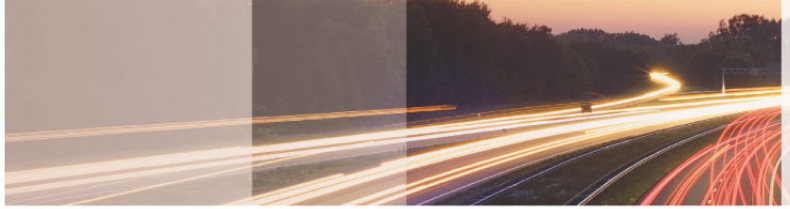


GARR

The Italian Academic & Research Network

www.garr.it

Tutorial Monitoring

Monitorare la rete con NetFlow

Nino Ciurleo, Alessandro Inzerilli, Simona Venuti

GARR WS9, Roma, 15.06.2009

Consortium GARR

Agenda

- Perché monitorare la rete con NetFlow
- Protocolli per l'esportazione dei flussi NetFlow
- Architettura del sistema di analisi dei flussi
- Suite Nfsen/Nfdump
- Utilizzo avanzato di Nfsen/Nfdump
- Monitoraggio della LAN con NetFlow
- Estendere le funzionalità di Nfsen/Nfdump

Perché monitorare la rete con NetFlow?

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

Perché monitorare la rete con NetFlow?

- Ogni APM potrebbe trovarsi davanti a domande di questo tipo nell'attività di tutti i giorni .....
  - Cosa ha causato questo picco nelle statistiche di traffico?
  - Quali sono i top talkers/ le top subnet della mia sede?
  - Quali sono le applicazioni piú utilizzate dai miei utenti?
  - Vedi traffico relativo a questo incidente avvenuto il 15/06/2009?
  - Puoi analizzare questo DoS? Da quali indirizzi IP parte?
- Gli strumenti tradizionali di monitoraggio basati sul protocollo SNMP non sono in grado di rispondere ...
- Tra i vari strumenti che possono venire in aiuto dell'APM c'è sicuramente NetFlow...

Limiti delle statistiche di traffico tradizionali

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

Possibilità offerte dall'analisi dei flussi NetFlow

- Attraverso il protocollo NetFlow è invece possibile ottenere informazioni più dettagliate sulla natura del traffico che attraversa i nostri gli apparati di rete (router, switch, server, etc.)

- Informazioni degli strati IP, di trasporto e non solo .....

Protocolli per l'esportazione dei flussi

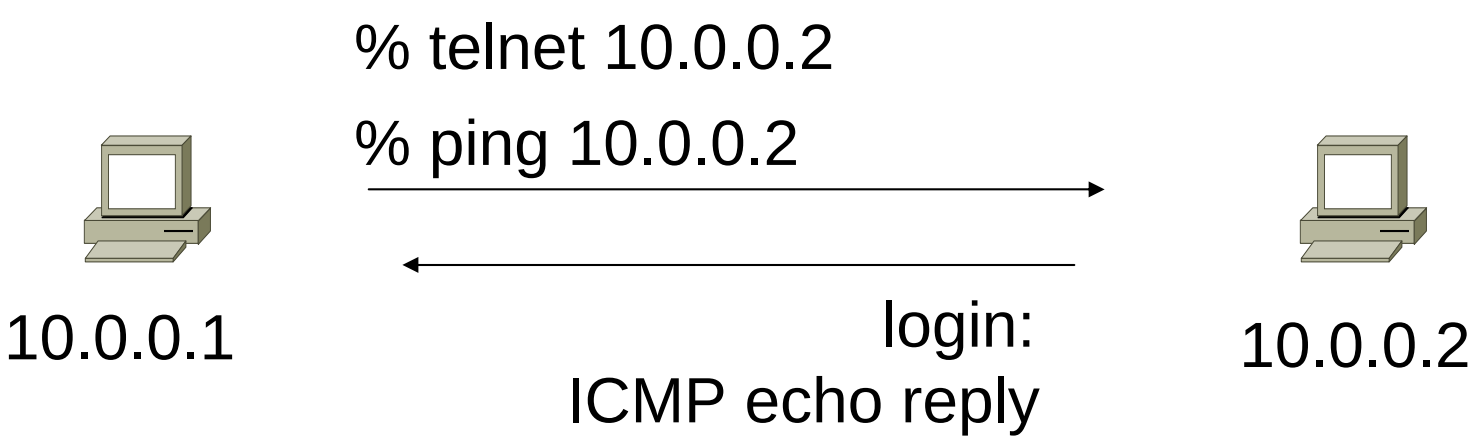
- Concetto di flusso NetFlow
- Versioni del protocollo NetFlow
  - NetFlow 5
  - NetFlow 9 (IPFIX)

Cos'è un flusso NetFlow?

- Per flusso si intende ogni comunicazione unidirezionale identificata da 7 campi degli header IP e UDP/TCP (definizione classica valida 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

Esempio di flusso



Active Flows

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

Protocolli per l'esportazione dei flussi

- NetFlow, inizialmente sviluppato da Cisco, è poi 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, supportata da quasi tutti i vendor e ancora la più utilizzata
  - V7, per gli switch catalyst della serie 5000
  - V8, come la 7 con in più la possibilità di esportare flussi aggregati
  - V9, più 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

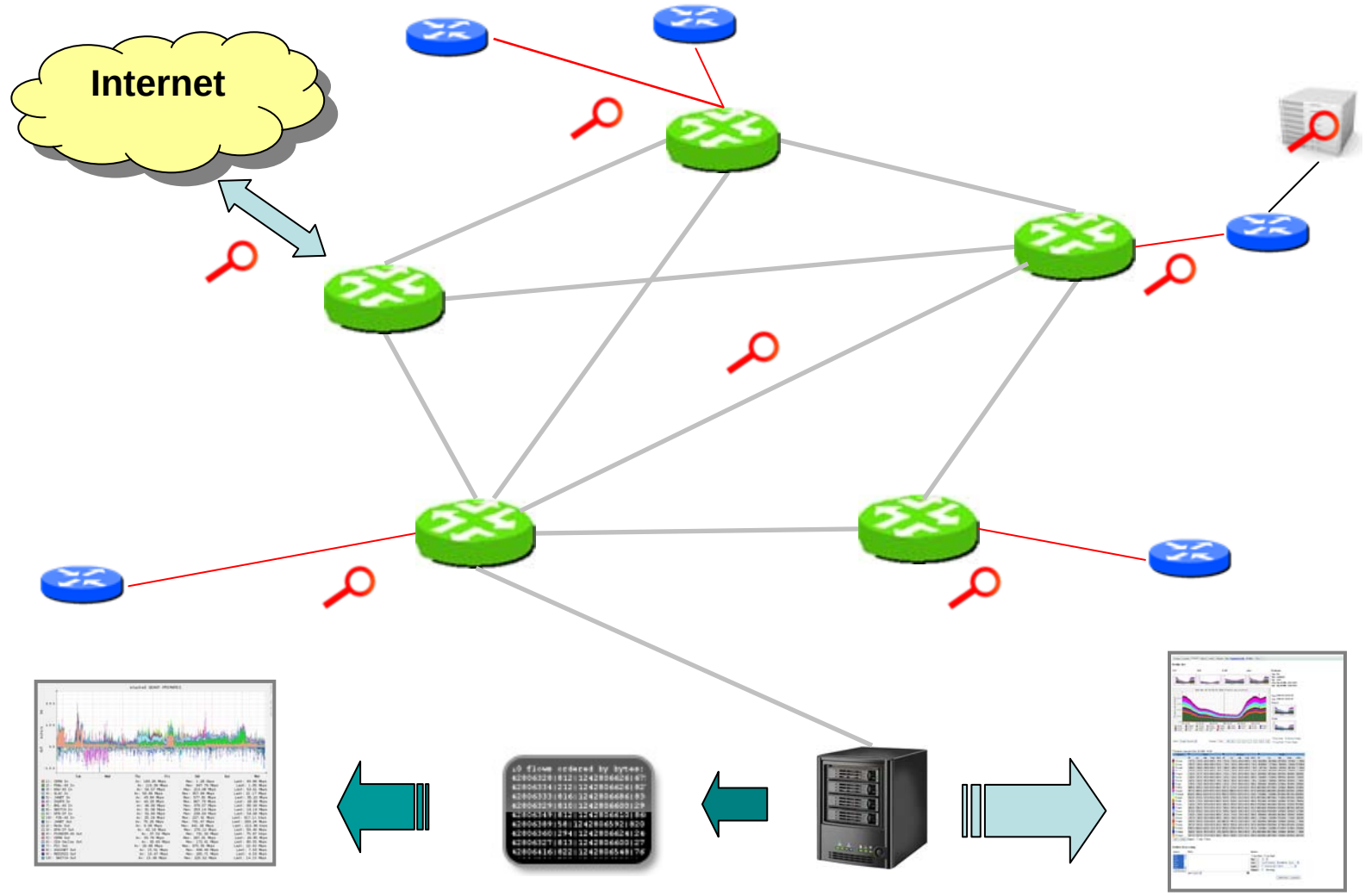
NetFlow versione 9

- Supporta IPv4, IPv6 ed MPLS
- informazioni trasportate sono:
  - Indirizzi 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
  - Tag VLAN
  - Label MPLS

Architettura del sistema di analisi dei flussi

- Esportazione dei flussi 
- Collezione  ed analisi 

Architettura del sistema di analisi dei flussi



Esportazione

- Il protocollo NetFlow (almeno la versone 5..) è supportato dalla maggior parte degli apparati di rete (Cisco, Juniper, Alcatel, etc..)
- è incluso nelle immagini degli OS degli apparati di rete (IoS, JunOS, etc.)
- Router e switch catturare i flussi di traffico, generano i pacchetti NetFlow e li esportano verso un collettore
- Server dotati di sw opportuno (es. nProbe) possono esportare flussi NetFlow
- Parametri da configurare sugli apparati:
  - Interfacce di cui si vogliono esportare i flussi
  - Indirizzo e porta del collettore
  - Versione di NetFlow
  - Timeout per l'esportazione (active e inactive timeout)
  - Eventuale tasso di campionamento
  - Template dei dati da esportare (solo NetFlow v9)

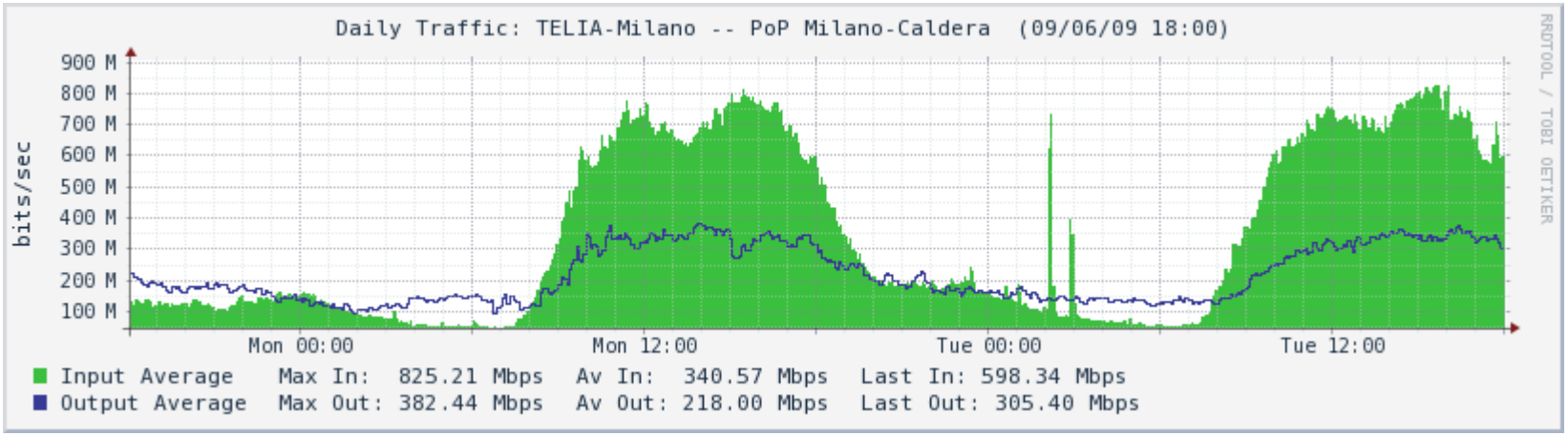
Sampling (1/2)

- La cattura dei flussi, la generazione e l'esportazione dei pacchetti NetFlow, a meno di HW dedicato, sono funzioni svolte dalla CPU dei router.
- Per evitare problemi dovuti al sovraccaricamento della CPU del router, Cisco ha introdotto il meccanismo del campionamento (sampling): invece di catturare tutti i pacchetti, il router prende un pacchetto ogni n, dove n è configurabile a piacere.
- Quando viene usato il sampling, il traffico visualizzato nei sistemi di analisi dei flussi è approssimato.
- Ma per la legge dei grandi numeri i conti tornano...

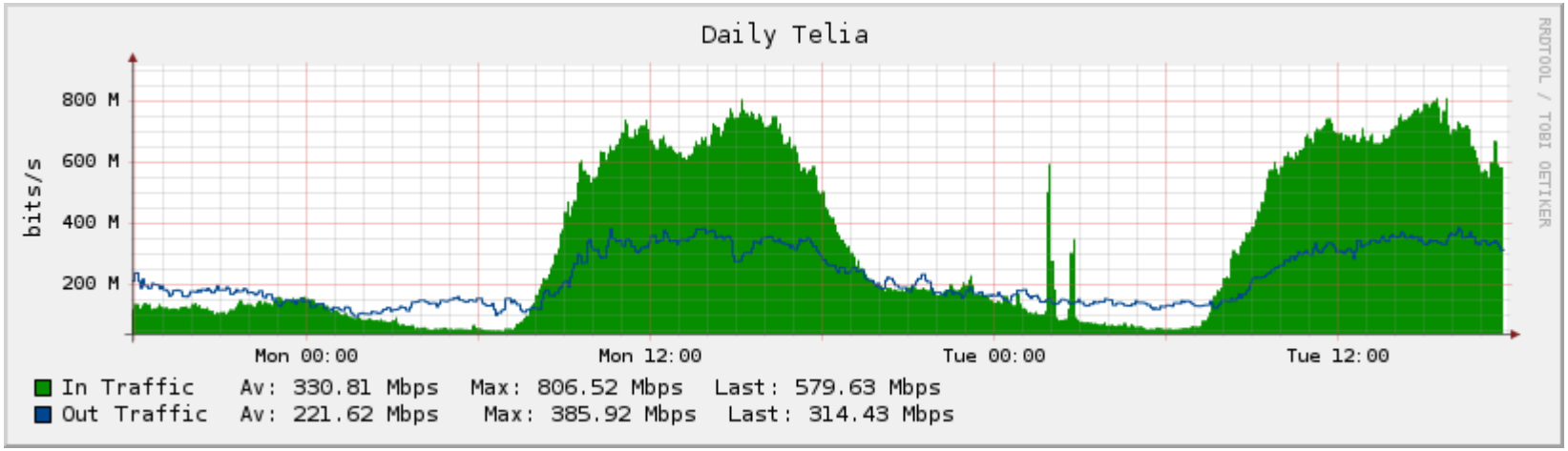
Sampling (2/2)

www.garr.it

Contatore del router



Analisi dei flussi con Sampling 1/1000 (ricostruito)



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 esportazione

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

Requisiti Hardware

- CPU
  - Minima: pentium 4 > 3Ghz
  - Consigliata: bi o quadri multicore cpu > 2Ghz
- RAM
  - Minima: 2GB
  - Consigliata: 4GB
- Dischi di alta capacità e ad accesso veloce
  - Da 40MB a 1.6G al giorno di dati per un router di trasporto (sampling 1/1000)
  - Dipende dalle esigenze

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

- 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>

Configurazione

- Nfsen.conf
  - $BASEDIR # directory base della suite
  - $BINDIR # directory dove di desiderano i binari
  - $HTMLDIR #root del webserver
  - $PROFILESTATDIR #dove devono essere salvati i dati
  - $BACK-END_PLUINGDIR #directory che ospita i plugin di back-end
  - $FRONTEND_PLUGINDIR #directory che ospita i front-end plugin
  - $SUBDIRLAYOUT #importante per motivi prestazionali
    - 0 default
    - 1 anno/mese/giorno
    - 2 anno/mese/giorno/ora
    - Etc..
  - %sources #hash table con le caratteristiche delle sorgenti (router, switch, server)
  - @plugins #array per l'associazione tra il profilo e il plugin installato

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
- Permette di lanciare un comando shell ogni volta che ha finito di scrivere un nuovo set di dati (con –x <comando>).
- Di solito viene gestito da Nfsen
- E' possibile lanciare manualmente Nfsen:
  - *Sintassi: nfcapd –w –D –l <source_name> -p <porta> -u <user> -g <group> -B <buffer size> -S <dir-hierarchy level> –l /<path>/source_name>*
  - *Esempio: nfcapd –w –D –l router1-p 12001 –B 200000 -S 1 –l /data/nfsen/profiles-data/live/router1*
- Non gestisce il sampling (in roadmap)

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 5,7 e 9
  - Supporta sFlow
  - 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)

