09:58:47.147	359.749	any	225.204.174.91	1514	2756	2.1 M	7	48689	794
2009-05-20 09:58:46.984	299.885	any	225.204.240.255	122	1392	2.0 M	4	55092	1483
2009-05-20 09:58:47.540	359.356	any	139.3.140.89	297	2007	1.9 M	5	44553	997
2009-05-20 09:58:47.444	359.261	any	224.20.182.157	1478	2433	1.7 M	6	40415	745
2009-05-20 09:58:46.984	299.885	any	159.127.154.129	51	1198	1.7 M	3	47938	1500
2009-05-20 10:02:02.549	17.876	any	225.18.67.245	16	1449	1.4 M	81	657509	1013

Summary: total flows: 161250, total bytes: 185.8 M, total packets: 265150, avg bps: 4.1 M, avg pps: 736, avg bpp: 734

Time window: 2009-05-20 09:58:46 - 2009-05-20 10:04:46

Total flows processed: 161250, Records skipped: 0, Bytes read: 8385120

Sys: 0.112s flows/second: 1439655.0 Wall: 0.109s flows/second: 1478828.7

IP piu' informati...:

```
nfsen:~# nfdump -r /data/nfsen/profiles-data/live/rtrm2/2009/06/01/nfcapd.200906011520 -o 'fmt: %ts %td %da %byt %fl' "src host www.ansa.it"
```

Resolving www.ansa.it ...

IPv4 address: 194.244.5.206 (194.244.5.206)

Date flow start	Duration	Dst IP Addr	Bytes	Flows
2009-06-01 15:19:46.354	0.000	192.222.119.198	1500	1
2009-06-01 15:19:47.490	0.000	192.222.119.198	1500	1
2009-06-01 15:19:49.490	0.000	192.222.119.198	40	1
2009-06-01 15:19:58.479	0.000	194.206.33.67	1500	1
2009-06-01 15:21:22.588	0.000	194.206.5.154	347	1
2009-06-01 15:21:42.761	0.000	168.170.192.19	40	1
2009-06-01 15:21:32.637	0.000	168.170.195.141	40	1
2009-06-01 15:21:02.716	0.000	168.170.115.252	48	1
2009-06-01 15:22:24.463	0.000	195.80.138.242	48	1
2009-06-01 15:22:58.452	0.000	168.170.192.62	1500	1

Summary: total flows: 10, total bytes: 6563, total packets: 10, avg bps: 273, avg pps: 0, avg bpp: 656

Time window: 2009-06-01 15:18:57 - 2009-06-01 15:24:57

Total flows processed: 131043, Records skipped: 0, Bytes read: 6814344

Sys: 0.028s flows/second: 4679940.0 Wall: 0.019s flows/second: 6680073.4

TopTalkers con aggregazione personalizzata

Statistiche aggregate per netmask /24: e protocollo in uscita sulla interfaccia 113 del router rtrm2 (Direzione GARR):

Netflow Processing

Source:
 rct1
 rto1
 rcts1
 rtpd1
 rtna1
 rtrm2
 All Sources

Filter:
 out if 113
 and <none>

Options:
☒ List Flows ☐ Stat TopN
Limit to: 10000 Flows
Aggregate
☒ proto
☐ srcPort ☐ srcIP
☐ dstPort ☒ dstIPv4/ 24
Sort: ☐ start time of flows
Output: long ☐ / IPv6 long
Clear Form process

```
** nfdump -M /data/nfsen/profiles-data/live/rtrm2 -T -r 2009/06/09/nfcapd.200906090150 -a -A proto,dstip4/24 -o long -c 10000
nfdump filter:
out if 113
Date flow start      Duration Proto      Src IP Addr:Port      Dst IP Addr:Port      Flags Tos  Packets  Bytes  Flows
2009-06-09 01:59:40.297 291.930 UDP        0.0.0.0:0 -> 193.206.158.0:0 ..... 0      263    64401   111
2009-06-09 01:59:59.298 270.818 UDP        0.0.0.0:0 -> 193.206.159.0:0 ..... 0       8    11552    7
2009-06-09 01:59:34.298 294.930 ICMP       0.0.0.0:0 -> 193.206.158.0:0 ..... 0      14     1400   13
2009-06-09 01:59:33.287 293.520 TCP        0.0.0.0:0 -> 193.206.158.0:0 .AP.SF 0     139    30099  115
2009-06-09 02:01:09.888  0.321 TCP        0.0.0.0:0 -> 193.206.159.0:0 .A...F 0       2       80    2
Summary: total flows: 248, total bytes: 107532, total packets: 426, avg bps: 2877, avg pps: 1, avg bpp: 252
Time window: 2009-06-09 01:59:32 - 2009-06-09 02:04:32
Total flows processed: 52740, Records skipped: 0, Bytes read: 2742528
Sys: 0.008s flows/second: 6592500.0 Wall: 0.008s flows/second: 6464027.5
```

TopTalkers con aggregazione presonalizzata



Sintassi (nfdump): “-s record/<misura> -A <campo/i da aggregare>”

Nota: alcune aggregazioni non sono supportate da nfsen

Top 10 AS sorgenti in ingresso sull’interfaccia avente ifindex 88:

```
nfdev:~$ nfdump -r /data/nfsen/profiles-data/live/rtrm2/2009/05/20/nfcapd.200905201000 -s record/bytes -A srcAS -o 'fmt: %sas %byt' "in if 88"
```

Aggregated flows 1477

Top 10 flows ordered by bytes:

Src AS	Bytes
15169	9.3 M
3356	5.6 M
22822	3.3 M
32934	1.6 M
6453	1.3 M
30058	934121
40824	900070
20473	823638
3320	624968
12322	622658

Summary: total flows: 30776, total bytes: 42.1 M, total packets: 50866, avg bps: 982085, avg pps: 141, avg bpp: 868

Time window: 2009-05-20 09:58:46 - 2009-05-20 10:04:46

Total flows processed: 161250, Records skipped: 0, Bytes read: 8385120

Sys: 0.036s flows/second: 4479042.2 Wall: 0.033s flows/second: 4796965.6

TopTalkers con aggregazione personalizzata



Le 15 coppie di subnet /24 che fanno piu' traffico:

```
nfdev:~$ nfdump -r /data/nfsen/profiles-data/live/rtrm2/2009/05/20/nfcapd.200905201400 -n 15 -A srcip4/24,dstip4/24 -s record/bytes
```

Aggregated flows 97223

Top 15 flows ordered by bytes:

Date flow start	Duration	Proto	Src IP Addr:Port	Dst IP Addr:Port	Packets	Bytes	Flows
2009-05-20 14:00:47.661	147.406	0	181.236.167.0:0	-> 190.145.223.0:0	1886	2.7 M	12
2009-05-20 14:00:47.661	115.573	0	181.236.167.0:0	-> 190.145.215.0:0	1796	2.6 M	12
2009-05-20 13:59:55.152	61.697	0	194.29.173.0:0	-> 181.236.167.0:0	1001	1.4 M	18
2009-05-20 14:02:46.055	123.094	0	195.155.83.0:0	-> 185.97.168.0:0	692	1038000	9
2009-05-20 13:59:49.975	299.796	0	198.72.64.0:0	-> 169.113.129.0:0	591	839092	63
2009-05-20 13:58:53.163	55.841	0	181.236.165.0:0	-> 188.121.158.0:0	502	750432	21
2009-05-20 13:58:50.670	296.337	0	168.170.152.0:0	-> 73.131.20.0:0	505	716592	12
2009-05-20 13:58:52.243	297.526	0	194.206.113.0:0	-> 192.225.109.0:0	371	549080	10
2009-05-20 13:59:40.489	248.515	0	194.206.113.0:0	-> 170.122.12.0:0	363	544500	7
2009-05-20 14:01:17.925	127.824	0	95.64.160.0:0	-> 188.80.192.0:0	360	540000	6
2009-05-20 13:59:52.428	296.694	0	194.201.138.0:0	-> 194.205.140.0:0	362	539781	59
2009-05-20 13:58:50.328	297.970	0	194.206.113.0:0	-> 79.40.54.0:0	337	502804	8
2009-05-20 13:58:51.130	298.494	0	194.206.113.0:0	-> 81.201.82.0:0	322	483000	19
2009-05-20 13:58:53.064	296.526	0	194.206.113.0:0	-> 216.138.183.0:0	331	482598	10
2009-05-20 13:59:13.108	276.661	0	194.206.113.0:0	-> 95.186.140.0:0	328	471493	17