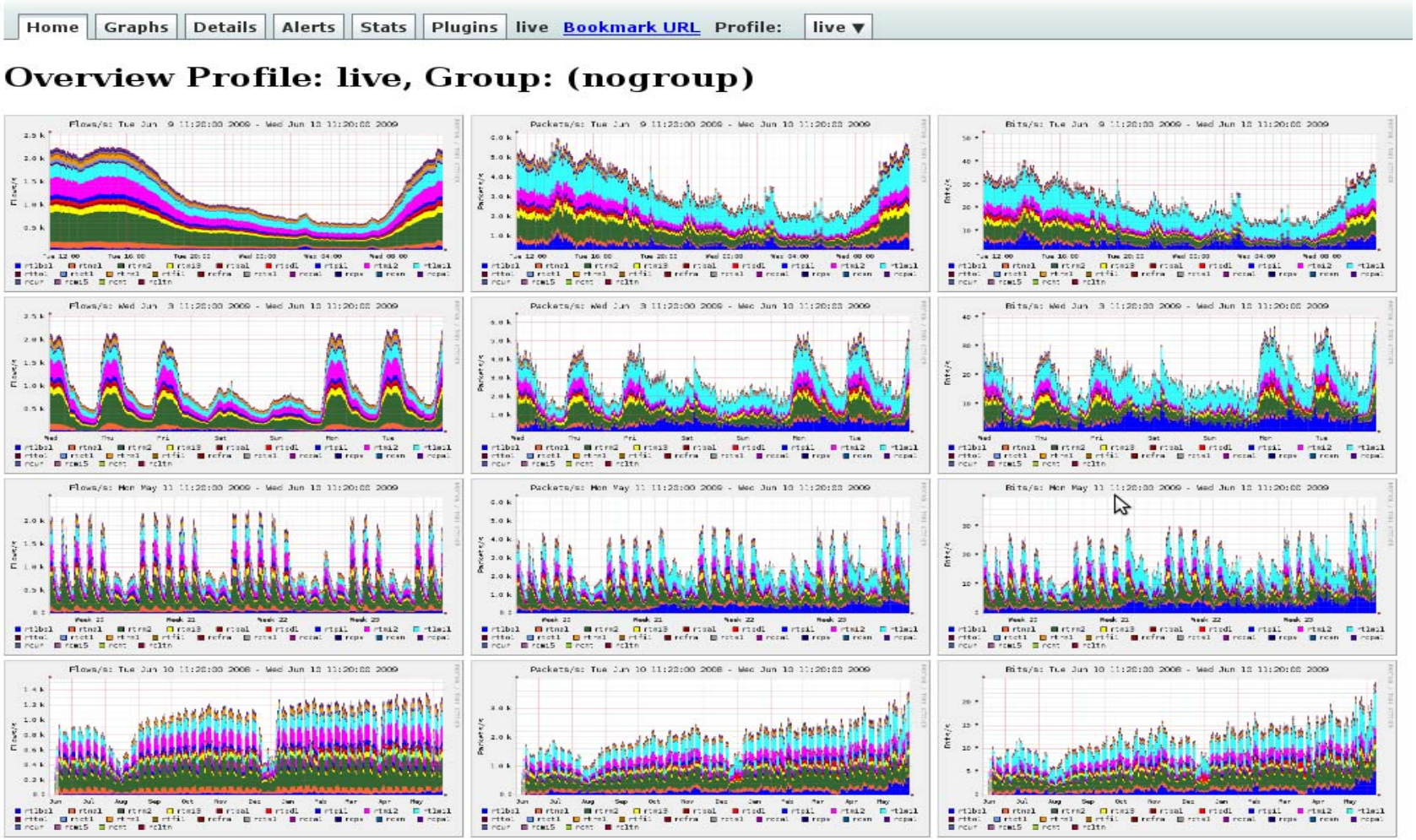
- Caratteristiche (continua ...):
  - Crea statistiche su aggregati di flussi (record) secondo un insieme di criteri, anche multipli, scelti a piacere (src/dst IP/NET, src/dst port, src/dst AS, protocol, ToS, interfacce, etc)
  - Permette la personalizzazione del formato di output e prevede un output machine readable per lo scripting
  - Consente l'anonimizzazione degli indirizzi IP tramite la libreria CryptoPan
  - è utilizzato in background da nfsen

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)

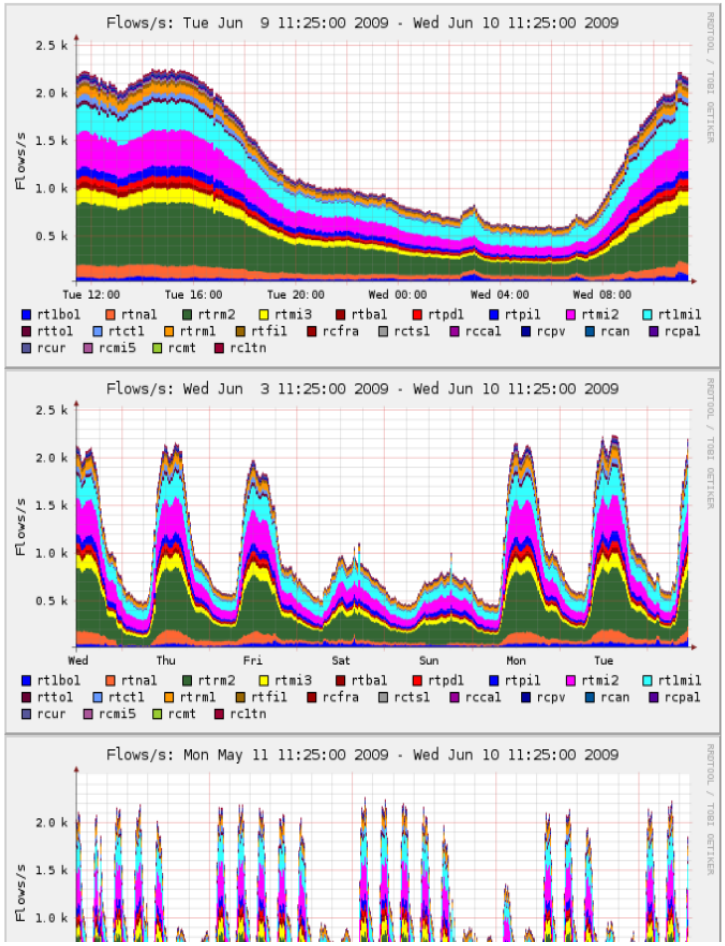
- Navigazione interfaccia

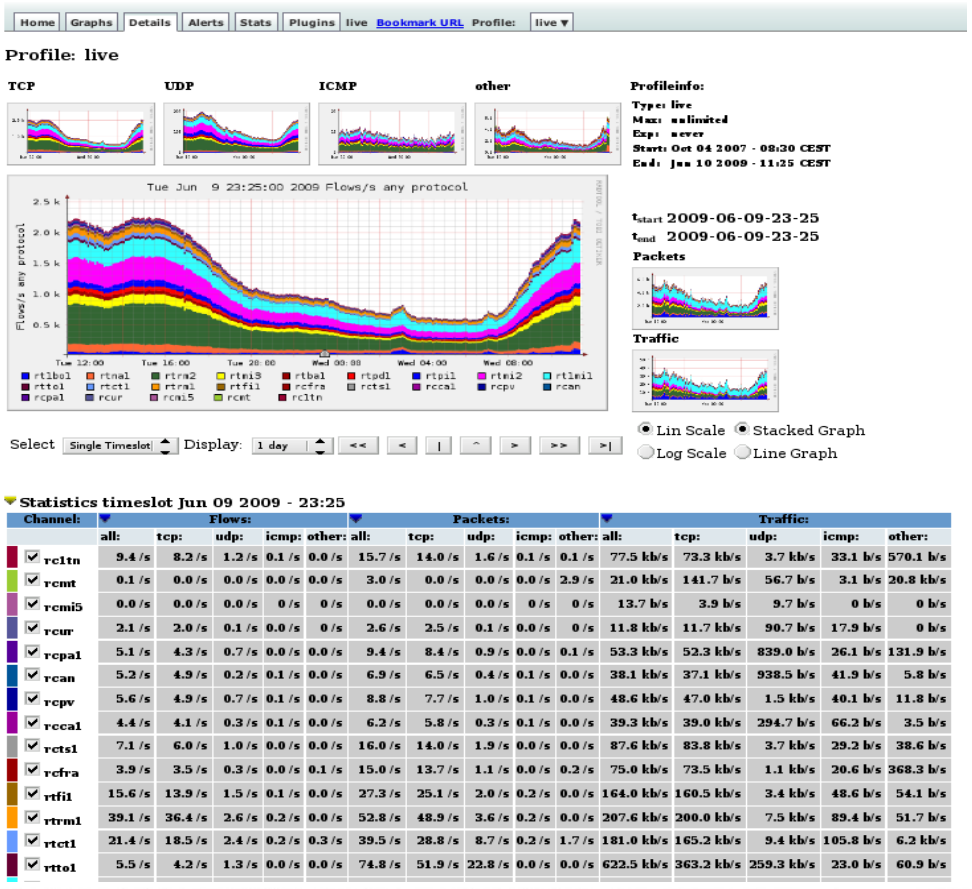


Interfaccia grafica di NfSen (2/3)

- Navigazione interfaccia – Dettagli

Profile: live, Group: (nogroup) - flows





Channel:	Flows:					Packets:					Traffic:				
	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:
rcltn	9.4 /s	8.2 /s	1.2 /s	0.1 /s	0.0 /s	15.7 /s	14.0 /s	1.6 /s	0.1 /s	0.1 /s	77.5 kb/s	73.3 kb/s	3.7 kb/s	33.1 b/s	570.1 b/s
rcmt	0.1 /s	0.0 /s	0.0 /s	0.0 /s	0.0 /s	3.0 /s	0.0 /s	0.0 /s	0.0 /s	2.9 /s	21.0 kb/s	141.7 b/s	56.7 b/s	3.1 b/s	20.8 kb/s
rcmi5	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	13.7 b/s	3.9 b/s	9.7 b/s	0 b/s	0 b/s
rcur	2.1 /s	2.0 /s	0.1 /s	0.0 /s	0 /s	2.6 /s	2.5 /s	0.1 /s	0.0 /s	0 /s	11.8 kb/s	11.7 kb/s	90.7 b/s	17.9 b/s	0 b/s
rcpal	5.1 /s	4.3 /s	0.7 /s	0.0 /s	0.0 /s	9.4 /s	8.4 /s	0.9 /s	0.0 /s	0.1 /s	53.3 kb/s	52.3 kb/s	839.0 b/s	26.1 b/s	131.9 b/s
rcan	5.2 /s	4.9 /s	0.2 /s	0.1 /s	0.0 /s	6.9 /s	6.5 /s	0.4 /s	0.1 /s	0.0 /s	38.1 kb/s	37.1 kb/s	938.5 b/s	41.9 b/s	5.8 b/s
rcpv	5.6 /s	4.9 /s	0.7 /s	0.1 /s	0.0 /s	8.8 /s	7.7 /s	1.0 /s	0.1 /s	0.0 /s	48.6 kb/s	47.0 kb/s	1.5 kb/s	40.1 b/s	11.8 b/s
rccal	4.4 /s	4.1 /s	0.3 /s	0.1 /s	0.0 /s	6.2 /s	5.8 /s	0.3 /s	0.1 /s	0.0 /s	39.3 kb/s	39.0 kb/s	294.7 b/s	66.2 b/s	3.5 b/s
rctsl	7.1 /s	6.0 /s	1.0 /s	0.0 /s	0.0 /s	16.0 /s	14.0 /s	1.9 /s	0.0 /s	0.0 /s	87.6 kb/s	83.8 kb/s	3.7 kb/s	29.2 b/s	38.6 b/s
rcfra	3.9 /s	3.5 /s	0.3 /s	0.0 /s	0.1 /s	15.0 /s	13.7 /s	1.1 /s	0.0 /s	0.2 /s	75.0 kb/s	73.5 kb/s	1.1 kb/s	20.6 b/s	368.3 b/s
rtfil	15.6 /s	13.9 /s	1.5 /s	0.1 /s	0.0 /s	27.3 /s	25.1 /s	2.0 /s	0.2 /s	0.0 /s	164.0 kb/s	160.5 kb/s	3.4 kb/s	48.6 b/s	54.1 b/s
rtrml	39.1 /s	36.4 /s	2.6 /s	0.2 /s	0.0 /s	52.8 /s	48.9 /s	3.6 /s	0.2 /s	0.0 /s	207.6 kb/s	200.0 kb/s	7.5 kb/s	89.4 b/s	51.7 b/s
rtctl	21.4 /s	18.5 /s	2.4 /s	0.2 /s	0.3 /s	39.5 /s	28.8 /s	8.7 /s	0.2 /s	1.7 /s	181.0 kb/s	165.2 kb/s	9.4 kb/s	105.8 b/s	6.2 kb/s
rtto1	5.5 /s	4.2 /s	1.3 /s	0.0 /s	0.0 /s	74.8 /s	51.9 /s	22.8 /s	0.0 /s	0.0 /s	622.5 kb/s	363.2 kb/s	259.3 kb/s	23.0 b/s	60.9 b/s

Interfaccia grafica di NfSen (3/3)

- Processing

Statistics timeslot Jun 09 2009 - 23:40

Channel:	Flows:					Packets:					Traffic:				
	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:
rtna1	45.4 /s	42.0 /s	3.0 /s	0.2 /s	0.1 /s	80.4 /s	75.8 /s	4.2 /s	0.3 /s	0.1 /s	507.2 kb/s	501.0 kb/s	5.7 kb/s	139.7 b/s	391.0 b/s
rtpi1	40.0 /s	31.4 /s	8.1 /s	0.4 /s	0.1 /s	81.0 /s	71.0 /s	9.4 /s	0.4 /s	0.2 /s	487.9 kb/s	474.6 kb/s	11.6 kb/s	288.1 b/s	1.4 kb/s
rtfi	14.6 /s	13.3 /s	1.1 /s	0.1 /s	0.1 /s	25.4 /s	23.8 /s	1.3 /s	0.2 /s	0.1 /s	169.2 kb/s	166.6 kb/s	2.4 kb/s	45.6 b/s	174.7 b/s
rtbo1	50.3 /s	49.2 /s	1.0 /s	0.1 /s	0.0 /s	482.5 /s	481.1 /s	1.3 /s	0.1 /s	0.0 /s	3.8 Mb/s	3.8 Mb/s	4.0 kb/s	43.7 b/s	167.1 b/s
rtmi2	136.6 /s	117.6 /s	17.3 /s	1.3 /s	0.4 /s	223.1 /s	196.6 /s	23.2 /s	1.4 /s	1.8 /s	1.1 Mb/s	1.1 Mb/s	43.7 kb/s	910.9 b/s	5.7 kb/s
rcfi	68.3 /s	34.2 /s	28.5 /s	5.4 /s	0.2 /s	1.2 k/s	1.1 k/s	105.7 /s	59.5 /s	2.4 /s	6.3 Mb/s	6.2 Mb/s	131.4 kb/s	41.8 kb/s	4.2 kb/s

All None Display: Sum Rate

Netflow Processing

Source: rtna1 rtpi1 rtfi rtbo1 rtmi2 rcfi — All Sources

Filter: and <none>

Options: List Flows / Stat TopN

Top: 10

Stat: Any IP Address order by flows

Limit: Packets > 0

Output: / IPv6 long

Clear Form process

```
** nfdump -M /mnt/nfsen/profiles-data/live/rtna1:rtpi1:rtfi:rtbo1:rtmi2:rcfi  -T  -r 2009/06/09/nfcapd.200906092340 -n 10 -s ip/flows
nfdump filter:
any
Top 10     IP Addr ordered by flows:
Date first seen          Duration Proto          IP Addr     Flows  Packets    Bytes      pps      bps   bpp
2009-06-09 23:37:44.169   357.448 any     193.205.145.128      3042     3043   227986        8     5102    74
2009-06-09 23:38:52.406   326.449 any     193.206.154.170      1838     5037   684572       15    16776   135
2009-06-09 23:39:04.801   299.498 any       192.167.90.2       1689     1741   281792        5     7527   161
2009-06-09 23:38:00.980   359.793 any     131.154.128.114      1537    11248   12.5 M       31   291026  1163
2009-06-09 23:38:47.588   329.763 any       192.167.90.8       1497    16990   9.6 M        51   245088   594
2009-06-09 23:38:55.761   323.030 any      193.205.58.21       1437    46878   31.2 M      145   809834   697
2009-06-09 23:37:43.890   357.560 any      193.205.144.3       1384     1384   96122        3     2150    69
2009-06-09 23:38:49.174    89.638 any      195.64.178.23       1327     2740   131785      30    11761    48
2009-06-09 23:38:00.981   358.814 any     131.154.129.92       1303     8844   9.5 M        24   222022  1125
2009-06-09 23:37:43.590   368.343 any     143.225.66.247       1215     1514   1.2 M         4    27290   829

Summary: total flows: 106526, total bytes: 443.7 M, total packets: 640506, avg bps: 6.0 M, avg pps: 1080, avg bpp: 726
Time window: 2009-06-09 23:34:26 - 2009-06-09 23:44:18
Total flows processed: 106526, skipped: 0, Bytes read: 5539460
Sys: 0.184s flows/second: 578911.0   Wall: 0.295s flows/second: 360054.2
```

Nfsen

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

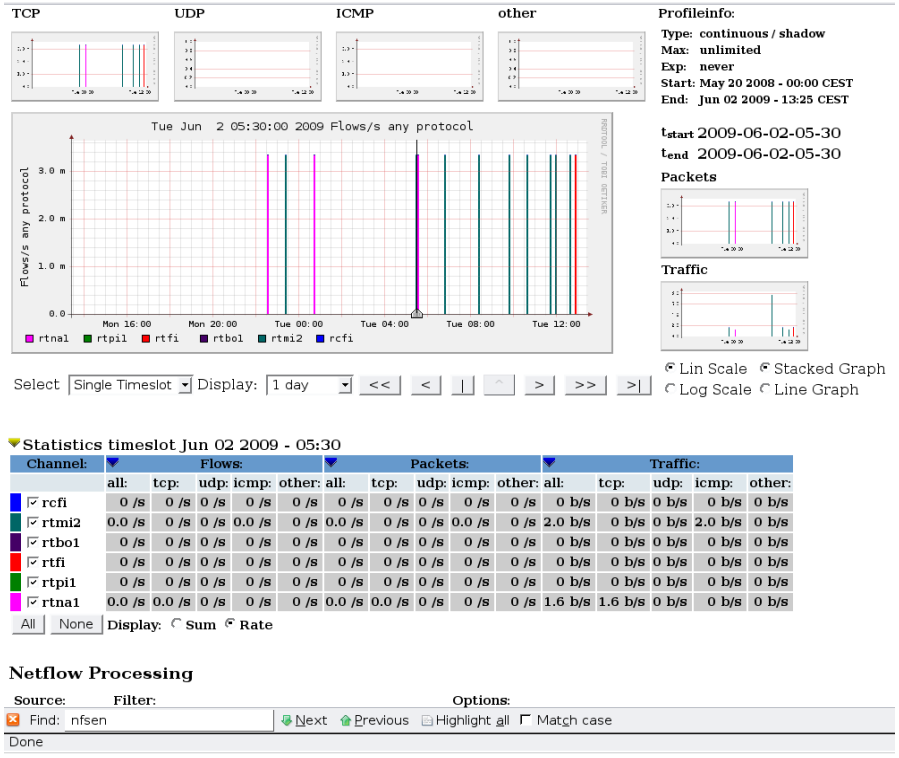
Profili

- 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' riferirsi a qualsiasi filtro (limite 255 caratteri)

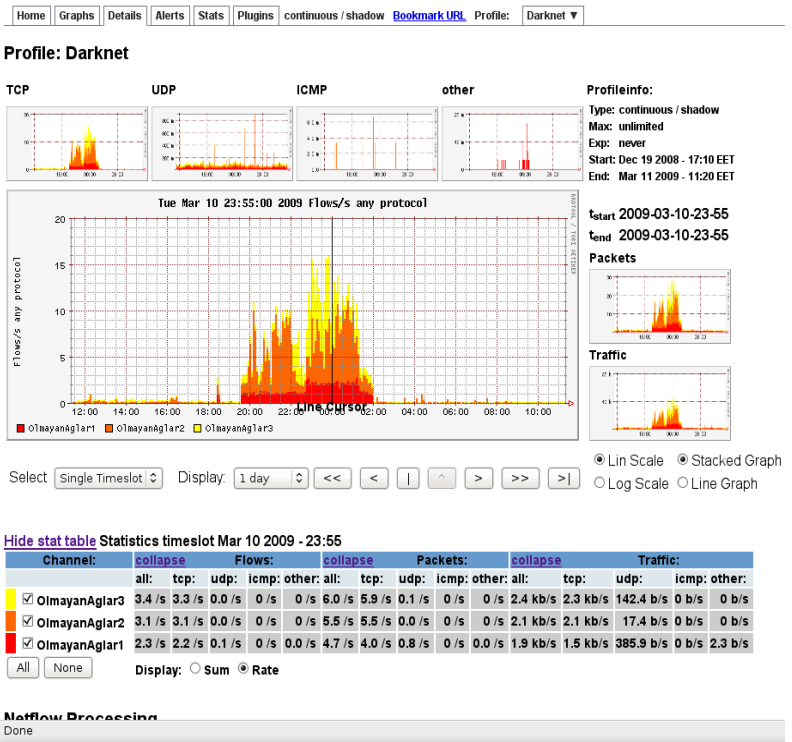
Profili: esempi

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

dove ip1, ip2, ..., ipN sono ip di botnet

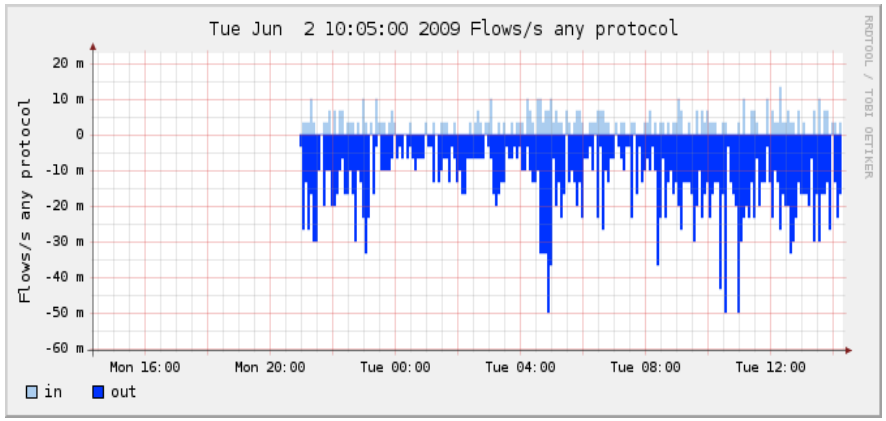


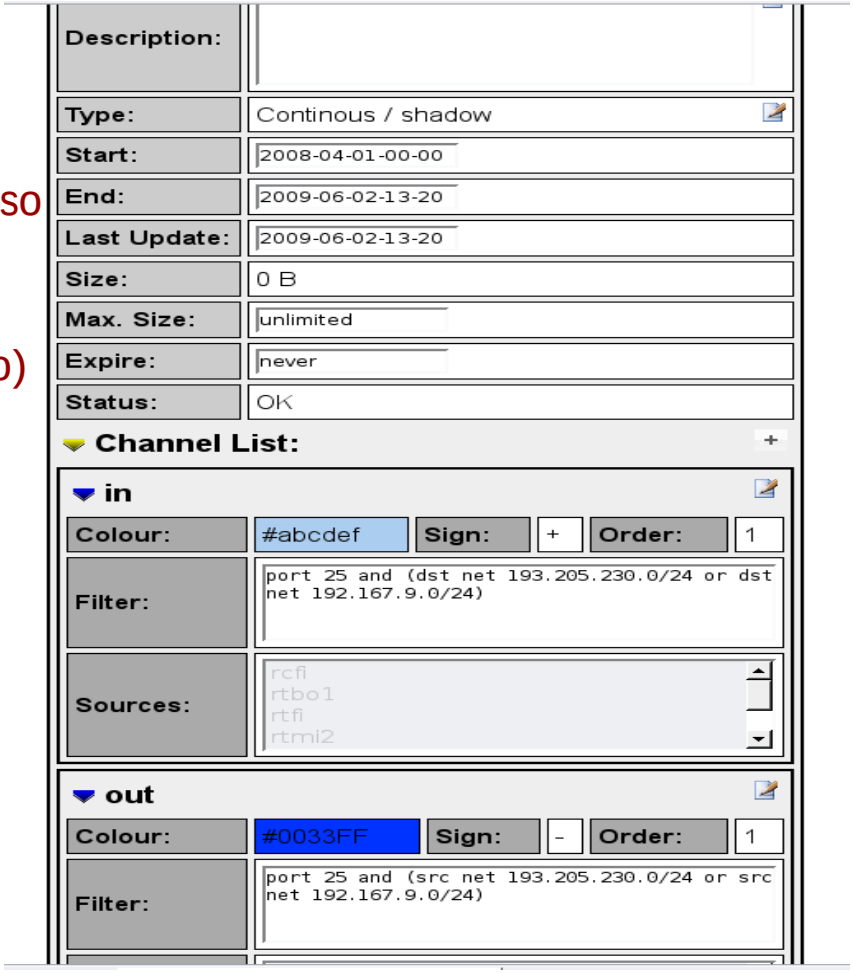
- Monitoraggio di una DarkNet
- il filtro è:
  - 'net rete.dark.net/24'



Profili: creazione

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





Profili: tipi

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

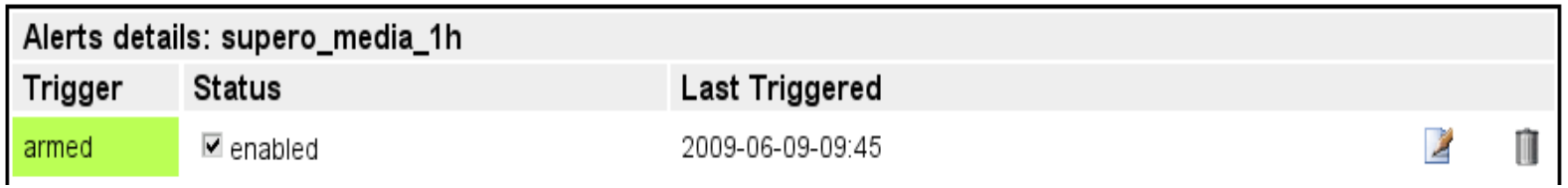
Alert

- Ogni 5 minuti viene controllata la condizione di alerting impostata
- Estrema flessibilità nelle impostazione di soglie e filtri
- Sistema automatico di “reazione” una volta che la condizione di alert è verificata
- Sistema automatico di “notifica” una volta che la condizione di alert è verificata

Alert: esempi di condizioni

- Manda un alert quando il numero di flussi nei 5 minuti è > 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 mi calcola chi sia l'ip coinvolto col 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 più complicato)

Alert: configurazione - stato



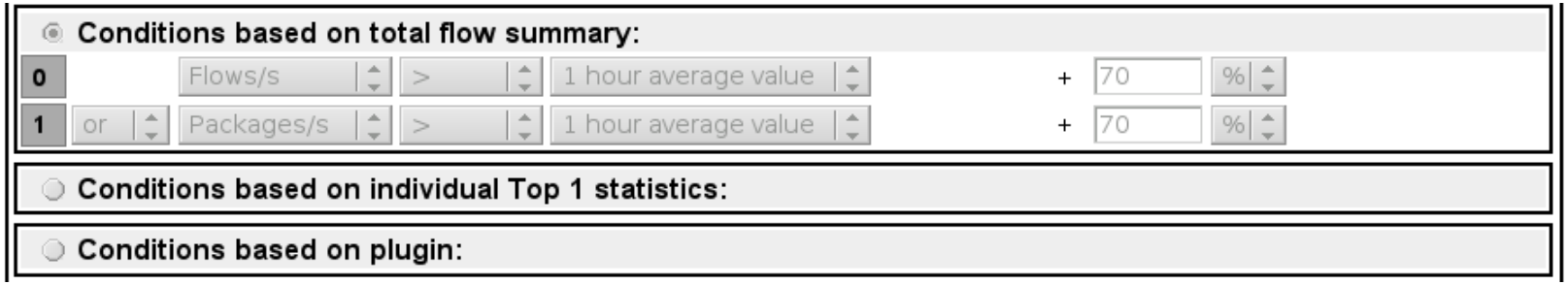
Alerts details: supero_media_1h

Trigger	Status	Last Triggered
armed	enabled	2009-06-09-09:45

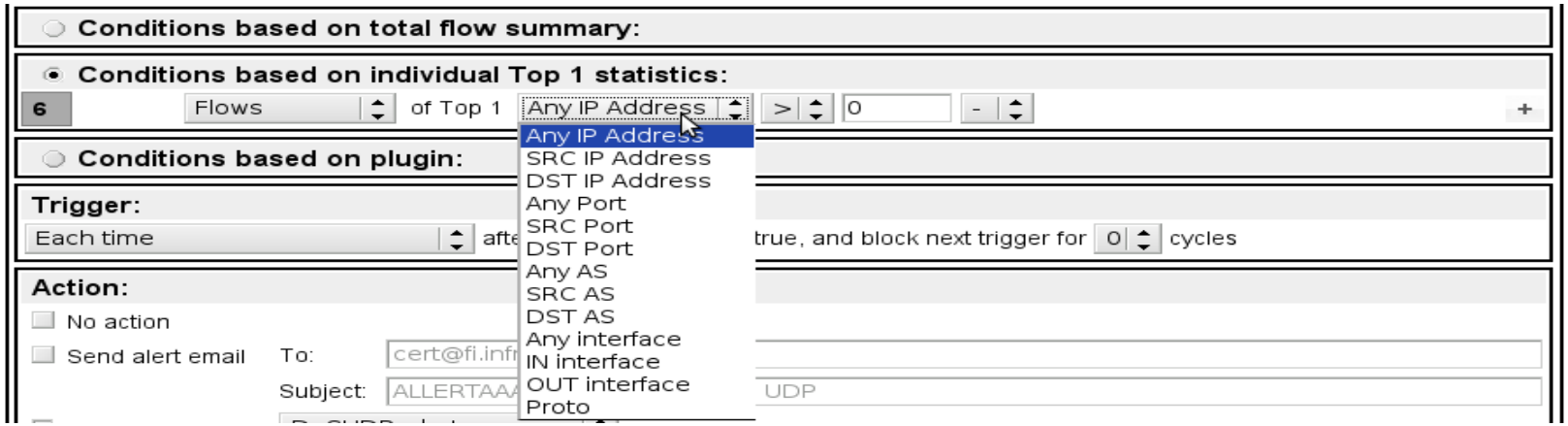
- Alerts details: nome alert
- Trigger: inactive, armed, armed 1/x, fired, fired - -, blocked 1/x
- Stato: abilitato o disabilitato
- Last Triggered: l'ultima volta che è scattato
- Modifica / Elimina

Alert: configurazione - condizioni

- Condizione sui totali:

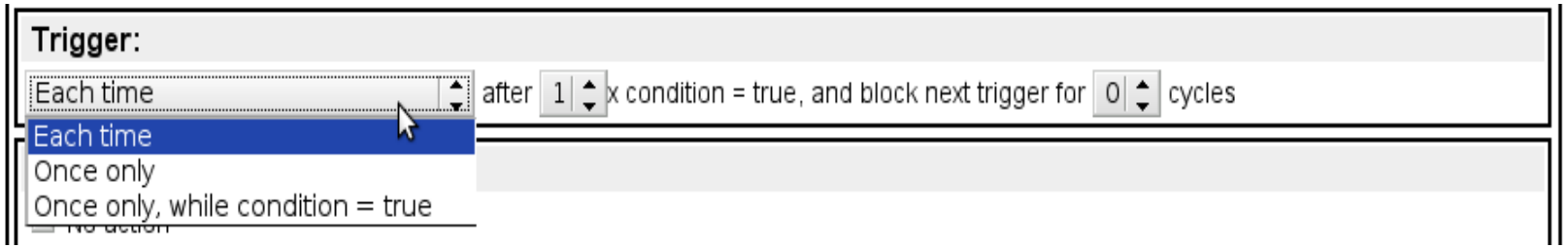


- Condizione su aggregati:



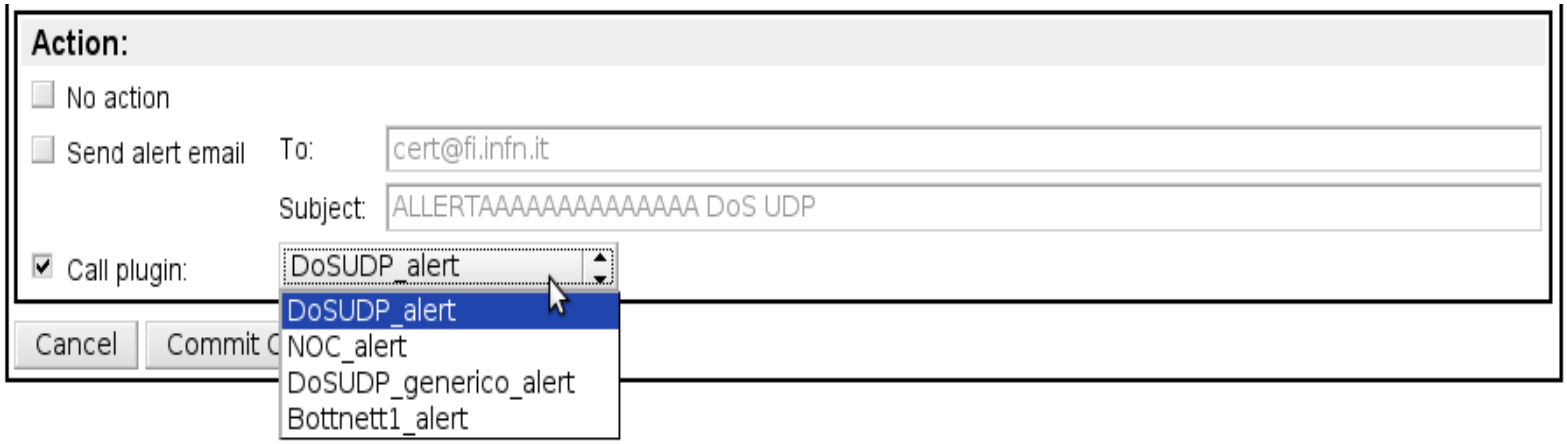
- Condizioni su plugin

Alert: configurazione - trigger



- Specifica quando fa scattare l'allarme:
  - Ogni volta o una sola volta o n volte
  - Dopo ogni quante volte
  - Dopo quanti cicli viene bloccato

Alert: configurazione - action



- Specifica cosa fare quando scatta l'allarme:
  - Niente
  - Manda una mail con indirizzo e oggetto
  - Esegue un programma esterno (plugin)