Summary: total flows: 185341, total bytes: 217.8 M, total packets: 308936, avg bps: 4.8 M, avg pps: 858, avg bpp: 739

Time window: 2009-05-20 13:58:49 - 2009-05-20 14:04:49

Total flows processed: 185341, Records skipped: 0, Bytes read: 9637876

Sys: 0.116s flows/second: 1597684.6 Wall: 0.145s flows/second: 1270860.3

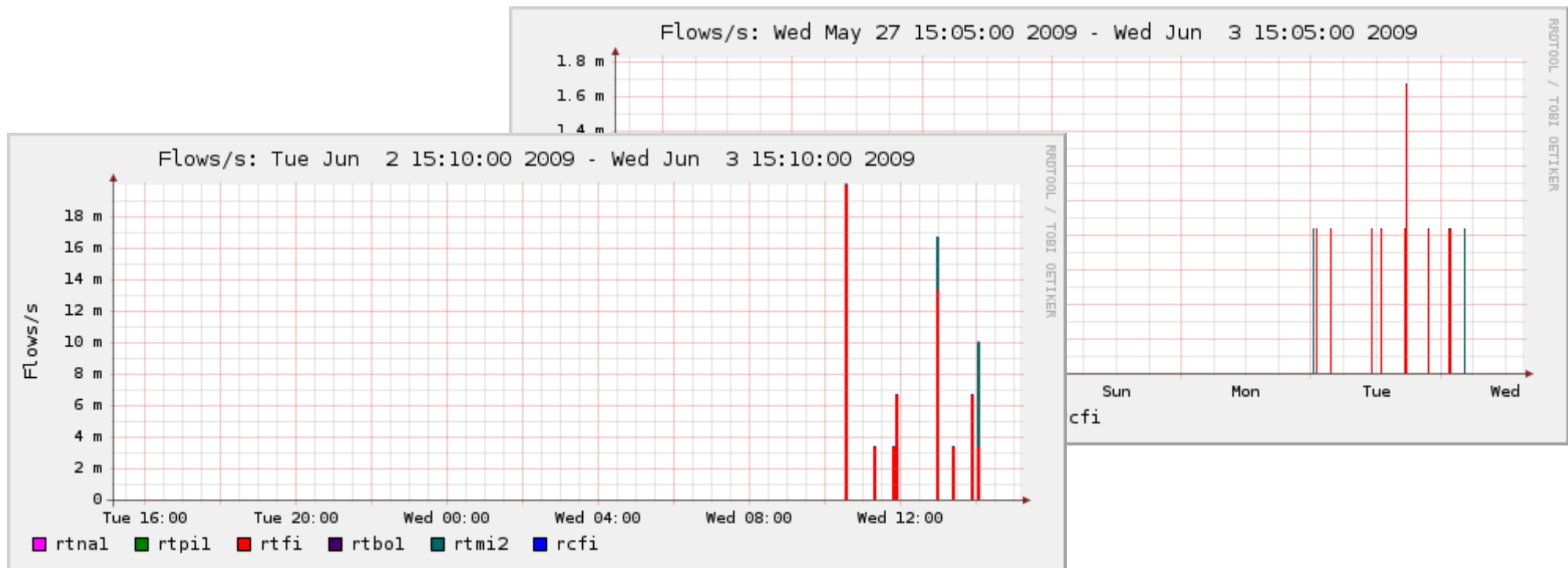
Analisi incidenti di sicurezza



- Segnalazione esterna di macchina che fa traffico anomalo
- Controllo del traffico sul router a quel timeslot
- Tracciamento di ip spoofati in caso di DDoS
- Come procedere in.....
 - Caso di PHISHING
 - controllo delle macchine che gli si sono collegate
 - Caso di Probe/Scan
 - controllo di cosa fa la macchina nel tempo precedente al timeslot
 - Caso di SPAM
 - controllo delle connesstioni sulla TCP/25
 - Caso di DoS
 - controllo dei picchi e dei flussi entranti/uscenti dalla macchina e dal router nelle vicinanze di quel timeslot

Tracciamento degli Host (1/2)

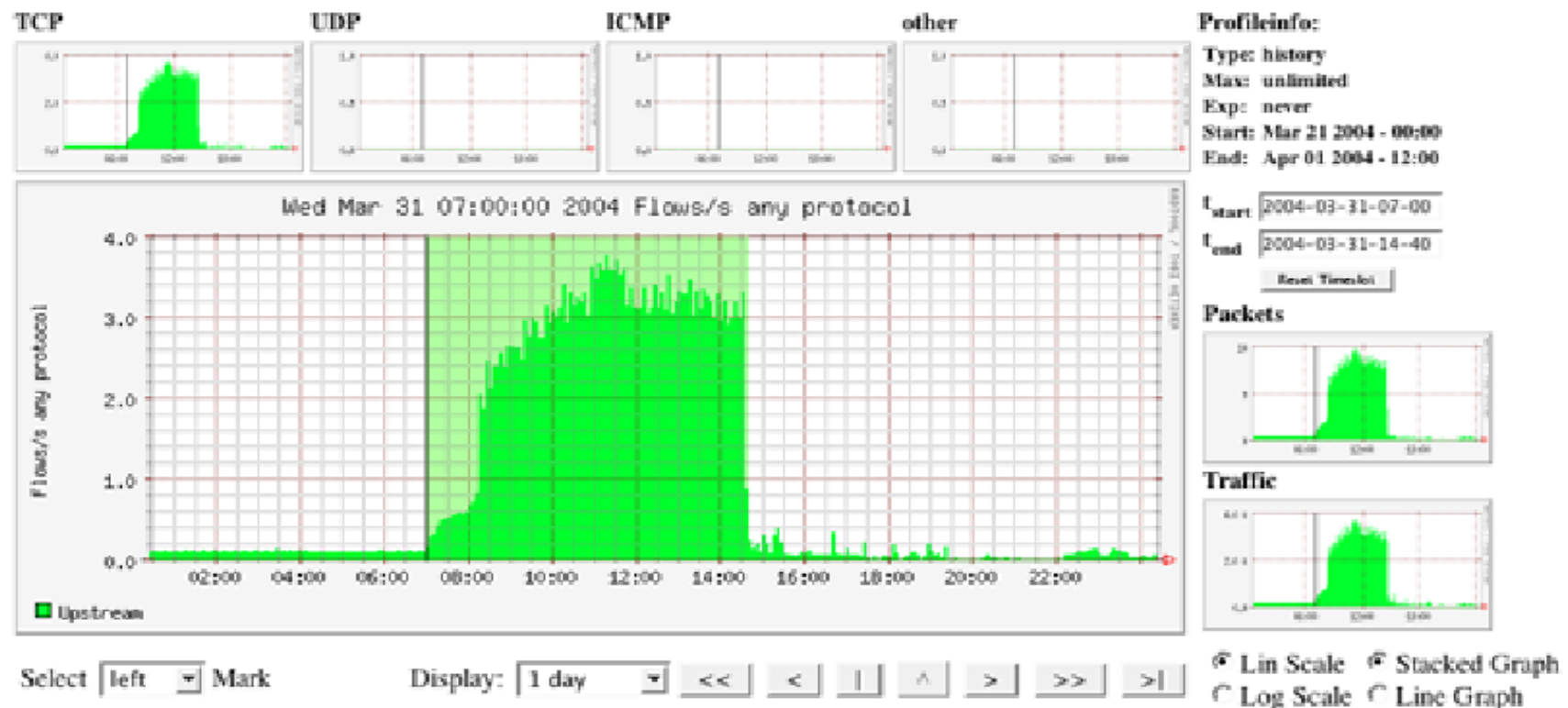
- Segnalazione di un host che fa traffico malevolo
- Profilatura dell'host a partire da qualche giorno o settimana prima della segnalazione tramite shadow profile
- Controllo continuo del traffico della macchina