Alert: infos

www.garr.it

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**

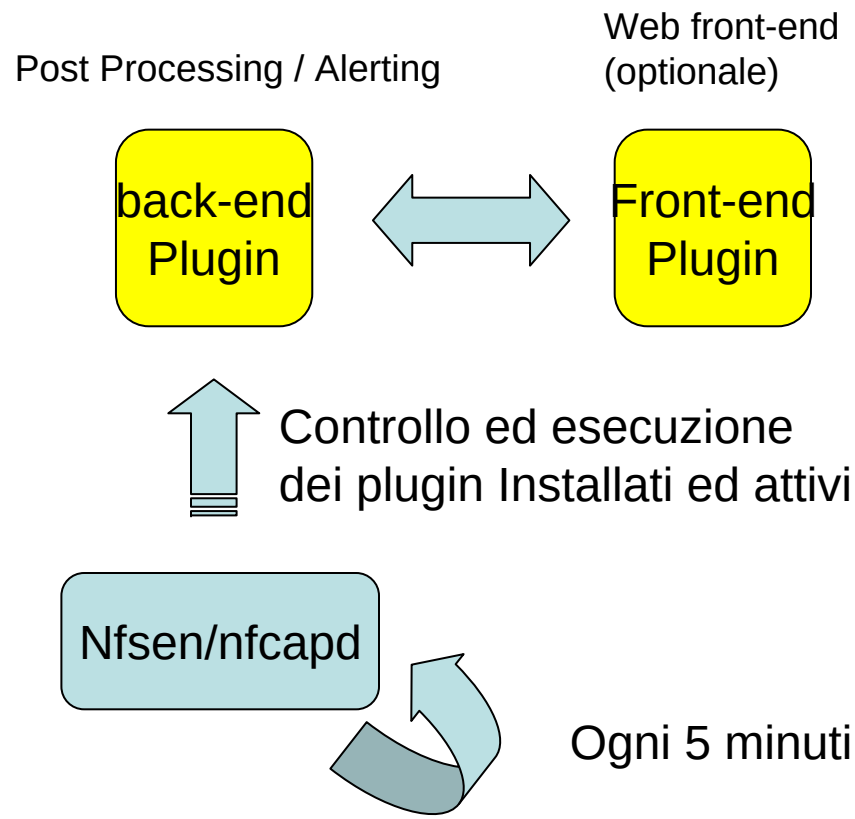
Plugin

- 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

Plugin: funzionamento

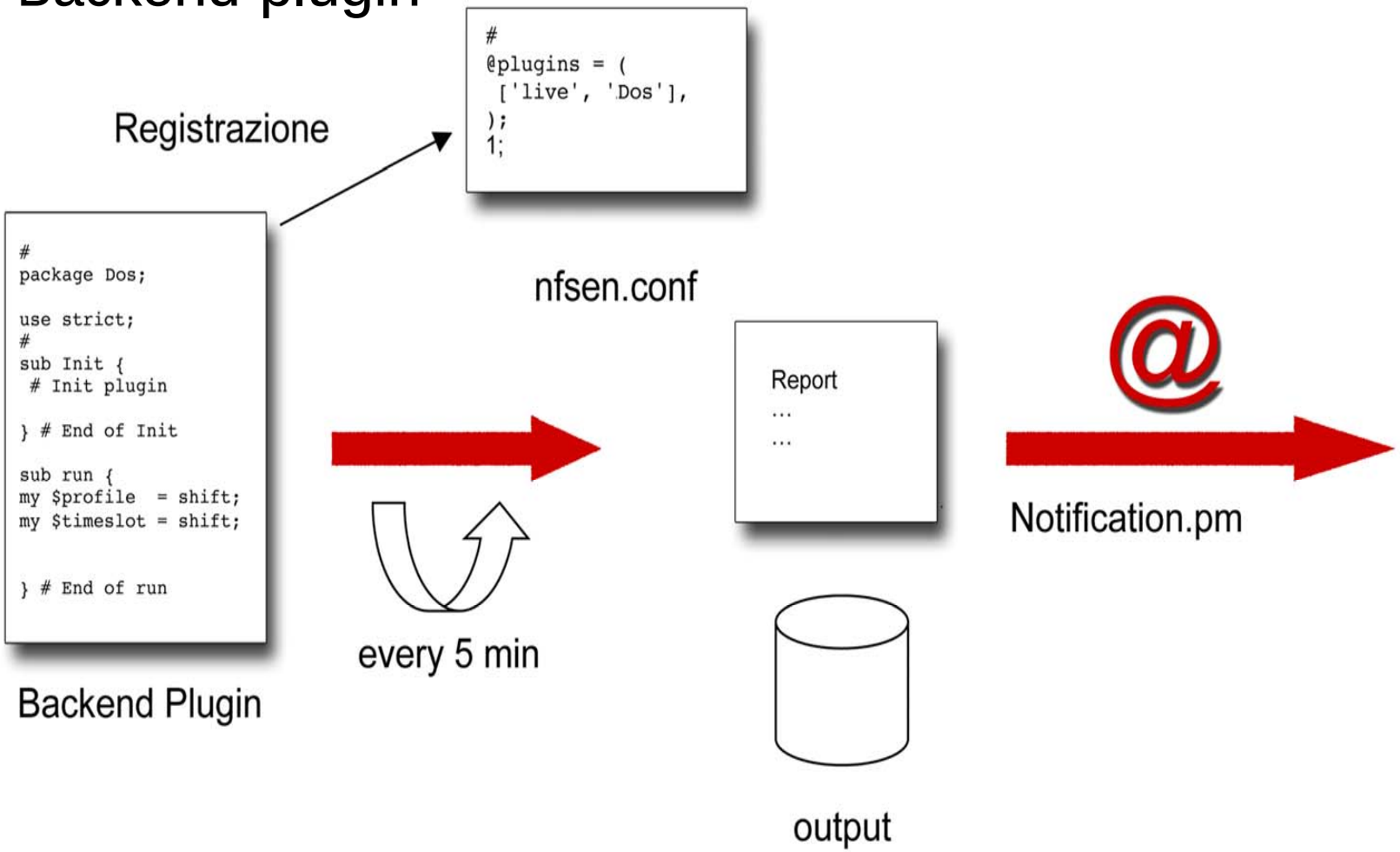
- Esistono due tipi di plugin:
  - Back-end plugin:
    - Sono moduli perl che vengono “agganciati” ad Nfsen
    - Fanno cose altrimenti non possibili con la suite
    - Sono richiamati automaticamente ogni 5 minuti
  - Front-end plugin:
    - Sono moduli PHP “agganciati” ad NfSen
    - Fanno parte delle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

Plugin: architettura (1/3)



Plugin: architettura (2/3)

- Backend plugin



Plugin: architettura (3/3)

- Frontend plugin

Registrazione

```
#
@plugins = (
 ['live', 'Dos'],
);
1;
```

nfsen.conf

```
#
package Dos;

use strict;
#
sub Init {
 # Init plugin

} # End of Init

sub run {
my $profile  = shift;
my $timeslot = shift;


} # End of run
```

every 5 min

output

```
/* Port Tracker
 * Plugin
 */
function
PortTracker_Run($plugi) {
 $portinfo =
GetTopN($plugin_id,
$_SESSION["${plugin_id}

…
} // End of
PortTracker_Run
```

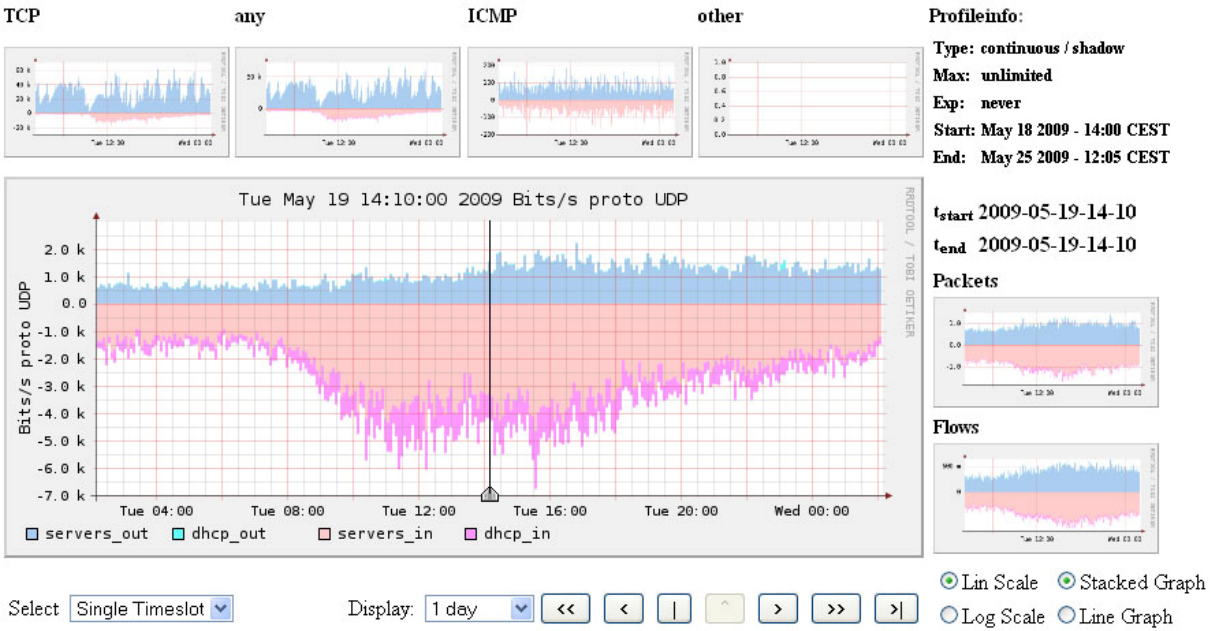
Utilizzo avanzato di Nfsen/Nfdump

- Monitoraggio della rete
  - Analisi su base:
    - 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”
  - Tracciamento Botnet

Monitoraggio della rete

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

Traffico suddiviso per subnet

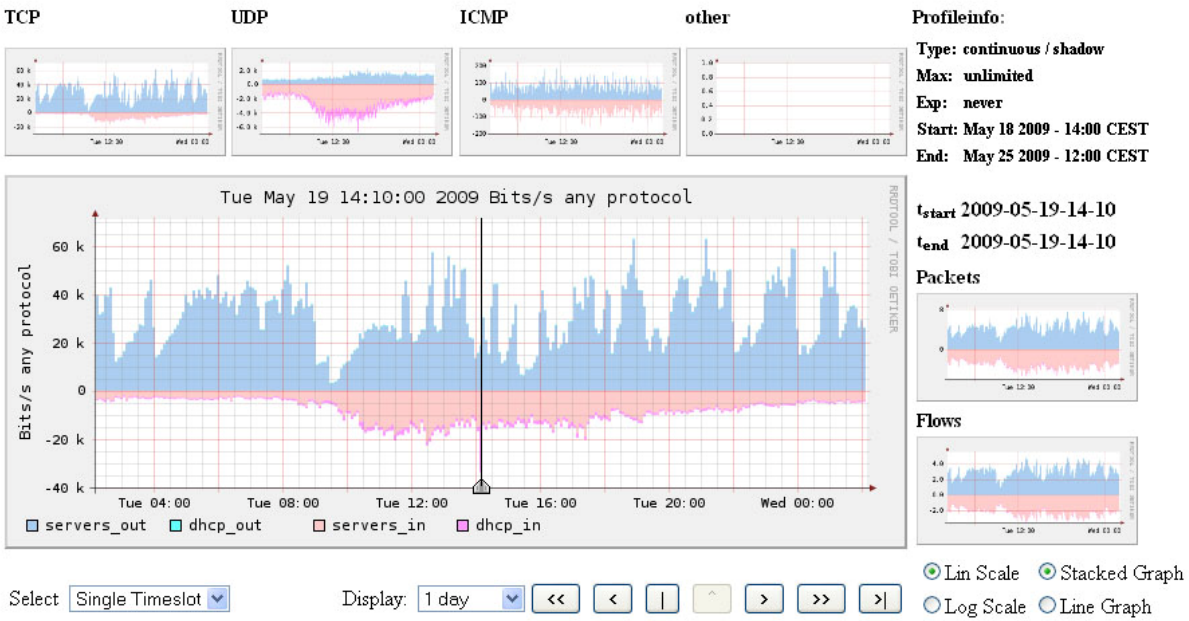


Statistics timeslot May 19 2009 - 14:10

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

Traffico suddiviso per subnet

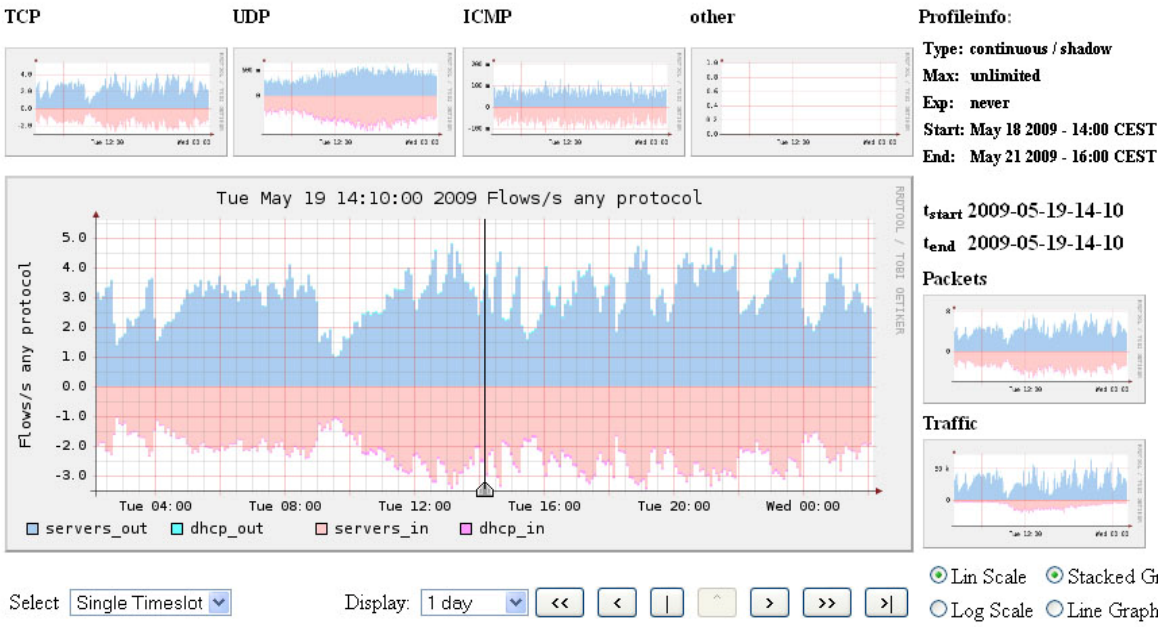


Statistics timeslot May 19 2009 - 14:10

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

Traffico suddiviso per subnet

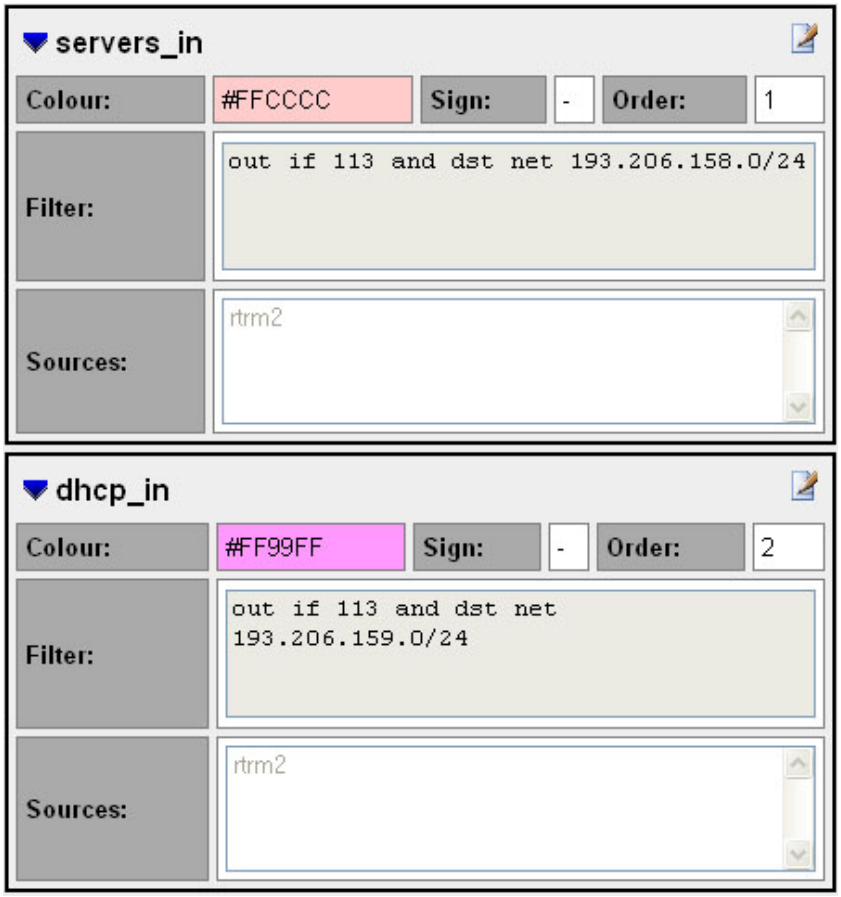


Statistics timeslot May 19 2009 - 14:10

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

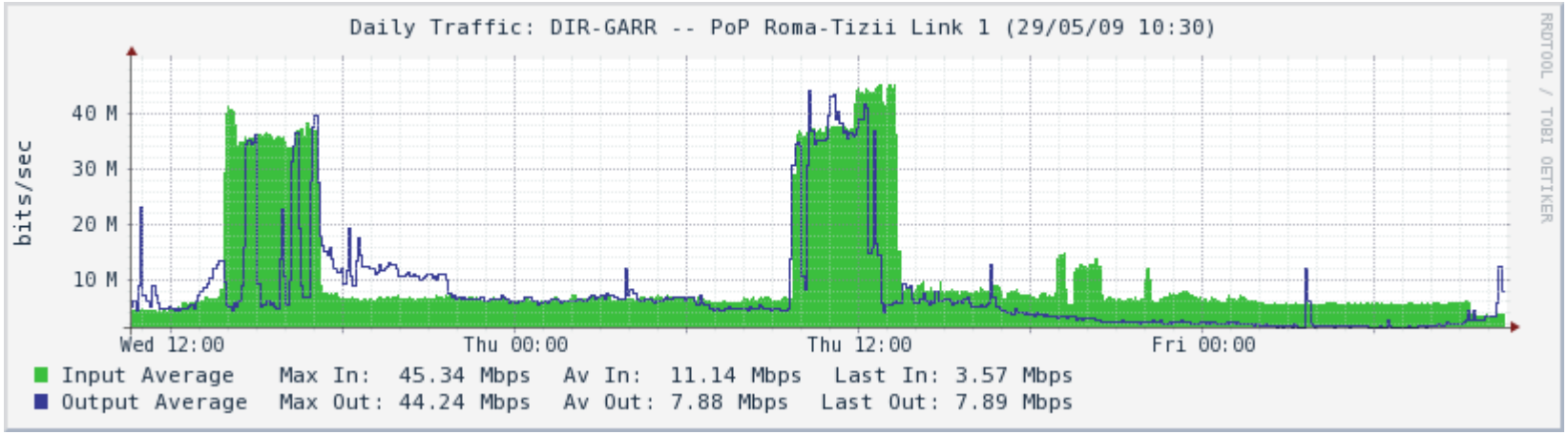
All None Display: Sum Rate

Relativi filtri.....