Tracciamento degli Host (2/2)

■ Profilatura di un host compromesso

Profile: bot-masters



Identificazione di traffico “malevolo”



Esempio (Presunti IP sorgenti di scanning e porte bersaglio):

```
nfdev:~$ nfdump -r /data/nfsen/profiles-data/live/router1/2009/05/20/nfcapd.200905201400 -A srcip,dstport -s record/packets "not proto icmp and bytes < 100 and bpp < 100 and packets < 5 and not port 80 and not port 53 and not port 110 and not port 123 and not port 22"
```

Aggregated flows 30404

Top 10 flows ordered by packets:

Date flow start	Duration	Proto	Src IP Addr:Port	Dst IP Addr:Port	Packets	Bytes	Flows
2009-05-20 13:58:56.362	289.563	0	87.177.37.47:0	-> 0.0.0.0: 1080	185	7614	182
2009-05-20 13:58:50.298	296.047	0	195.155.76.158:0	-> 0.0.0.0: 443	66	2912	66
2009-05-20 13:59:52.617	282.893	0	168.170.104.98:0	-> 0.0.0.0: 445	54	2584	54
2009-05-20 13:58:53.117	295.026	0	212.238.91.31:0	-> 0.0.0.0: 445	47	2256	45
2009-05-20 13:58:51.228	292.875	0	168.170.128.34:0	-> 0.0.0.0: 445	45	2160	45
2009-05-20 13:58:51.678	292.002	0	168.173.164.192:0	-> 0.0.0.0: 445	43	1995	43
2009-05-20 13:58:52.974	289.299	0	169.113.129.189:0	-> 0.0.0.0: 445	42	2016	42
2009-05-20 13:59:51.397	295.530	0	168.170.170.9:0	-> 0.0.0.0: 445	42	2016	42
2009-05-20 13:58:51.872	296.238	0	168.170.186.180:0	-> 0.0.0.0: 445	38	1824	38
2009-05-20 13:58:50.895	287.792	0	168.170.118.100:0	-> 0.0.0.0: 445	36	1720	35

Summary: total flows: 41946, total bytes: 2.1 M, total packets: 45408, avg bps: 47839, avg pps: 126, avg bpp: 47

Time window: 2009-05-20 13:58:49 - 2009-05-20 14:04:49

Total flows processed: 185341, Records skipped: 0, Bytes read: 9637876

Sys: 0.128s flows/second: 1447897.4 Wall: 0.682s flows/second: 271443.8

Identificazione di traffico “malevolo”



Esempio (Presunti IP che fanno DoS):

```
nfdev:~$ nfdump -r /data/nfsen/profiles-data/live/router1/2009/05/20/nfcapd.200905201400 -a -l 40000 -o extended 'proto udp and bpp < 200'
```

Date flow start	Duration	Proto	Src IP Addr:Port	Dst IP Addr:Port	Packets	Bytes	pps	bps	Bpp	Flows
2009-06-03 14:37:50.813	414.316	UDP	210.188.235.117:32807	-> xxx.xxx.8.9:53	87874	3.6 M	212	72960	43	16
2009-06-03 13:59:17.225	3922.643	UDP	XXX.XX.80.47:1194	-> 84.221.68.167:6244	119843	19.3 M	30	41373	169	106

Summary: total flows: 651996, total bytes: 92.4 M, total packets: 1012985, avg bps: 193379, avg pps: 252, avg bpp: 95

Time window: 2009-06-03 13:58:17 - 2009-06-03 15:05:06

Total flows processed: 7938624, Records skipped: 0, Bytes read: 412816524

Sys: 1.268s flows/second: 6260354.4 Wall: 1.190s flows/second: 6666272.6

Identificazione di traffico “malevolo”



SSH scan

```
nfdev:~$ nfdump -r /data/nfsen/profiles-data/live/router1/2009/05/20/nfcapd.200905201400 -A srcip -s record/packets  
"proto TCP and dport 22 and flags S and not flags AFRPU"
```

2009-06-04 16:59:54.057	xxx.80.208.239:35823	->	yyy.200.199.203:22	S.	1	60	1
2009-06-04 16:59:39.123	xxx.80.208.239:44577	->	yyy.200.195.226:22	S.	1	60	1
2009-06-04 17:01:54.084	xxx.80.208.239:53769	->	yyy.200.237.133:22	S.	1	60	1
2009-06-04 17:01:31.024	xxx.80.208.239:60829	->	yyy.200.232.39:22	S.	1	60	1
2009-06-04 16:59:53.279	xxx.80.208.239:53731	->	yyy.200.199.108:22	S.	1	60	1
2009-06-04 17:00:28.348	xxx.80.208.239:34654	->	yyy.200.211.104:22	S.	1	60	1
2009-06-04 16:59:31.080	xxx.80.208.239:54641	->	yyy.200.194.28:22	S.	1	60	1
2009-06-04 16:59:18.316	xxx.80.208.239:34426	->	yyy.200.189.162:22	S.	1	60	1
2009-06-04 17:00:34.093	xxx.80.208.239:51640	->	yyy.200.212.76:22	S.	1	60	1
2009-06-04 16:58:54.459	xxx.80.208.239:58954	->	yyy.200.182.14:22	S.	1	60	1
2009-06-04 16:59:35.459	xxx.80.208.239:58471	->	yyy.200.195.5:22	S.	1	60	1
2009-06-04 17:01:01.080	xxx.80.208.239:52688	->	yyy.200.222.197:22	S.	1	60	1

Summary: total flows: 105, total bytes: 6300, total packets: 105, avg bps: 214, avg pps: 0, avg bpp: 60

Time window: 2009-06-04 16:58:17 - 2009-06-04 17:05:07

Total flows processed: 565374, Records skipped: 0, Bytes read: 29400036

Sys: 0.068s flows/second: 8313834.5 Wall: 0.064s flows/second: 8823628.6

Netflow sulla LAN/NAT



- switch (Sflow)
 - Open vSwitch
- server (nprobe)
 - monitoring NAT, firewall, server

Sflow: caratteristiche



- Il protocollo Sflow, oltre a supportare i protocolli IP/ICMP/UDP/TCP, supporta anche:
 - Packet headers
 - Ethernet/802.3
 - IP/ICMP/UDP/TCP
 - IPX Appletalk
- Nel layer 2 supporta le interfaccia di ingresso e uscita ma anche:
 - Priorita' 802.1p Ingresso/Uscita
 - VLAN 802.1Q Ingresso/Uscita
- Configurabile via SNMP
- Piattaforme HW: 3com, Alcatel-Lucent, Extreme networks, Force10 networks, HP, Juniper (EX series),

■ Netflow su cisco (solo su serie 4500/6500)

```
Router(config-if)# mls flow ip destination-source
Router(config-if)# mls nde src_address 10.1.1.37 version 8
Router(config-if)# interface vlan1
Router(config-if)# ip route-cache flow
Router(config-if)# interface fastethernet 3/2
Router(config-if)# ip address 10.200.8.2 255.255.255.0
Router(config-if)# ip route-cache flow
Router(config-if)# ip flow-export source vlan1
Router(config-if)# ip flow-export version 5
Router(config)# ip flow-export destination 10.1.1.99 9999
```

■ Sflow su Juniper switch

```
[edit protocols sflow]
user@switch# set collector <ip-address>
set collector udp-port <port-number>
set interfaces interface-name
set polling-interval seconds
set sample-rate number
```

Open vSwitch (sflow)



```
ovs-vsctl -- --id=@sflow create sflow agent=${AGENT_IP}  
target="\${COLLECTOR_IP}:${COLLECTOR_PORT}" header=${  
{HEADER_BYTES} sampling=${SAMPLING_N} polling=${  
{POLLING_SECS} -- set bridge ${BRIDGE}
```

- AGENT_IP = interfaccia verso il collettore
- COLLECTOR_IP/PORT = indirizzo/porta collettore
- HEADER_BYTES = bytes di header
- SAMPLING_N = sampling rate
- POLLING_SECS = intervallo di campionatura
- BRIDGE = nome bridge

Open vSwitch (netflow)



```
ovs-vsctl -- set bridge ovsbr1 netflow=@nf -- --id=@nf create  
NetFlow targets=\"${COLLECTOR_IP}:${COLLECTOR_PORT}\" \  
active-timeout=${TIMEOUT}
```

- COLLECTOR_IP/PORT=indirizzo/porta collettore netflow
- TIMEOUT= timeout flussi

- **Scopo:**

- Esportazione dei flussi su sistemi linux (es. firewall/NAT)
 - Controllo del traffico interno al NAT o di singola macchine

- **Come:**

- Utilizzando un'applicazione che crea pacchetti NetFlow analizzando i pacchetti entranti e uscenti dalle interfacce in modalita' promiscua

- **Cosa:**

- nProbe
- flowprobe
- Ipt-netflow (NetFlow iptables module)
- Fprobe-ulog

- **Architettura:**

La macchina linux con il probe installato agisce come un router, esportando i pacchetti NetFlow verso il collector (nel nostro caso sempre `nfcapd`)

Abbiamo scelto e testato **nProbe** per le caratteristiche di performances, stabilita' e affidabilita'

E' opensource per gli enti accademici e di ricerca

Installazione nProbe e collector 1/2



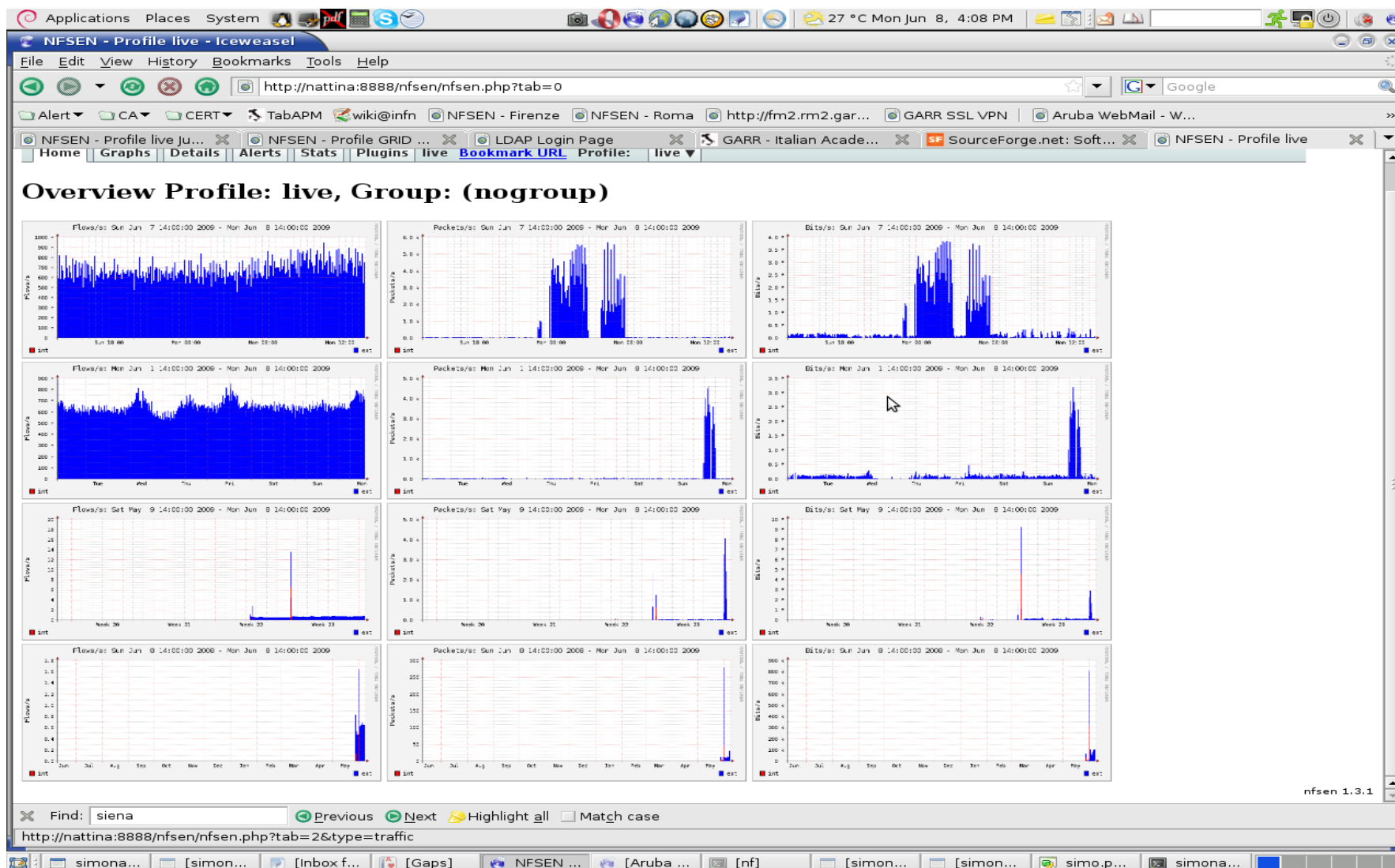
- Sulla macchina linux che fa NAT
 - Download nProbe
 - `tar -zxvf nprobe-XXX.tgz`
 - `./autogen.sh && make && make install`
 - Comandi d'esportazione:
 - `nprobe -G -i eth0 -n localhost:10000`
 - `nprobe -G -i eth1 -n localhost:10001`

- Installazione Nfsen/nfdump
- Configurazione Nfsen
 - Aggiungere a ./etc/nfsen.conf le righe relative alle due interfacce del NAT

```
%sources = (  
  'ext'  => { 'port'  => '10000', 'col' => '#0000ff', 'type' => netflow' },  
  'int'   => { 'port'  => '10001', 'col' => '#ff0000' },  
);
```

`./install.pl ./etc/nfsen.conf`

Risultato



Dettagli



Processing

The screenshot displays the NfSen web interface in a browser window. The main content area shows a table of statistics for the time period Jun 01 2009 - 08:20 to Jun 01 2009 - 08:55. Below this, the 'Netflow Processing' section is active, showing a filter for 'ip 131.175.1.35' and various options for displaying the data. The bottom of the interface shows the output of the processing, including a list of top 500 flows ordered by bytes and a summary of the total flows and bytes processed.