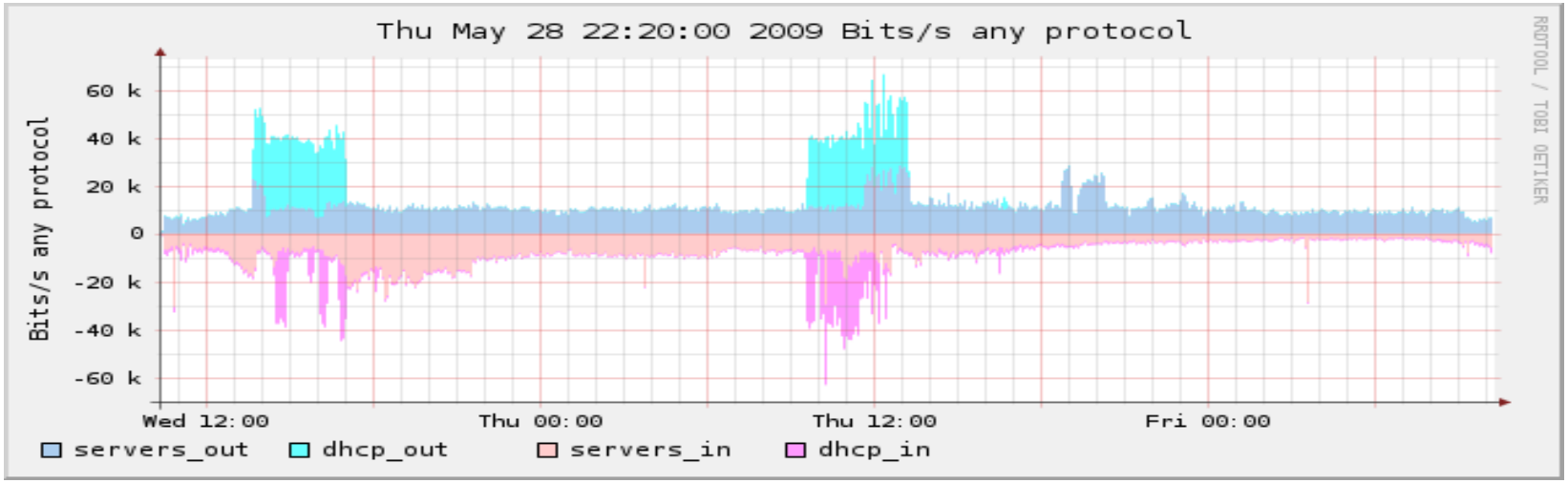


SNMP vs NetFlow

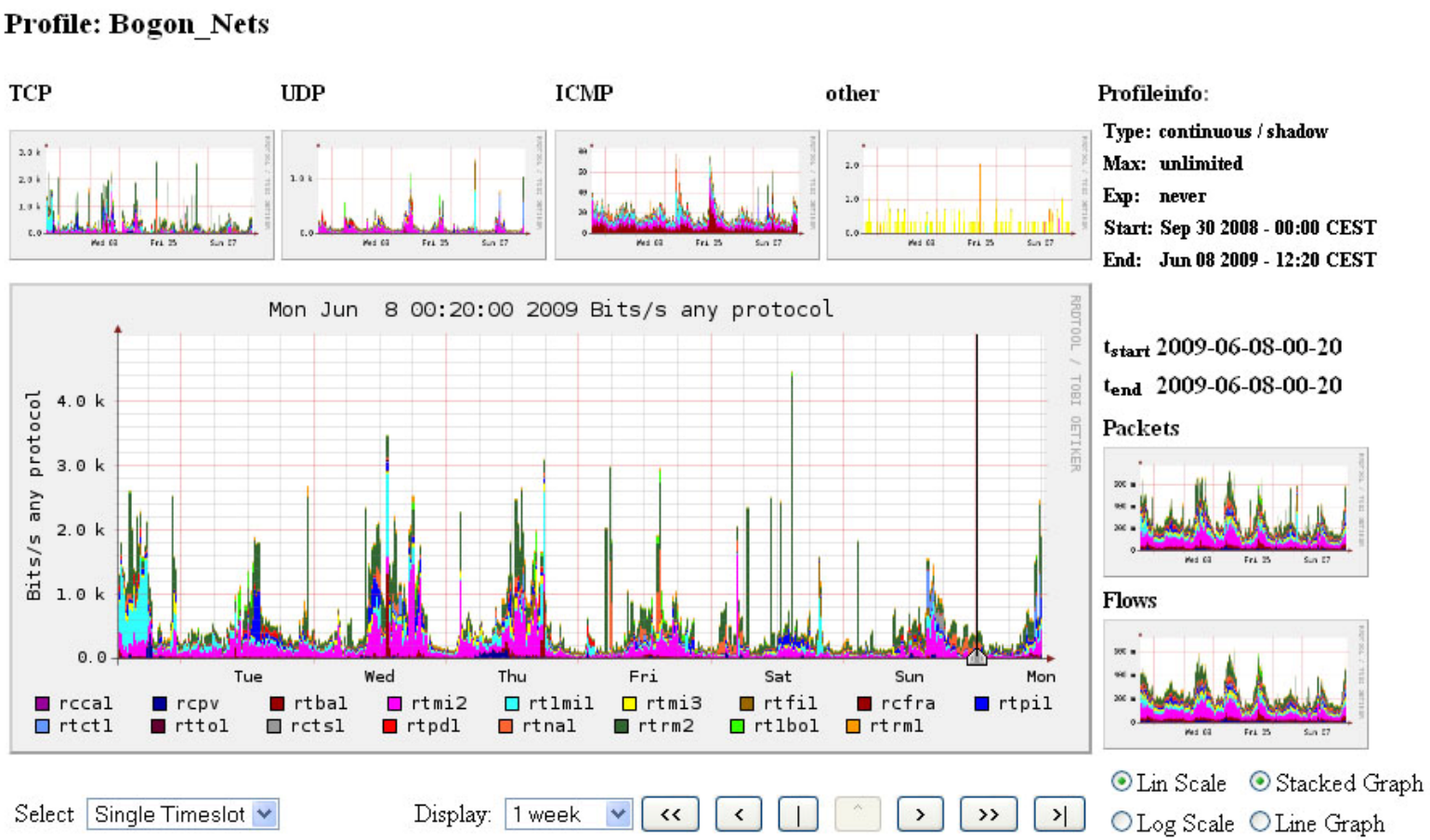
GINS (SNMP)



Profilo Nfsen (NetFlow)

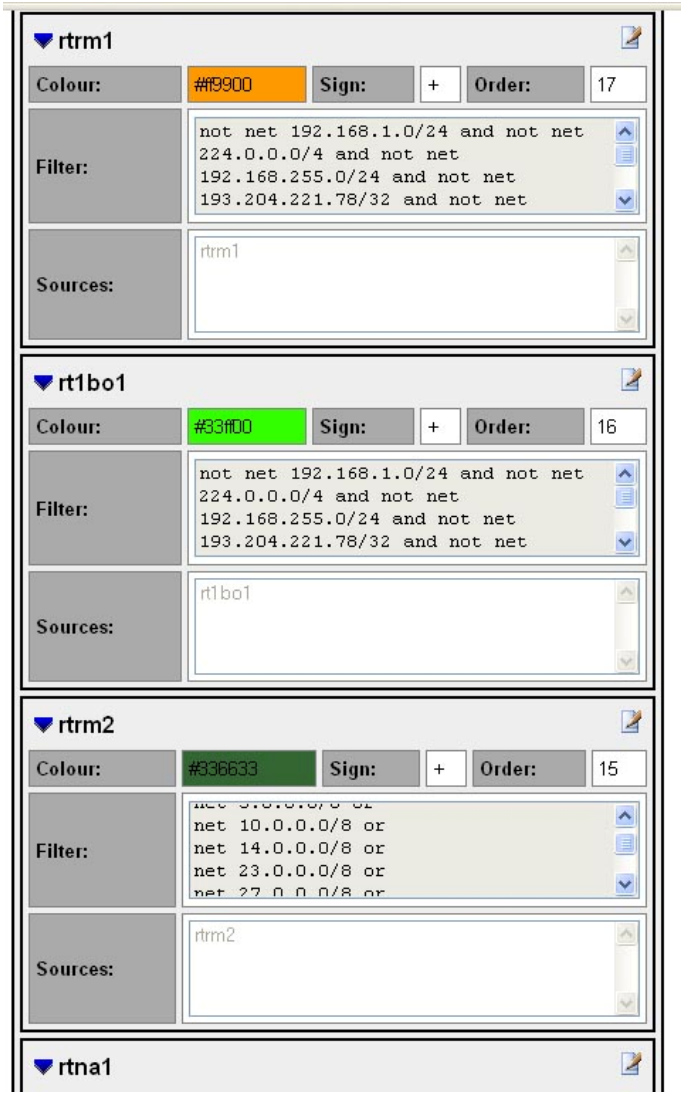


Bogon Network



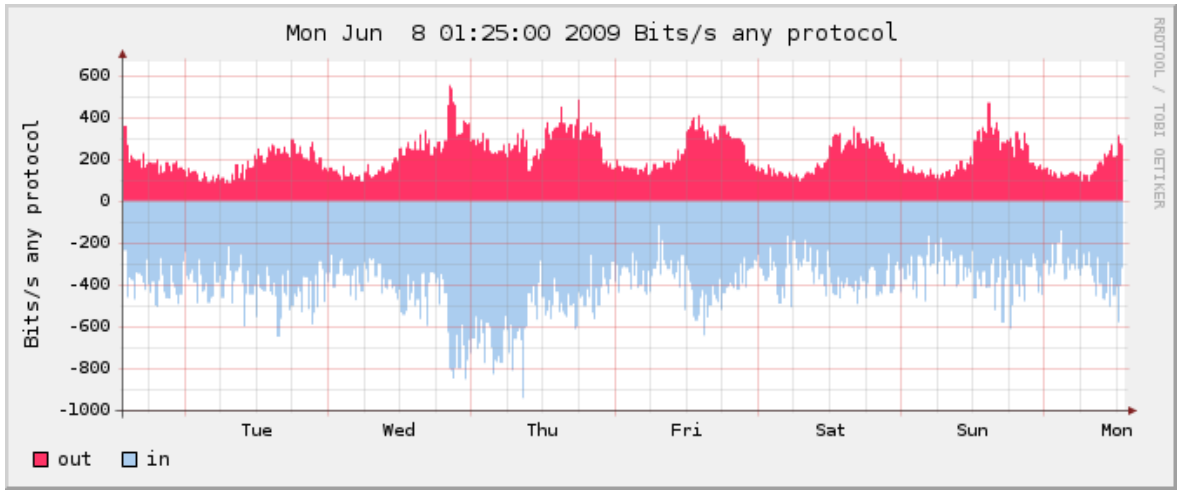
Bogon Network

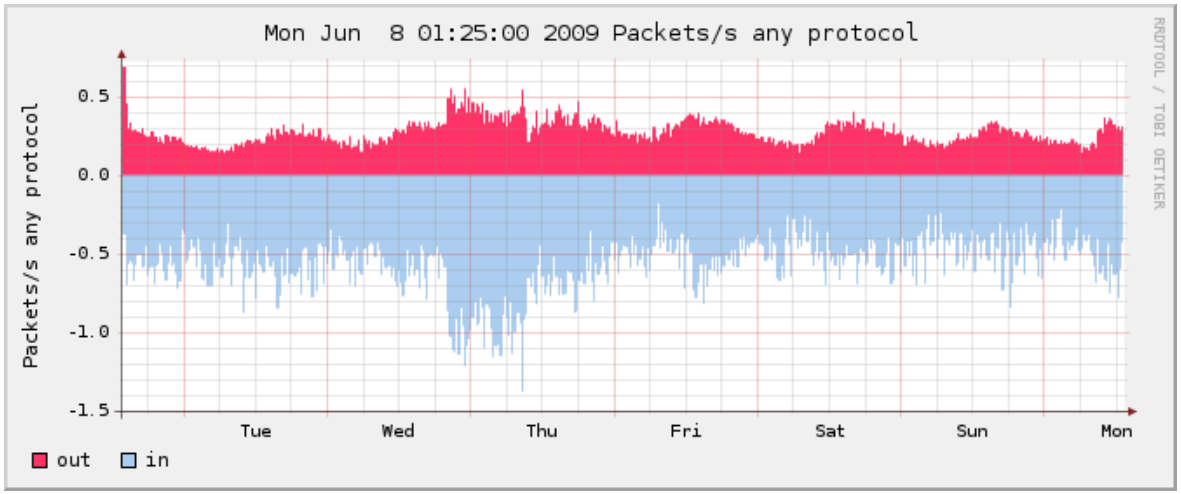
Bogon Network



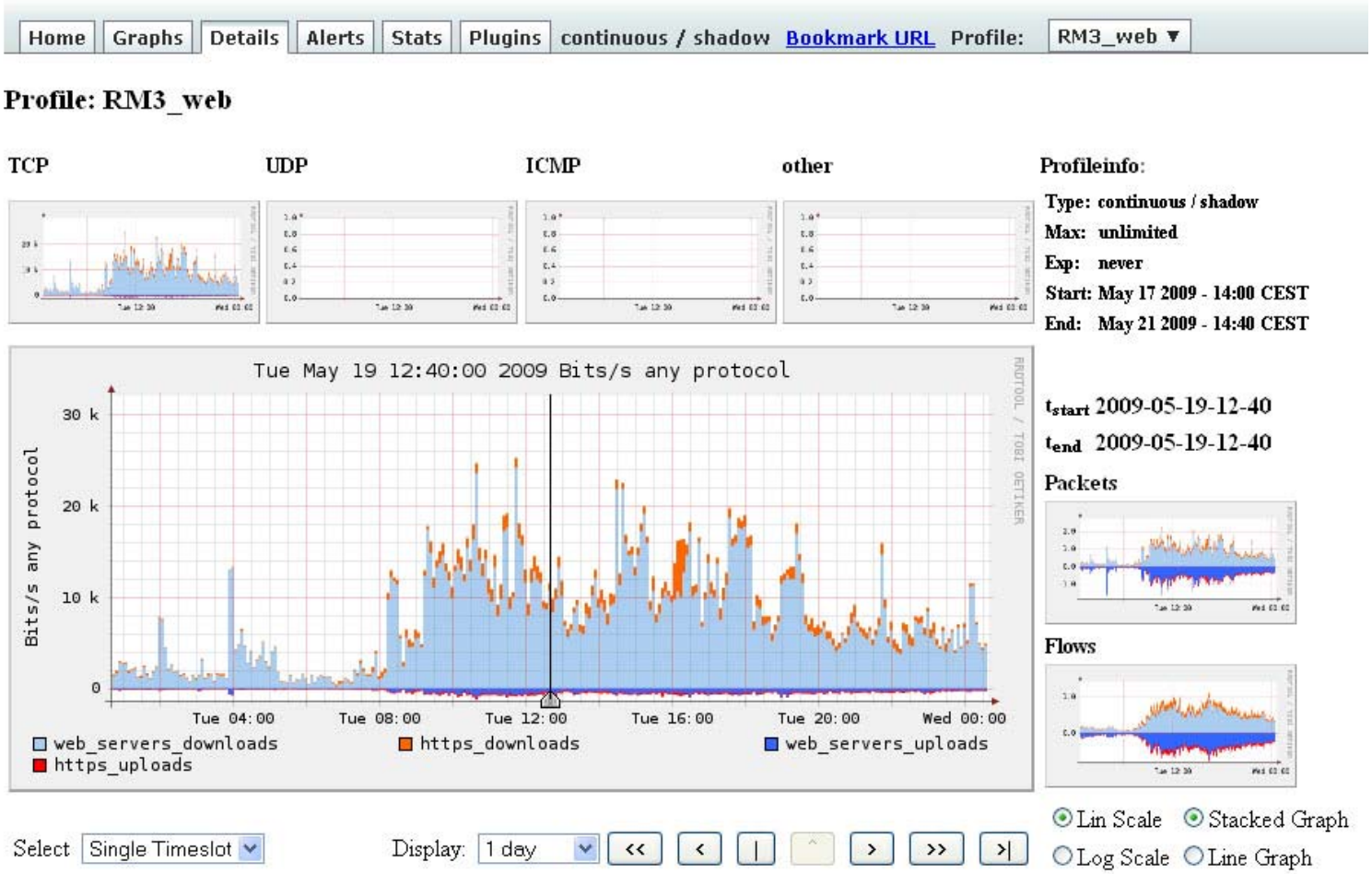
ICMP su Upstream Provider

ICMP su Upstream Provider

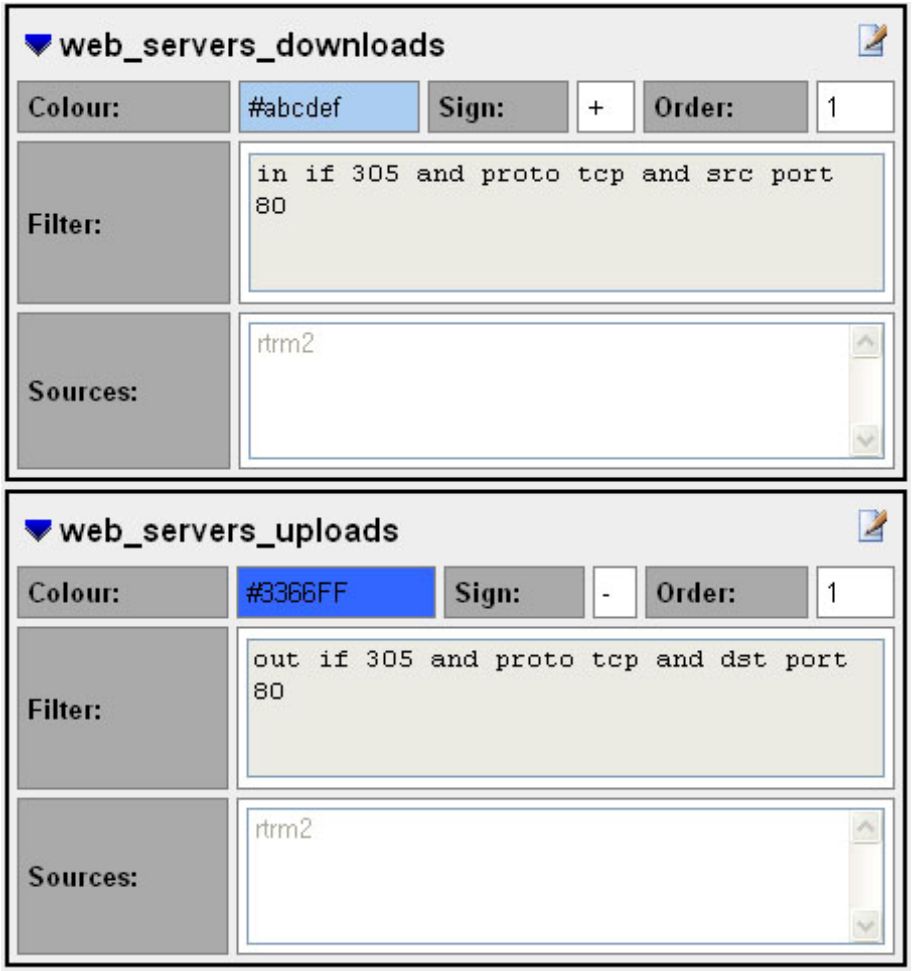




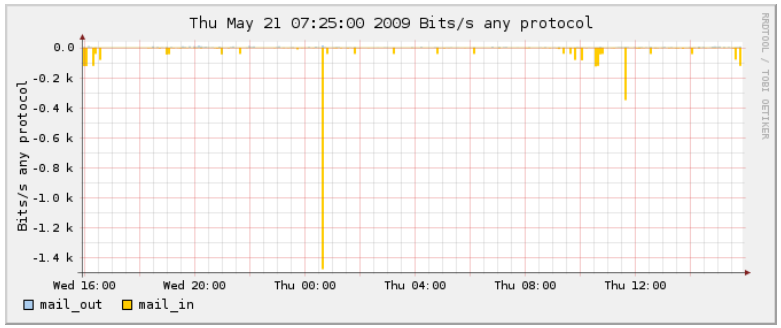
Web Server

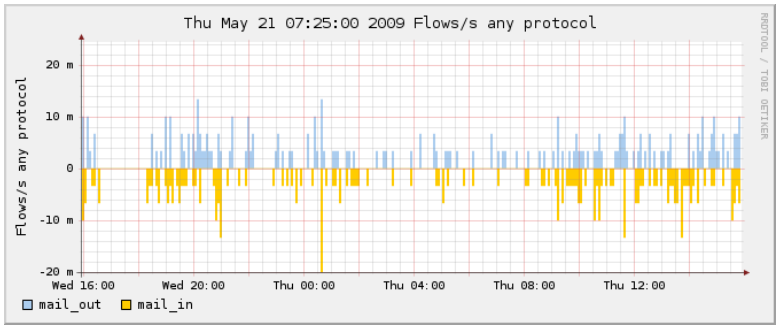

Home
Graphs
Details
Alerts
Stats
Plugins
continuous / shadow
Bookmark URL
Profile:
RM3_web
Profile: RM3_web
TCP
UDP
ICMP
other
Profileinfo:
Type: continuous / shadow
Max: unlimited
Exp: never
Start: May 17 2009 - 14:00 CEST
End: May 21 2009 - 14:40 CEST
Tue May 19 12:40:00 2009 Bits/s any protocol
Bits/s any protocol
30 k
20 k
10 k
0
Tue 04:00
Tue 08:00
Tue 12:00
Tue 16:00
Tue 20:00
Wed 00:00
web_servers_downloads
https_downloads
web_servers_uploads
https_uploads
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

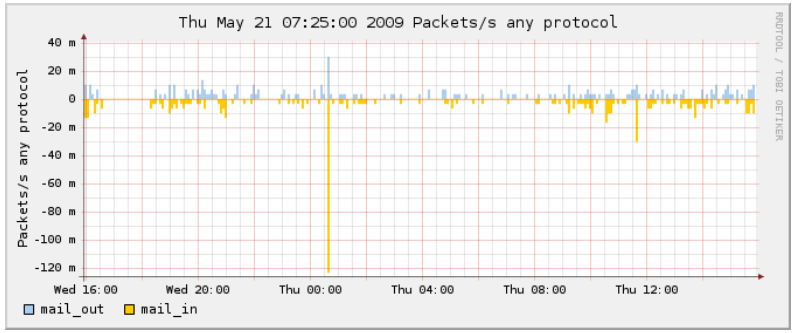
Relativi filtri.....



Mail Server

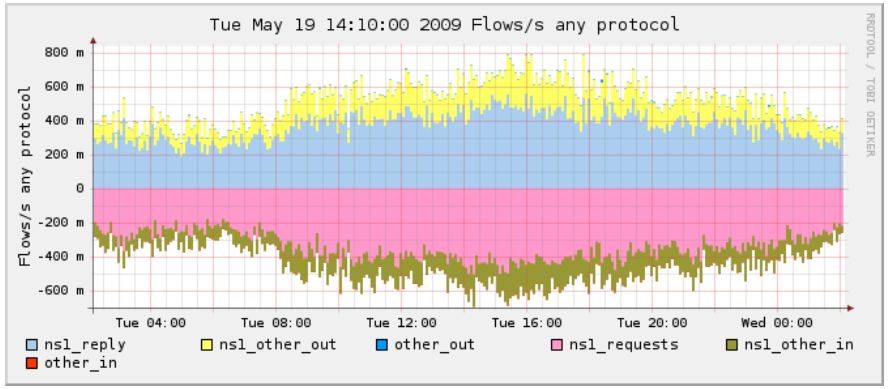

Thu May 21 07:25:00 2009 Bits/s any protocol
Bits/s any protocol
0.0
-0.2 k
-0.4 k
-0.6 k
-0.8 k
-1.0 k
-1.2 k
-1.4 k
Wed 16:00
Wed 20:00
Thu 00:00
Thu 04:00
Thu 08:00
Thu 12:00
mail_out
mail_in
RRDTOOL / TOBI OETIKER

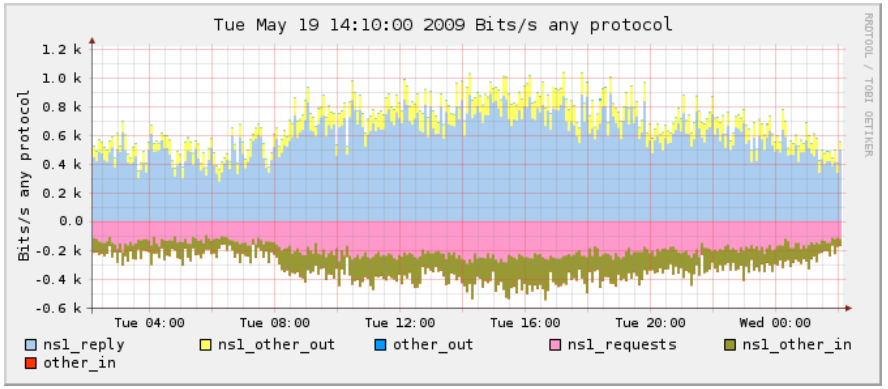

Thu May 21 07:25:00 2009 Flows/s any protocol
Flows/s any protocol
20 m
10 m
0
-10 m
-20 m
Wed 16:00
Wed 20:00
Thu 00:00
Thu 04:00
Thu 08:00
Thu 12:00
mail_out
mail_in
RRDTOOL / TOBI OETIKER

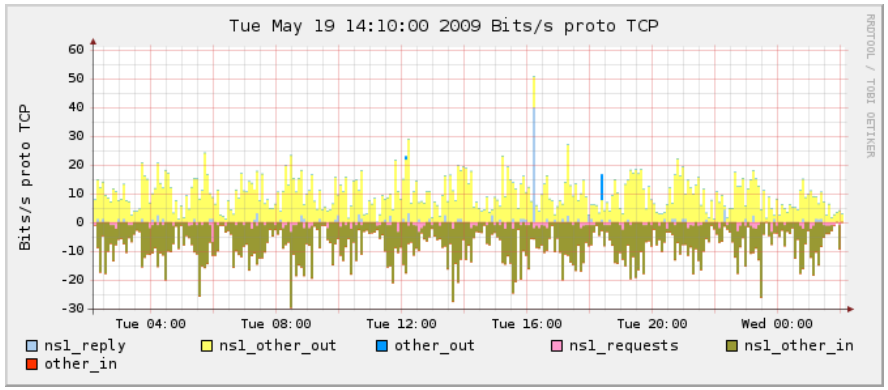

Thu May 21 07:25:00 2009 Packets/s any protocol
Packets/s any protocol
40 m
20 m
0
-20 m
-40 m
-60 m
-80 m
-100 m
-120 m
Wed 16:00
Wed 20:00
Thu 00:00
Thu 04:00
Thu 08:00
Thu 12:00
mail_out
mail_in
RRDTOOL / TOBI OETIKER

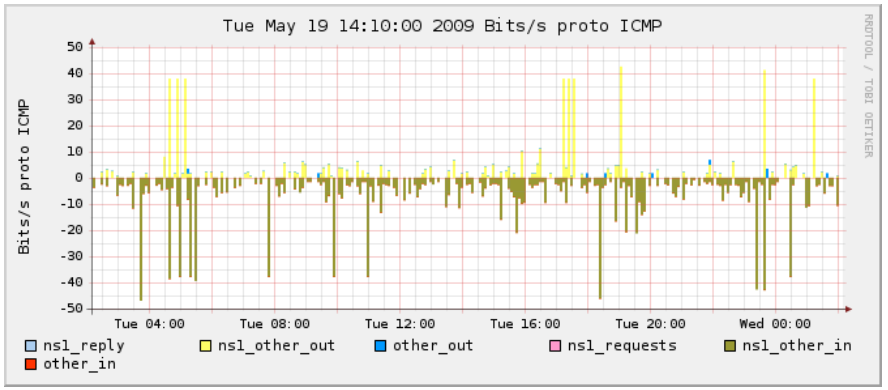
Name Server

www.garr.it

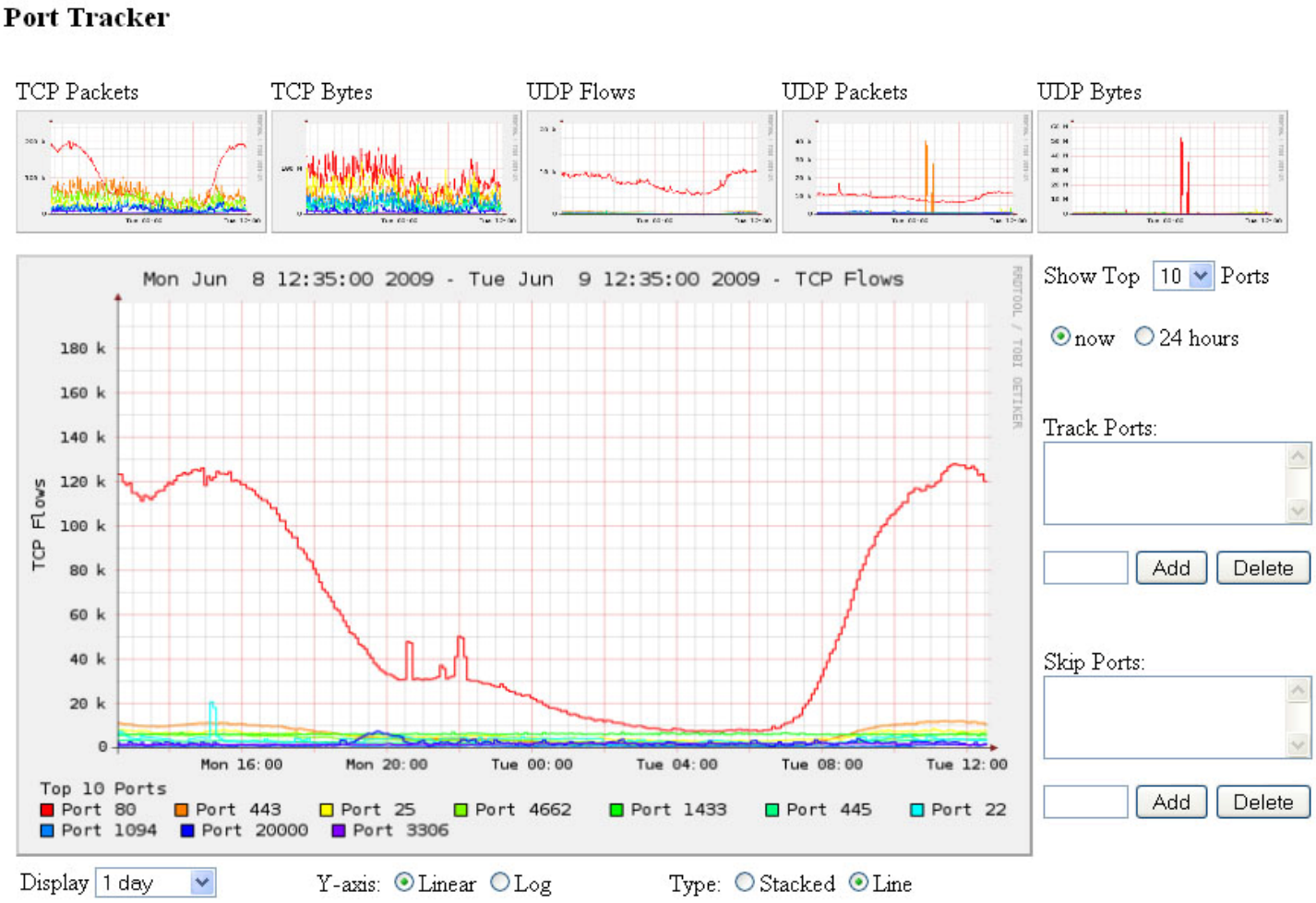








Porte (plugin PortTracker)



Porte (plugin PortTracker)