Channel:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:
ext	13.2 /s	12.7 /s	0.4 /s	0.1 /s	0 /s	1.4 k/s	1.4 k/s	0.6 /s	0.1 /s	0 /s	10.1 Mb/s	10.1 Mb/s	876.9 b/s	41.2 b/s	0 b/s
int	12.6 /s	12.6 /s	0.1 /s	0.0 /s	0 /s	1.4 k/s	1.4 k/s	0.1 /s	0.0 /s	0 /s	10.1 Mb/s	10.1 Mb/s	49.1 b/s	1.4 b/s	0 b/s

Netflow Processing

Source: ext, int
Filter: ip 131.175.1.35
Options: List Flows, Stat TopN
Top: 500
Stat: Flow Records, order by bytes
Aggregate: proto, srcPort, dstPort
Limit: Packets, 0
Output: long, / IPv6 long
Clear Form, process

```
** nfdump -M /var/nfsen/profiles-data/live/int -T -R 2009/06/01/nfcpd.200906010820:2009/06/01/nfcpd.200906010855 -n 500 -s record/bytes -A srcip,dstip
nfdump filter:
ip 131.175.1.35
Aggregated flows 4
Top 500 flows ordered by bytes:
Date flow start      Duration Proto      Src IP Addr:Port      Dst IP Addr:Port      Flags Tos  Packets  Bytes  Flows
2009-06-01 08:25:26.429 2048.893 0      131.175.1.35:0      ->      10.0.0.3:0      .AP.SF 0      224880 234.9 M 14658
2009-06-01 08:25:26.422 2048.900 0      131.175.1.35:0      ->      10.0.0.3:0      .APRSF 0      224066 15.2 M 14659
2009-05-28 15:24:22.634 0.844 0      131.175.1.35:0      ->      10.0.0.2:0      .AP.SF 0      82 90042 5
2009-05-28 15:24:22.626 0.674 0      10.0.0.2:0      ->      131.175.1.35:0      .AP.SF 0      62 4954 4

Summary: total flows: 29326, total bytes: 250.2 M, total packets: 449090, avg bps: 6506, avg pps: 1, avg bpp: 584
Time window: 2009-05-28 15:24:22 - 2009-06-01 08:59:35
Total flows processed: 30305, Records skipped: 0, Bytes read: 1575956
Sys: 0.004s flows/second: 7576250.0 wall: 0.003s flows/second: 9420267.3
```

Estendere le capacita' della suite



- scripting + nfdump
 - -o pipe
 - -o csv
- nfsen plugins

Nfdump -o pipe

- Nfdump prevede un output machine readable tramite l'opzione: *-o pipe*
- Il formato di output e' riportato nella tabella, ogni campo e' separato da un "pipe"(|).
- Gli indirizzi IP sono rappresentati da numeri interi.
- L'uso di filtri e dell'aggregazione non modifica il formato di output, ma solo il riempimento dei campi.

Address family	PF_INET or PF_INET6
Time first seen UNIX time seconds	
msec first seen Mili seconds first seen	
Time last seen UNIX time seconds	
msec last seen Mili seconds first seen	
Protocol	Protocol
Src address	Src address as 4 consecutive 32bit numbers
Src port	Src port
Dst address	Dst address as 4 consecutive 32bit numbers.
Dst port	Dst port
Src AS	Src AS number
Dst AS	Dst AS number
Input IF	Input Interface
Output IF	Output Interface
TCP Flags	000001 FIN. 000010 SYN 000100 RESET 001000 PUSH 010000 ACK 100000 URGENT
Tos	Type of Service
Packets	Packets
Bytes	Bytes

Nfdump -o pipe



■ Esempio di output:

2|1242719921|778|1242719921|778|6|0|0|0|3235805120|52616|0|0|0|1134240473|1863|24869|8075|114|88|24|0|1|45

- In cui: 3235805120 Corrisponde a: 192.222.119.192
- 24869 e' AS sorgente e 8075 quello destinazione
- 114 e' l'snmp ifindex di ingresso e 88 l'snmp ifindex di uscita
- 1 e' il pacchetto che contiene 45 bytes

Nfdump -o csv



- Su prima riga nome campi
- Tutti i campi del flusso (lista flussi), alcuni campi specifici per query aggregate
- IP in notazione decimale (IPv6 esadecimale), Date formato YYYY-MM-DD hh:mm:ss, volume di traffico in bytes

Come scrivere nuovi plugin

- Nfsen plugins:

Ogni 5 minuti, quando vengono scritti i flussi su disco e sono riempite le strutture RRD, nfsen lancia i plugin installati.

Sono composti da due parti:

1. Backend, scritti in linguaggio perl, hanno lo scopo di elaborare i dati
2. Frontend, scritti in linguaggio php, fanno da interfaccia al backend

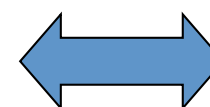
- I file di un plugin devono avere lo stesso nome:

- Backend: <nome_plugin>.pm
- Frontend: <nome_plugin>.php

Post Processing /
Alerting

Web frontend
(opzionale)

Backend
Plugin

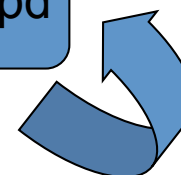


Frontend
Plugin



Controllo ed esecuzione
dei plugin Installati ed attivi

Nfsen/nfcapd



Ogni 5 minuti

- Esempi di configurazione esportazione sui router Juniper e Cisco:
 - JunOS v5
 - JunOS v9
 - Cisco v5
 - Cisco v9
 - Cisco FNF

■ Definizione del filtro:

```
set firewall family inet filter NETFLOW-SAMPLE term default then sample  
set firewall family inet filter NETFLOW-SAMPLE term default then accept
```

■ Configurazione interfacce:

```
set interfaces fe-0/1/0 unit 0 family inet filter input NETFLOW-SAMPLE  
set interfaces fe-0/1/0 unit 0 family inet filter output NETFLOW-SAMPLE
```

■ Configurazione collezionatore

```
set forwarding-options sampling input family inte rate 1000  
set forwarding-options sampling output cflowd 172.16.0.1 port 20000  
set forwarding-options sampling output cflowd version 5  
set forwarding-options sampling output cflowd autonomous-system-type [peer|origin]
```

Esempi di configurazione: JunOS (v9)



- Configurazione interfacce:

- set interfaces fe-0/1/0 unit 0 family inet filter input NETFLOW-SAMPLE*
 - set interfaces fe-0/1/0 unit 0 family inet filter output NETFLOW-SAMPLE*

- Configurazione esportazione e sampling:

- set forwarding-options sampling input family inet rate 1*
 - set forwarding-options sampling output cflowd 172.16.0.1 port 20000*
 - set forwarding-options sampling output cflowd 172.16.0.1 version9 template prova-template*
 - set forwarding-options sampling output interface sp-0/2/0 source-address 10.0.0.1*

- Configurazione del filtro:

- set firewall family inet filter NETFLOW-SAMPLE term default then sample*
 - set firewall family inet filter NETFLOW-SAMPLE term default then accept*

- Configurazione interfacce

interface FastEthernet0/0

ip route cache flow

- Esportazione dei flussi

ip flow-export

ip flow-export version 5

ip flow-export destination 172.16.0.1 20000

ip flow-cache timeout active 5

ip flow-cache timeout inactive 1000

Esempi di configurazione: Cisco(v9)



- **Configurazione Sampling:**
flow-sampler-map sampling1-1000
mode random one-out-of 1000
- **Configurazione interfacce:**
interface X/X/X
flow-sampler sampling1-1000
- **Configurazione collettore:**
ip flow-capture fragment-offset
ip flow-capture packet-length
ip flow-capture vlan-id
ip flow-capture icmp
ip flow-capture mac-addresses
ip flow-export version 9 origin-as bgp-nexthop
ip flow-export destination 172.16.0.1 20000

■ Definizione template:

```
flow record template_prova  
match ipv4 source address  
match ipv4 source mask  
match ipv4 destination address  
match ipv4 destination mask  
match transport source-port  
match transport destination-port  
match transport tcp source-port  
match transport tcp destination-port  
match interface input  
match interface output  
collect datalink mac source address input  
collect routing next-hop address ipv4 bgp  
collect ipv4 section payload size 100
```

■ Definizione del monitor:

flow monitor prova_monitor

record template_prova

exporter prova_exporter

cache timeout active 30

statistics packet protocol

statistics packet size

■ Definizione exporter

flow exporter prova_exporter

destination 172.16.0.1

transport udp 20000

Esempi di configurazione: Cisco(FNF)



- Configurazione Interfacce

interface FastEthernet0/0

ip address X.X.X.X X.X.X.X

*ip flow monitor **prova_monitor** sampler **prova_SAMPLER** input*

*ip flow monitor **prova_monitor** sampler **prova_SAMPLER** output*

- Configurazione Sampler

sampler prova_SAMPLER

mode deterministic 1 out-of 1000

exit

Domande?

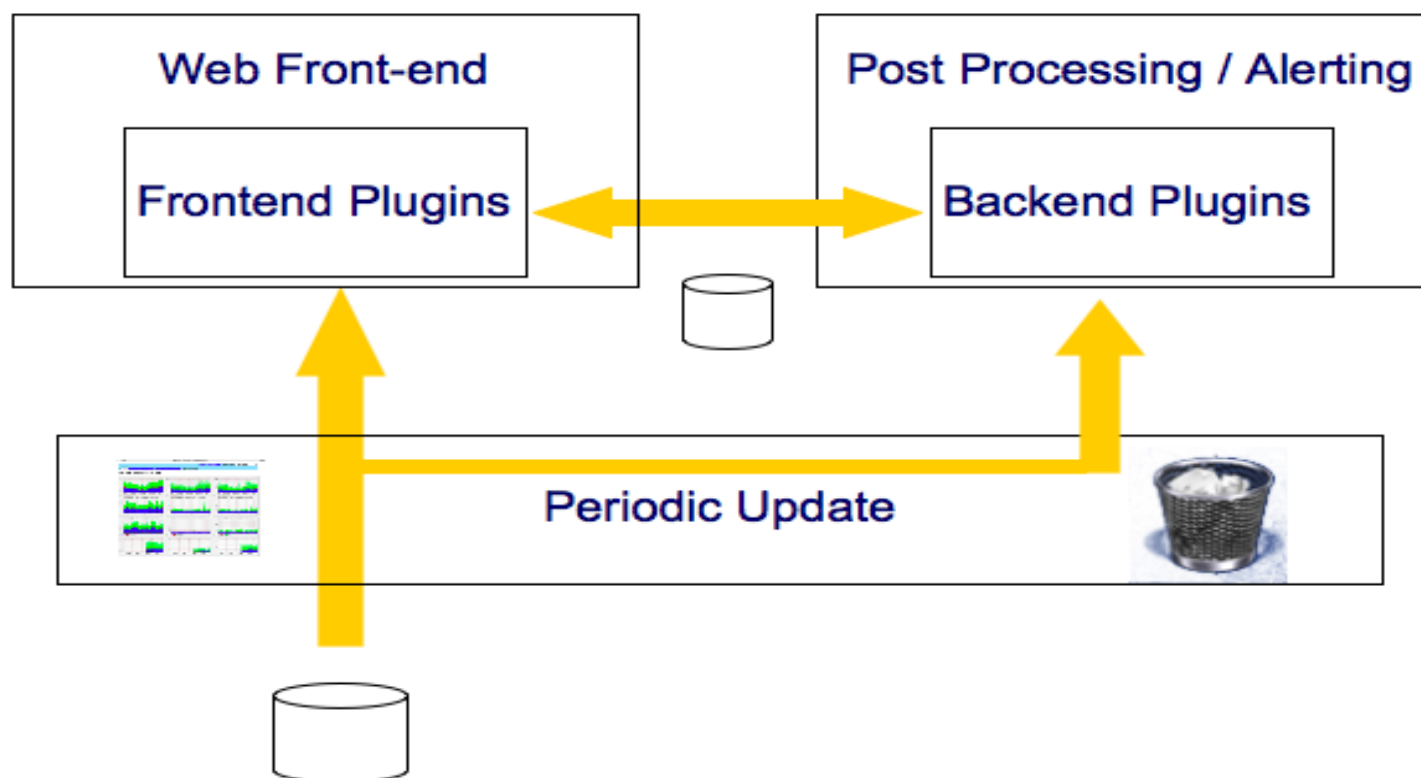


Grazie

nino.ciurleo@garr.it

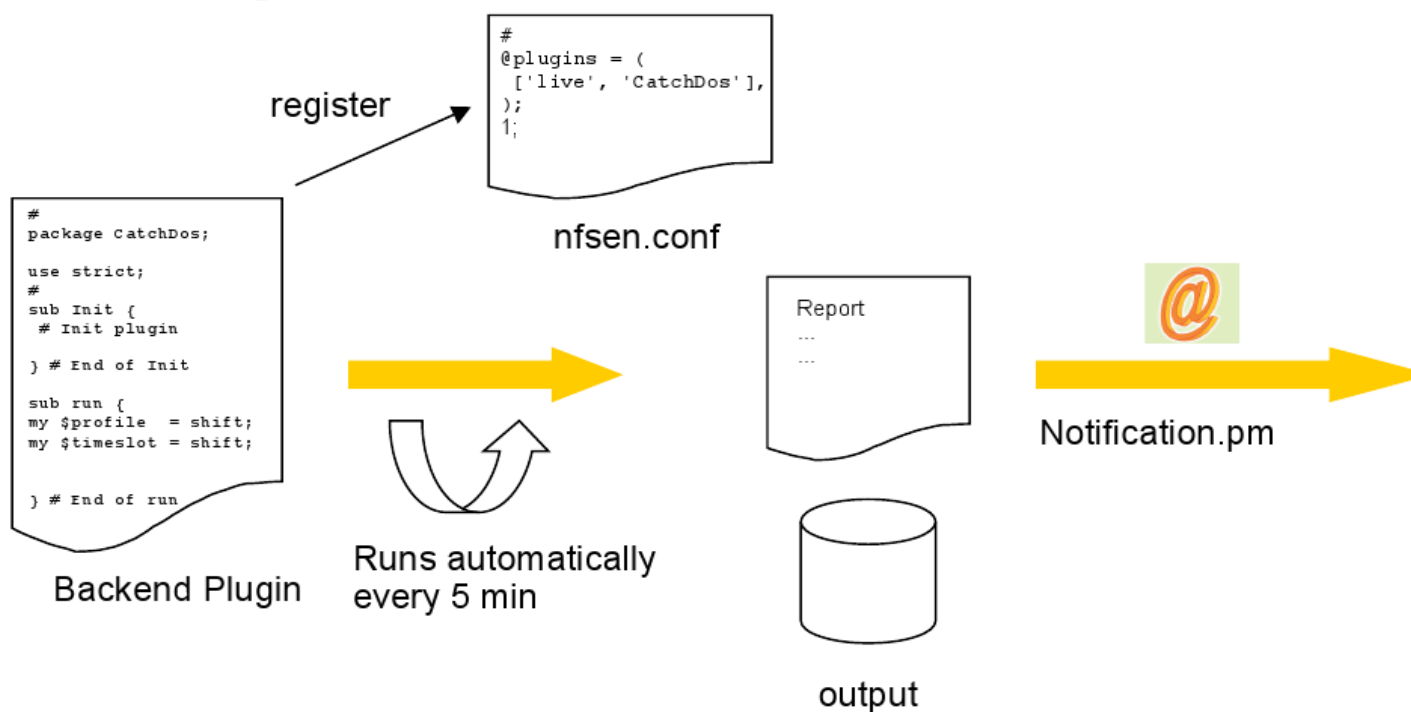
sw.dev@garr.it

Plugin: architettura (1/3)