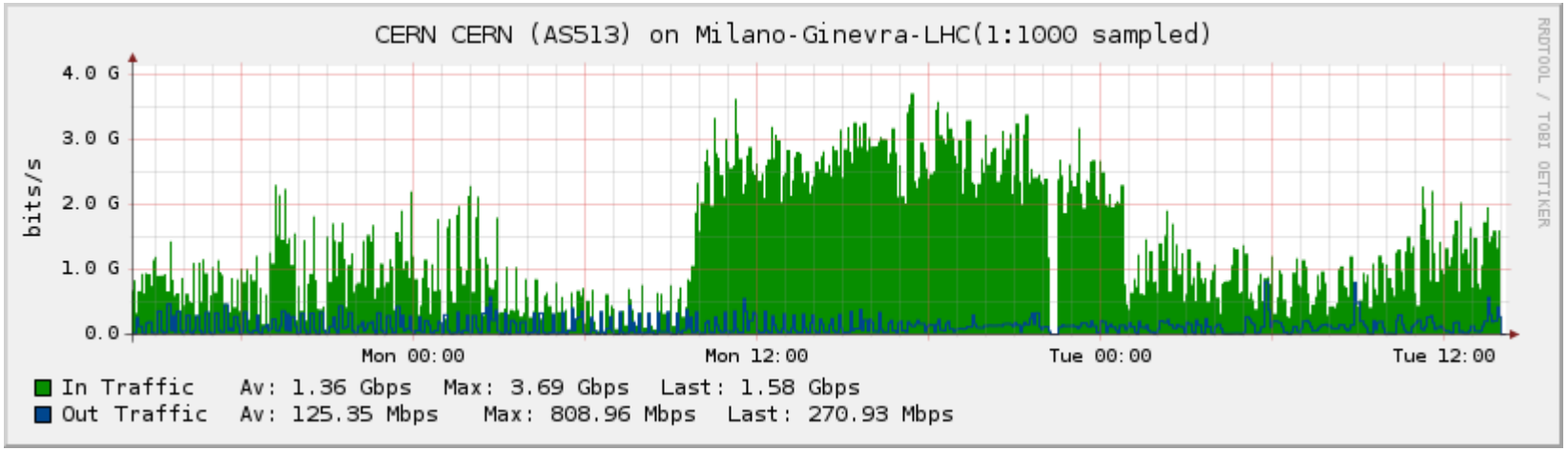
Top 10 Statistics

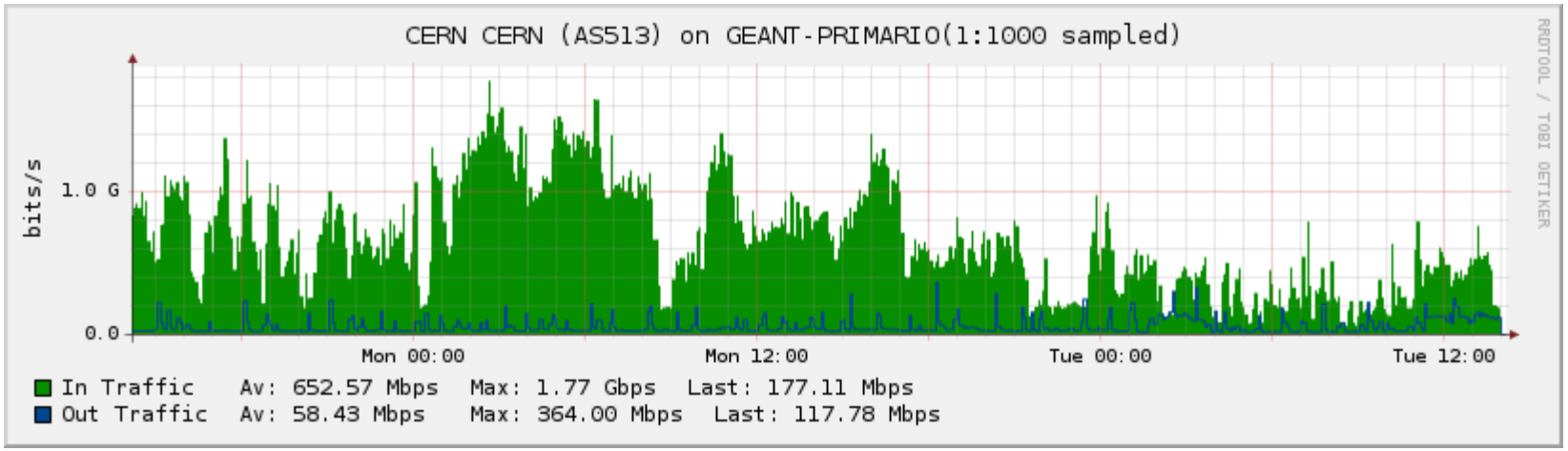
	TCP						UDP					
	Flows		Packets		Bytes		Flows		Packets		Bytes	
Rank	Port	Count	Port	Count	Port	Count	Port	Count	Port	Count	Port	Count
1	80	120126	80	184201	20000	51858147	53	10010	53	11766	0	1235438
2	443	10751	20000	37431	20002	26949676	123	610	0	1217	53	927008
3	25	7748	20002	21256	20001	23740002	15000	514	15000	1137	15000	597559
4	4662	6425	20001	18067	80	22876497	0	355	61964	959	20937	542082
5	1433	5781	443	13352	20004	13172939	20937	288	1031	790	4500	314591
6	445	3802	20004	11455	20007	12730217	6277	240	1032	774	6972	313937
7	22	3315	25	10548	20006	11276099	4500	239	20937	717	7000	288322
8	1094	1751	20006	9375	20003	8524008	6970	225	4500	669	15836	225801
9	20000	1605	20007	9194	20011	8516207	4672	215	15836	651	40827	209949
10	3306	1391	4662	8680	4662	7850731	137	206	123	617	6970	205372

Numeri AS (plugin AsTracker)

Numeri AS (plugin AsTracker)

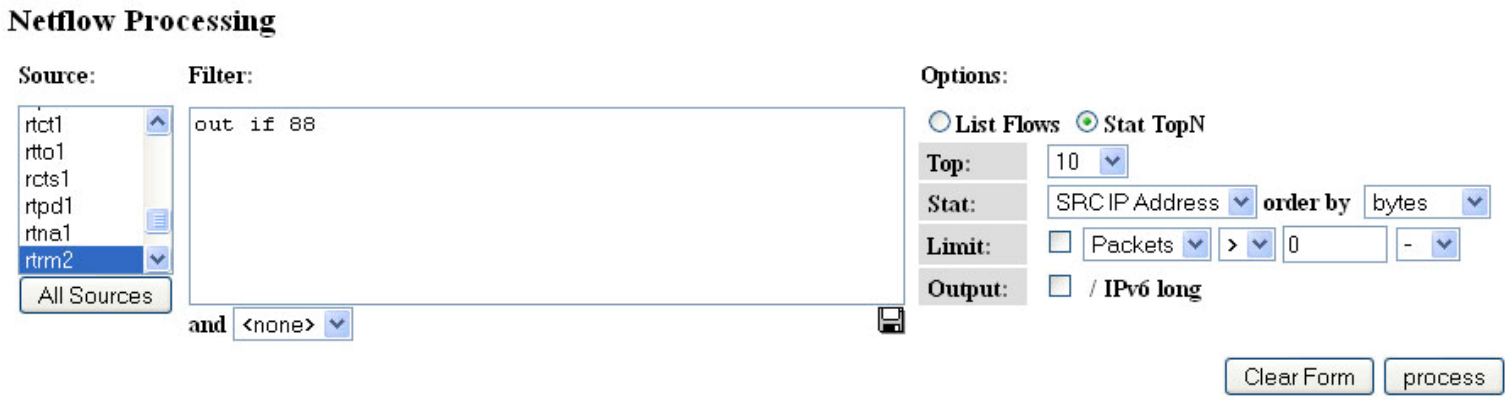
www.garr.it





TopTalkers (1/6)

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



** 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

TopTalkers (2/6)

IP che hanno avuto il maggior troughout:

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

TopTalkers (3/6)

IP più 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 (4/6) con aggregazione personalizzata