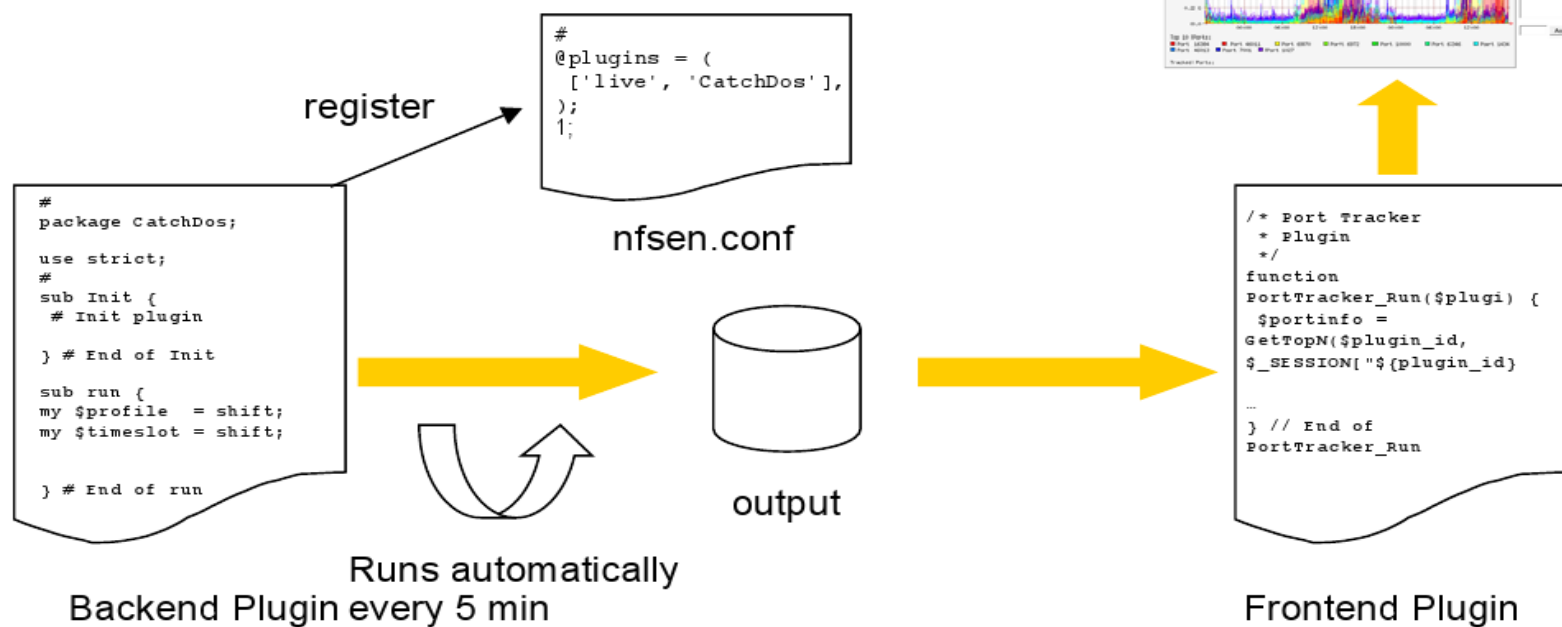
Plugin: architettura (2/3)

Backend Plugins:



Plugin: architettura (3/3)

Backend / Frontend Plugins:



Funzioni predefinite:

Init

- Questa funzione viene chiamata quando il plugin viene caricato. Lo scopo e' di dare al plugin la possibilita' di inicializzarsi.
- I valori di ritorno sono 0 in caso di successo e 1 in caso contrario.
- Nel caso la funzione Init fallisca il plugin non viene caricato.

Es.

```
sub Init {  
    return 0;  
}
```

Cleanup

- Questa funzione viene chiamata nel momento in cui nfsend viene fermato. Lo scopo e' di dare la possibilita' al plugin di terminare in maniera corretta.

Es.

```
sub Cleanup {  
    syslog("info", "plugin Terminato");  
}
```

Backend plugin



Run

Questa funzione viene chiamata da nfsen ogni volta che ci sono nuovi flussi da analizzare.

- E' il cuore del plugin.

Es.

```
sub Run {  
  my $argref = shift;  
  my $profile = $$argref{'profile'}; #il nome del profilo sul quale e' installato  
  my $profilegroup = $$argref{'profilegroup'}; #gruppo di profili  
  my $timeslot = $$argref{'timeslot'}; #il timeslot corrente (es. 200905201015)  
  
  # qui dovrebbe essere inserito il codice del plugin  
}
```

Altre variabili disponibili nella funzione Run:

```
my $profilepath = NfProfile::ProfilePath($profile, $profilegroup); #il path su disco che contiene i dati del profilo  
my $all_sources = join ':', keys %{$profileinfo{'channel'}}; # la stringa con tutte i router sorgenti separate da ":"  
my $netflow_sources = "$NfConf::PROFILEDATADIR/$profilepath/$all_sources"; # la stringa che contiene sia il path che le sorgenti (da passare a nfdump con l'opzione -R)  
my %profileinfo = NfProfile::ReadProfile($profile, $profilegroup); #un hash table che contiene tutti i dati relativi al profilo in uso
```

Frontend plugin (opzionale)



- Funzioni predefinite:

- <nome_plugin>_ParseInput** (obbligatoria)

- E' chiamata prima di ogni output sul browser ed ha la funzione di parsare i dati dei form.
 - Questa funzione viene chiamata solamente quando sull'interfaccia di nfsen e' selezionato il tab dei Plugin
 - Puo' essere utilizzata per ritornare messaggi

Es.

```
function pluginprova_ParseInput( $plugin_id ) {  
    SetMessage('error', "Errore dal plugin di prova!");  
    SetMessage('warning', "Warning dal plugin di prova!");  
    SetMessage('alert', "Alert dal plugin di prova!!");  
    SetMessage('info', "Info dal plugin di prova!");  
}
```

- <nome_plugin>_Run** (obbligatoria)

- Questa funzione viene chiamata dopo che sono stati spediti al browser l'header della pagina e la barra di navigazione

Es.

```
function <nome_plugin>_Run( $plugin_id ) {  
    // qui va il codice  
}
```

Comucazione tra backend e frontend (1/2)



■ Nel backend

Al fine di potere eseguire una funzione del plugin di backend da quello di frontend, occorre definire alcune variabili.

■ Gestione delle richieste del frontend:

```
our %cmd_lookup = (  
  'try' => \&RunProc, #la funzione RunProc del backend si chiamera' try nel frontend.  
);  
sub RunProc {  
  my $socket = shift; #la socket di comunicazione con il frontend  
  my $opts = shift; # contiene i parametri  
} # End of RunProc
```

■ Ritorno dei risultati al frontend:

```
$args{'string'}="Ciao dal Backend"  
Nfcomm::socket_send_ok($socket, \%args);  
Nfcomm::socket_send_error($socket, "Errore");
```

Comucazione tra backend e frontend (2/2)



■ Nel frontend:

■ Richieste al backend:

```
$out_list = nfsend_query("Pluginname::try", $opts);
```

- Il primo parametro indica il nome della funzione del backend da chiamare
- il secondo e' un hash table che contiene i parametri
- \$out_list e' l'hash table con i valori di ritorno

■ Risposte del backend:

```
print $out_list['string'];
```

Esempio completo

Frontend:

```
// Richieste al backend:

$opts = array();

// due scalari
$opts['colour1'] = '#72e3fa';
$opts['colour2'] = '#2a6f99';

// un array
$opts['colours'] = array ( '#12af7d', '#56fc7b' );

// chiamata alla funzione in backend
$out_list = nfsend_query("Pluginname::try", $opts);

//Risposte del backend:

// if $out_list == FALSE – it's an error
if ( !is_array($out_list) ) {
    SetMessage('error', "Errore del plugin di Backend");
    return FALSE;
}

$string = $out_list['string'];
$othercolours = $out_list['othercolours'];

print "Il Backend ha risposto: <b>$string</b><br>\n";
```

Backend:

#Gestione delle richieste del frontend:

```
our %cmd_lookup = (
    'try' => \&RunProc,
);
sub RunProc {
    my $socket = shift;
    my $opts = shift; # riferimento ad un hash
```

```
# Lettura dati del frontend
#due scalari
```

```
my $colour1 = $$opts{'colour1'};
my $colour2 = $$opts{'colour2'};
```

```
# un puntatore ad array
my $colours = $$opts{'colours'};
```

#Ritorno dei risultati al frontend:

```
# esempio controllo d'errore
if ( !exists $$opts{'colours'} ) {
    Nfcomm::socket_send_error($socket, "Missing value");
    return;
}
my %args;
my @othercolours = ( 'rosso', 'blu' );
$args{'string'} = "Ciao dal plugin di backend."
$args{'othercolours'} = \@othercolours;

Nfcomm::socket_send_ok($socket, \%args);
} # fine della funzione RunProc
```

Condizione di Alert computata da un plugin:

- Per far saltare un Alert in base ad una condizione complessa, si usa un plugin per matcharla.
- La funzione **alert_condition** viene lanciata da nfsen dopo l'applicazione dei filtri ai flussi.

```
sub alert_condition {  
    my $argref = shift;  
    my $alert    = $$argref{'alert'};  
    my $alertflows = $$argref{'alertfile'};  
    my $timeslot = $$argref{'timeslot'};  
    syslog('info', "Alert condition called: alert: $alert, alertfile: $alertflows, timeslot: $timeslot");  
    # Add your code here  
    return 1;  
}
```

- Impostare un'azione complessa scaturita da un alert: si usa un plugin con la funzione **alert_action**

```
sub alert_action {  
  my $argref = shift;  
  my $alert   = $$argref{'alert'};  
  my $timeslot = $$argref{'timeslot'};
```

```
  syslog('info', "Alert action function called: alert: $alert, timeslot: $timeslot");  
  # Add your code here  
  return 1;  
}
```

La funzione ParseForm



- Nfsen fornisce una funzione che consente di accedere ai dati presenti nell'array \$_POST di php: ParseForm.
- Essa ritorna una lista con i dati e gli errori:

list (\$process_form, \$has_errors) = ParseForm(\$parse_opts);

\$parse_opts e' un array che contiene tanti elementi quanti sono gli elementi di \$_POST che si vogliono parsare.

Ad ogni elemento e' associato un array associativo che descrive le opzioni legate al parametro; le chiavi dell'array sono:

- Required (0 o 1)
- Allow_null (0 o 1)
- Default (valore di default della variabile)
- Match (reg exp o array per verificarne il valore)
- Validate (nome della funzione per un'ulteriore validazione del valore)
function function_validate(&\$var,\$opts)
&\$var e' il puntatore al valore da validare
\$opts e' l'array di parametri di quel valore.

Esempio: plugin DoS



- Cosa fa:
 - Plugin di alert che ogni volta che il traffico UDP supera una certa soglia definita come DoS, riempie una pagina web con l'ip della macchina GARR coinvolta nel DoS e l'orario in cui e' avvenuto

Esempio: plugin DoS



- Come funziona:
 - un plugin di backend: calcola il traffico UDP e lo confronta con delle soglie stabilite precedentemente
 - Un plugin di alert: se la soglia e' superata fa scattare alcuni script che calcolano gli IP coinvolti, estrapola l'ip GARR e dati di interesse
 - Un plugin di frontend: visualizza una tabella HTML con l'ip GARR e i dati relativi all'attacco DoS

Esempio: plugin DoS



- Backend plugin NOC.pm:

```
package NOC;
my $nf_filter = 'proto UDP';
my @output =    `$nfdump -M $netflow_sources -T \
                -r nfcapd.$timeslot -n 100 -s ip/flows \
                '$nf_filter`;
system ("/mnt/nfsen/plugins/NOC_creo_tabella 2>
        /mnt/nfsen/plugins/appoggio.noc")
```

Esempio: plugin DoS

- Alert plugin: NOC_alert.pm:

```
package NOC_alert;  
my $nf_filter = 'proto UDP';  
my @output =    `$nfdump -M $netflow_sources -T -r \  
                nfcapd.$timeslot -n 10 -s ip/flows \  
                '$nf_filter`';  
system ("/mnt/nfsen/plugins/elaboro_NOC $timeslot \  
2> /mnt/nfsen/plugins/log_NOC");
```

Esempio: plugin DoS

- Frontend plugin \$http_dir/plugins/NOC.php:

```
<?php
/* NOC plugin*/
function NOC_ParseInput( $plugin_id ) {}
function NOC_Run( $plugin_id ) {
$VARDIR = "/mnt/nfsen/www/plugins";
print "<h3>NOC Logfile</h3>\n";
$logfile = "$VARDIR/NOC_output.log";
echo htmlspecialchars($logfile) ;
include $logfile;
print "</pre>\n";
} // End of NOC_Run
?>
```

Esempio: plugin DoS

■ Configurazione alert:

Home Graphs Details Alerts Stats Plugins live [Bookmark URL](#) Profile: live ▼

Alerts details: NOC_alert

Trigger	Status	Last Triggered
fired	☒ enabled	2009-06-08-16:55

Filter applied to 'live' profile:

rtнал
rtпil
rtfi
rtbol

proto UDP

☒ **Conditions based on total flow summary:**

0	Total flows	>	Absolute value	5	
1	and	Flows/s	>	24 hour average value	20

☐ **Conditions based on individual Top 1 statistics:**

☐ **Conditions based on plugin:**

Trigger:

Each time after 1 x condition = true, and block next trigger for 0 cycles

Action:

☐ No action

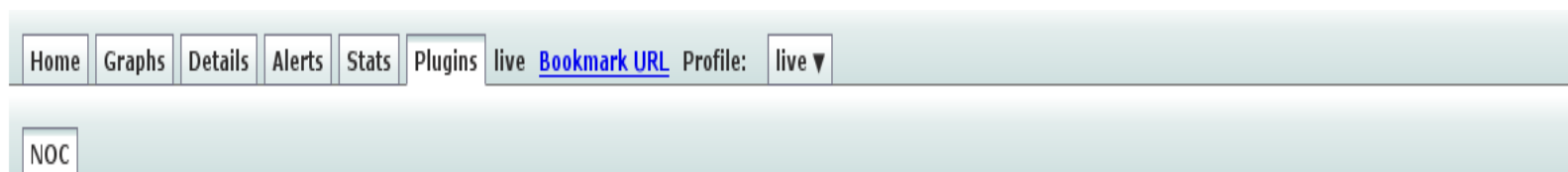
☐ Send alert email To:

Subject: Alert triggered

☒ Call plugin: NOC_alert

Esempio: plugin DoS

■ RISULTATO:



NOC Logfile

/mnt/nfsen/www/plugins/NOC_output.log

**Questi sono i DoS sulla rete GARR della giornata
aggiornati a Mon Jun 8 17:05:39 CEST 2009**

Num	Timeslot	Data/Ora	Duration	IP	Indirizzo	Flussi	Pacchetti	Bytes	Pkts/sec	Bytes/sec	Bytes per pkt
1	200906081645	Mon Jun 8 16:45:43 CEST 2009	299.605	192.167.90.2	dns2.iue.it	6329	6393	885474	21	23643	138
2	200906081650	Mon Jun 8 16:50:43 CEST 2009	486.088	212.189.201.89		5627	17774		36	137008	468
3	200906081655	Mon Jun 8 16:55:43 CEST 2009	300.081	192.167.90.2	dns2.iue.it	6371	6435	890134	21	23730	138
4	200906081700	Mon Jun 8 17:00:43 CEST 2009	329.689	192.167.90.2	dns2.iue.it	5919	5982	836961	18	20309	139

Mon Jun 8 17:05:39 CEST 2009

nfsen 1.3b-20070824

1. Quale protocollo si usa per l'esportazione dei flussi sugli switch:
 1. Netflow
 2. Sflow (CORRETTA)
 3. Sugli switch non e' possibile esportare flussi
2. Nel protocollo Neflow un flusso e':
 1. Unidirezionale (CORRETTA)
 2. Bidirezionale
 3. Uguale ad un pacchetto
3. I profili di nfsen servono per:
 1. Generare allarmi in base alle condizioni verificatesi nei flussi
 2. Ottenere una rappresentazione grafica di un particolare tipo di traffico (CORRETTA)
 3. Estendere le funzionalita' di nfsen
4. I plugins di nfsen servono per:
 1. Generare allarmi in base alle condizioni verificatesi nei flussi
 2. Ottenere una rappresentazione grafica di un particolare tipo di traffico
 3. Estendere le funzionalita' di nfsen (CORRETTA)