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


```
** 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   ......   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 (5/6) 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'intefaccia 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 (6/6) con aggregazione personalizzata

Le 15 coppie di subnet /24 che fanno più 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

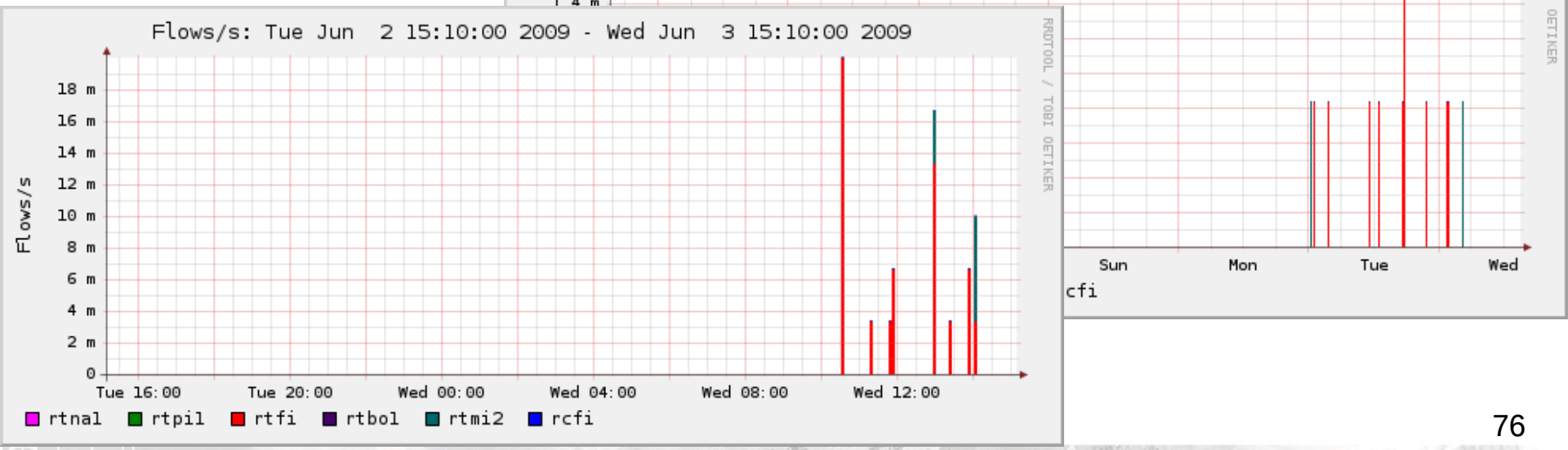
Analisi degli incidenti di sicurezza

- Segnalazione esterna di macchina che fa traffico anomalo
- Controllo del traffico sul router a quel timeslot

- 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
    - Tracciamento di IP spoofati in caso di DdoS
    - 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

Identificazione di traffico “malevolo” (1/3)

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” (2/3)

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	->	1.1.8.9:53	87874	3.6 M	212	72960	43	16
2009-06-03 13:59:17.225	3922.643	UDP	1.4.80.47:1194	->	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” (3/3)

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	**1.80.208.239**:35823 ->	2.200.199.203:22	....S.	1	60	1
2009-06-04 16:59:39.123	**1.80.208.23**9:44577 ->	2.200.195.226:22	....S.	1	60	1
2009-06-04 17:01:54.084	**1.80.208.239**:53769 ->	2.200.237.133:22	....S.	1	60	1
2009-06-04 17:01:31.024	**1.80.208.239**:60829 ->	2.200.232.39:22	....S.	1	60	1
2009-06-04 16:59:53.279	**1.80.208.239**:53731 ->	2.200.199.108:22	....S.	1	60	1
2009-06-04 17:00:28.348	**1.80.208.239**:34654 ->	2.200.211.104:22	....S.	1	60	1
2009-06-04 16:59:31.080	**1.80.208.239**:54641 ->	2.200.194.28:22	....S.	1	60	1
2009-06-04 16:59:18.316	**1.80.208.239**:34426 ->	2.200.189.162:22	....S.	1	60	1
2009-06-04 17:00:34.093	**1.80.208.239**:51640 ->	2.200.212.76:22	....S.	1	60	1
2009-06-04 16:58:54.459	**1.80.208.239**:58954 ->	2.200.182.14:22	....S.	1	60	1
2009-06-04 16:59:35.459	**1.80.208.239**:58471 ->	2.200.195.5:22	....S.	1	60	1
2009-06-04 17:01:01.080	**1.80.208.239**:52688 ->	2.200.222.197:22	....S.	1	60	1

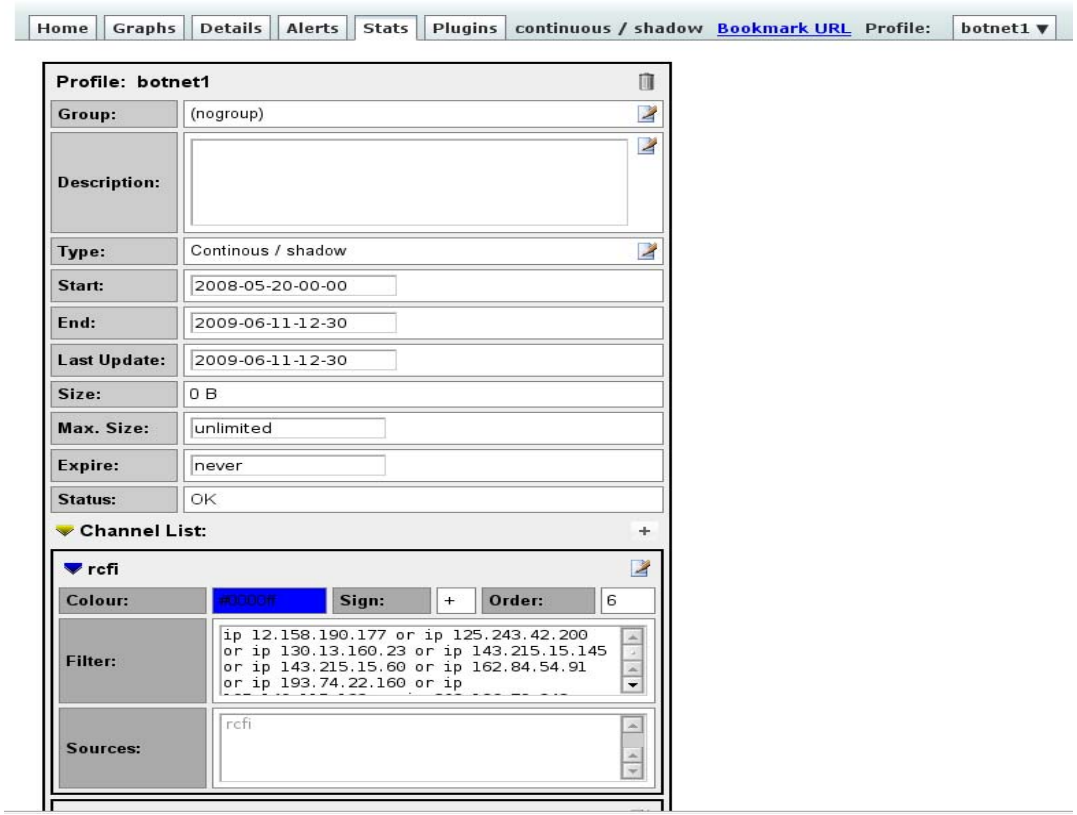
Summary: total flows: 105, total bytes: 6300, total packets: 105, avg bps: 214, avg pps: 0, avg bpp: 60
Time window: 2009-06-04 16:58:17 - 2009-06-04 17:05:07
Total flows processed: 565374, Records skipped: 0, Bytes read: 29400036
Sys: 0.068s flows/second: 8313834.5 Wall: 0.064s flows/second: 8823628.6

Tracciamento BotNet

- Due indirizzi di monitoraggio e tracciamento BotNet:
  - Profilatura degli host BotNet Controller da una lista di server di BotNet noti
  - Sistema di plugin e alert per notifica ogni volta che una macchina GARR si connette a uno di questi server BotNet

Tracciamento BotNet - profilatura

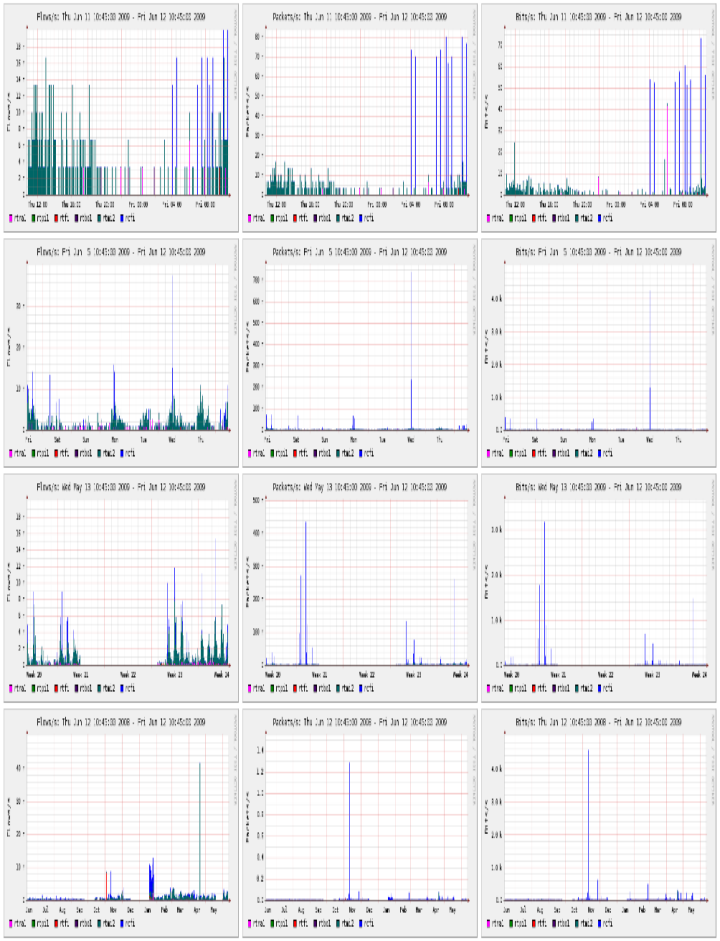
- Creazione di un profile detto “Botnet1” che comprende tutte le connessioni alla lista di server considerati Bot Controller



Tracciamento BotNet - profilo

www.garr.it

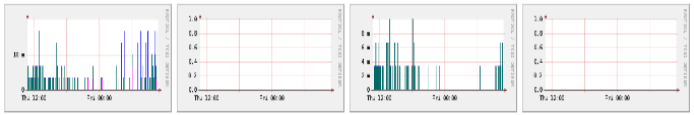
Overview Profile: botnet1, Group: (nogroup)



Home | Graphs | Details | Alerts | Stats | Plugins | continuous / shadow | Bookmark URL | Profile: botnet1

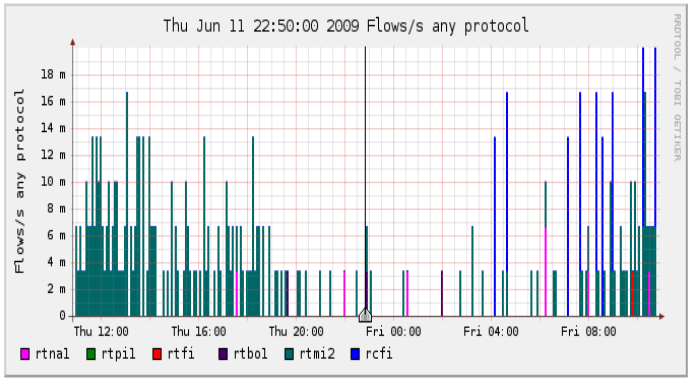
Profile: botnet1

TCP UDP ICMP other



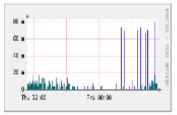
Profileinfo:

Type: continuous / shadow
Max: unlimited
Exp: never
Start: May 20 2008 - 00:00 CEST
End: Jun 12 2009 - 10:50 CEST

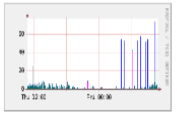


t_{start} 2009-06-11-22-50
t_{end} 2009-06-11-22-50

Packets



Traffic



Select Single Timeslot Display: 1 day << < | ^ > >> >|

Lin Scale Stacked Graph
Log Scale Line Graph

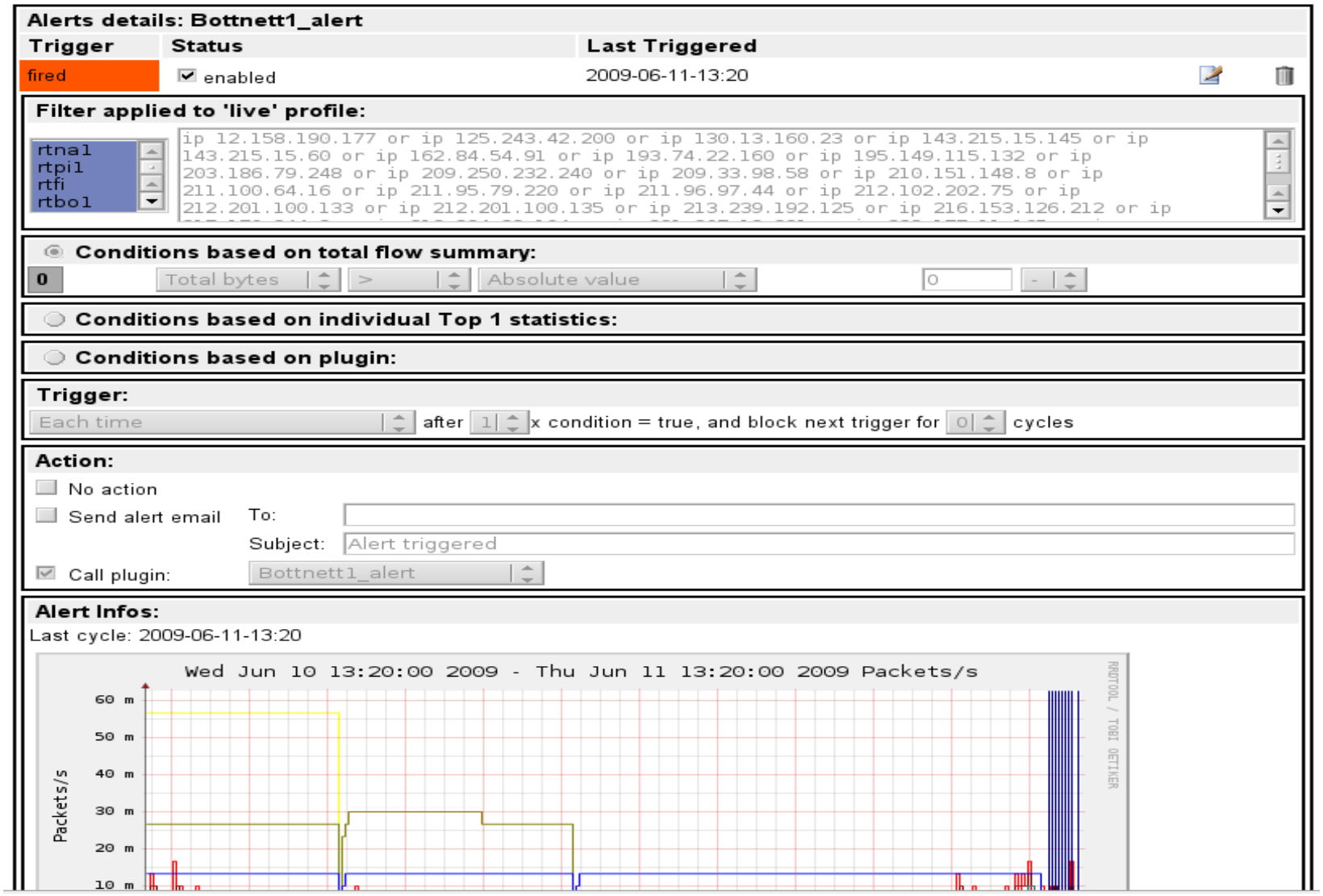
Statistics timeslot Jun 11 2009 - 22:50

Channel:	Flows:					Packets:					Traffic:				
	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 b/s	0 b/s	0 b/s	0 b/s	0 b/s
rtmi2	0.0 /s	0.0 /s	0 /s	0 /s	0 /s	0.0 /s	0.0 /s	0 /s	0 /s	0 /s	1.3 b/s	1.3 b/s	0 b/s	0 b/s	0 b/s
rtbo1	0.0 /s	0.0 /s	0 /s	0 /s	0 /s	0.0 /s	0.0 /s	0 /s	0 /s	0 /s	1.3 b/s	1.3 b/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

Tracciamento Botnet - plugin/alert

- Creazione di un alert che “suona” ogni volta che una macchina fa una connessione alla lista dei BotNet noti
- Allo scattare dell'alert viene eseguito un plugin che estrapola da tutto il traffico l'ip GARR che si connette, cerca l'indirizzo dell'APM e invia a cert@garr.it una mail di notifica con l'indicazione dei flussi di traffico presumibilmente malevoli

Tracciamento Botnet - alert



NetFlow sulla LAN

- switch (sFlow, NetFlow)
- server (nProbe)
  - monitoring NAT, firewall, server

sFlow

- Su alcuni brand di switch, in alternativa a NetFlow, è disponibile sFlow
- La tecnologia sFlow, oltre ai protocolli IP/ICMP/UDP/TCP, supporta anche:
  - Ethernet/802.3
  - 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),

Server

- **Scopo:**
  - Fare monitoring di sistemi dietro un firewall/NAT linux (no router o switch o firewall appliance).
  - In questo modo si riesce a vedere quale IP interno al NAT effettua una data connessione, utile per disaggregare il traffico NAT nelle singole macchine interne
- **Come:**
  - Utilizzando un programma che crea pacchetti NetFlow
  - Utilizzando i dati di flussi e pacchetti entranti e uscenti dalle interfacce del NAT

Server

- **Cosa:**
  - nProbe
  - flowprobe
  - NetFlow iptables module

- **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** (sviluppato da Luca Deri http://www.ntop.org/nProbe.html) per le caratteristiche di altissime performance, stabilità e affidabilità

  E, ovviamente, è opensource per gli enti accademici e di ricerca

Installazione nProbe e collector (1/2)

- Sulla macchina che fa NAT
  - Download nProbe (mail da www.ntop.org)
  - gzip -d nprobe-1.tgz
  - tar xvf nprobe-1.tar
  - ./autogen.sh && make && make install
  - Comandi d'esportazione:
    - **nprobe -G -i eth0 -n localhost:10000**
    - **nprobe -G -i eth1 -n localhost:10001**

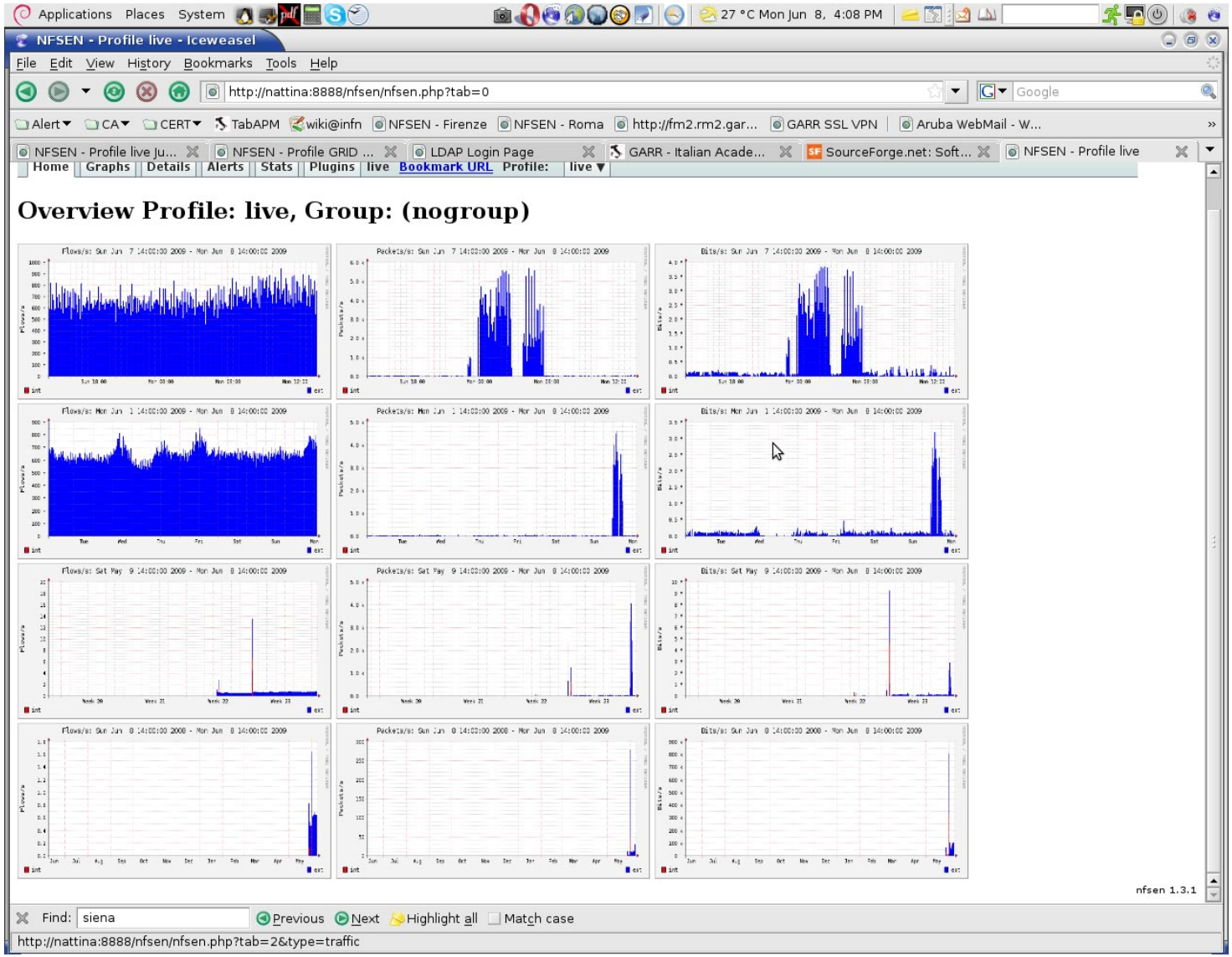
Installazione nProbe e collector (2/2)

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

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

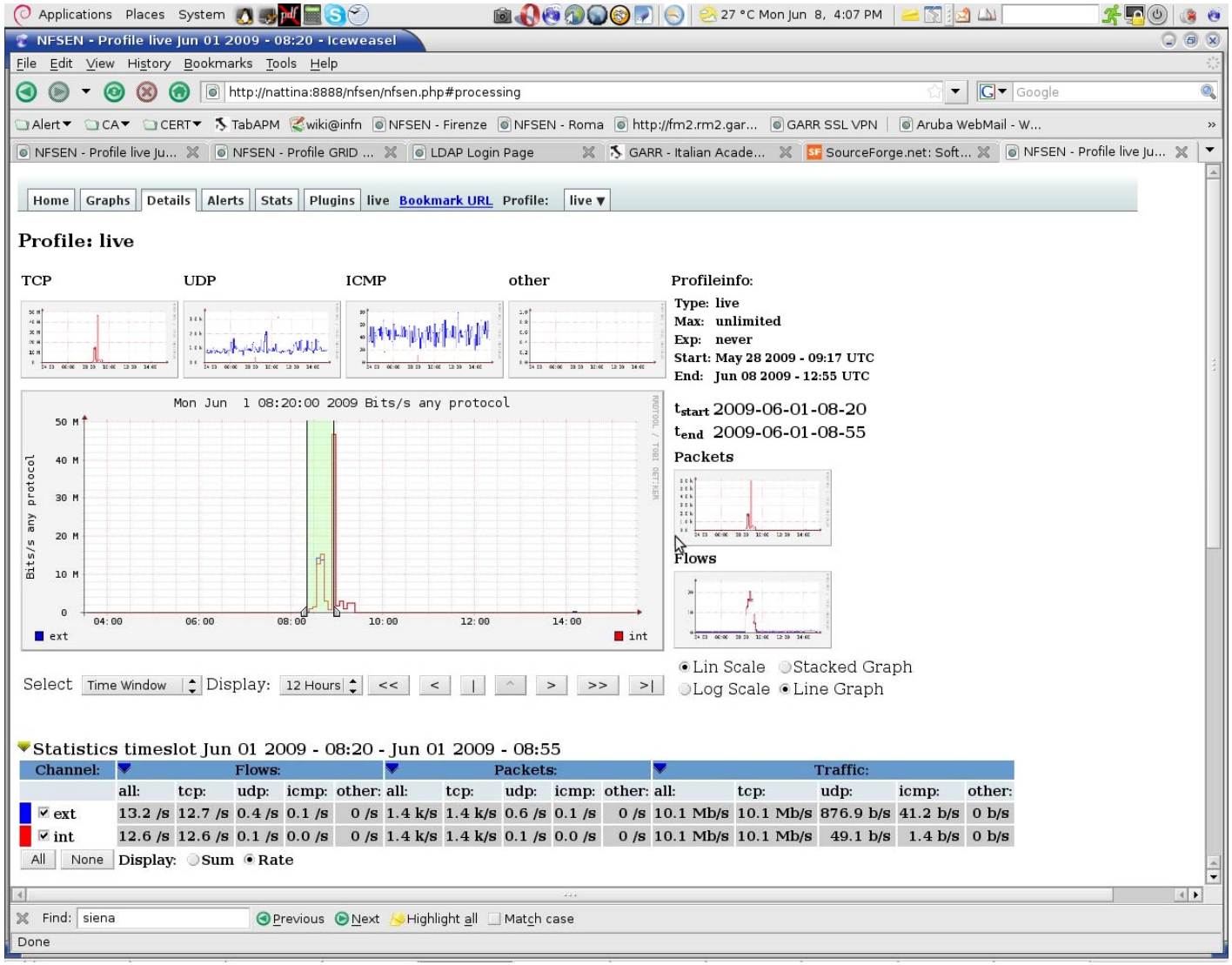
./install.pl ./etc/nfsen.conf

Risultato

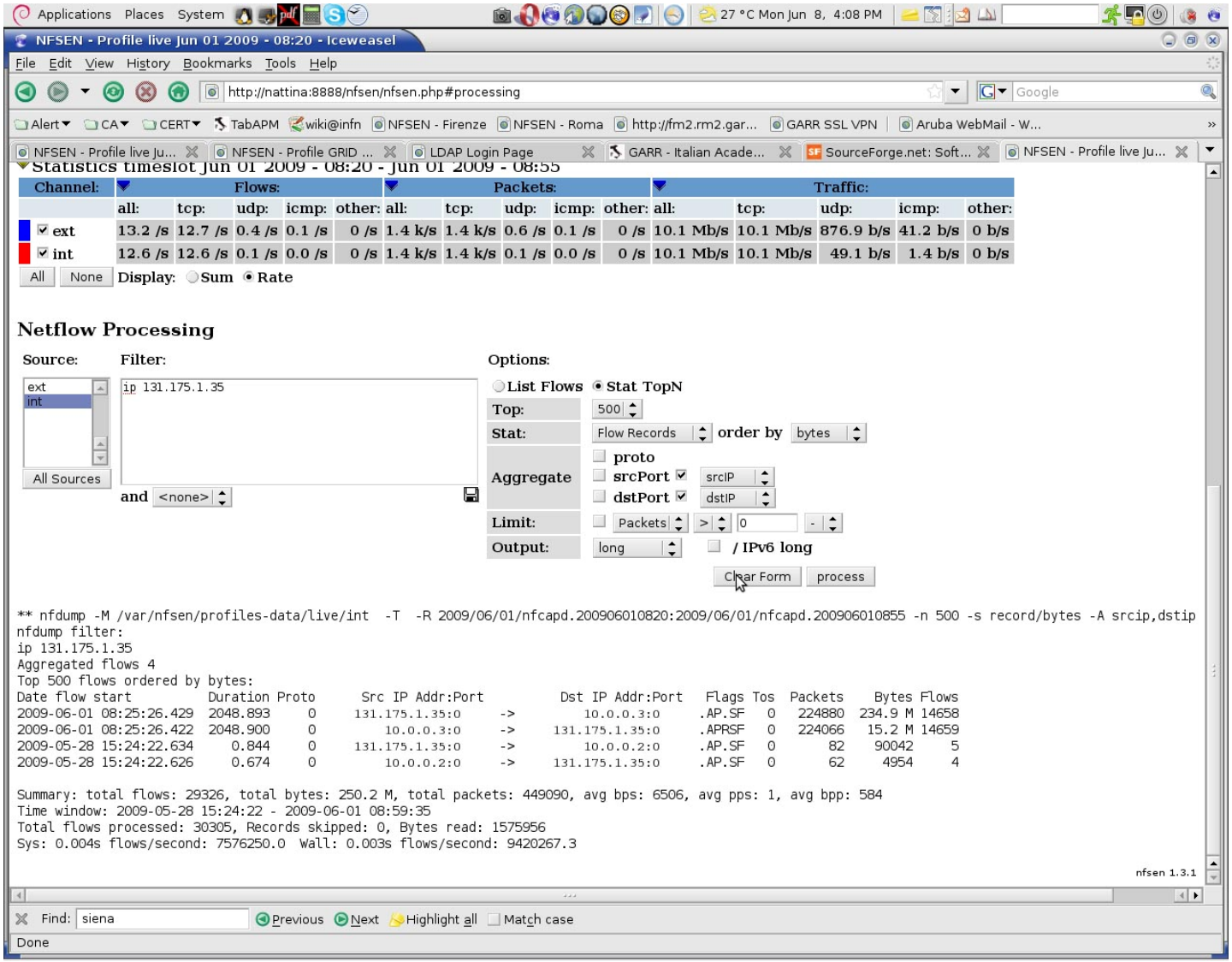


www.garr.it

Dettagli



Processing



Estendere le funzionalità di Nfsen/Nfdump

- scripting + nfdump
- nfsen plugins
  - Come scrivere nuovi plugin
  - Es. Plugin DoS

Scripting+nfdump

- Nfdump prevede un output machine readable tramite l'opzione: *-o pipe*
- Il formato di output è riportato nella tabella, ogni campo è 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

Scripting+nfdump

- 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 è AS sorgente e 8075 quello destinazione
  - 114 è l'snmp ifindex di ingresso e 88 l'snmp ifindex di uscita
  - 1 è il pacchetto che contiene 45 bytes

Come scrivere nuovi plugin

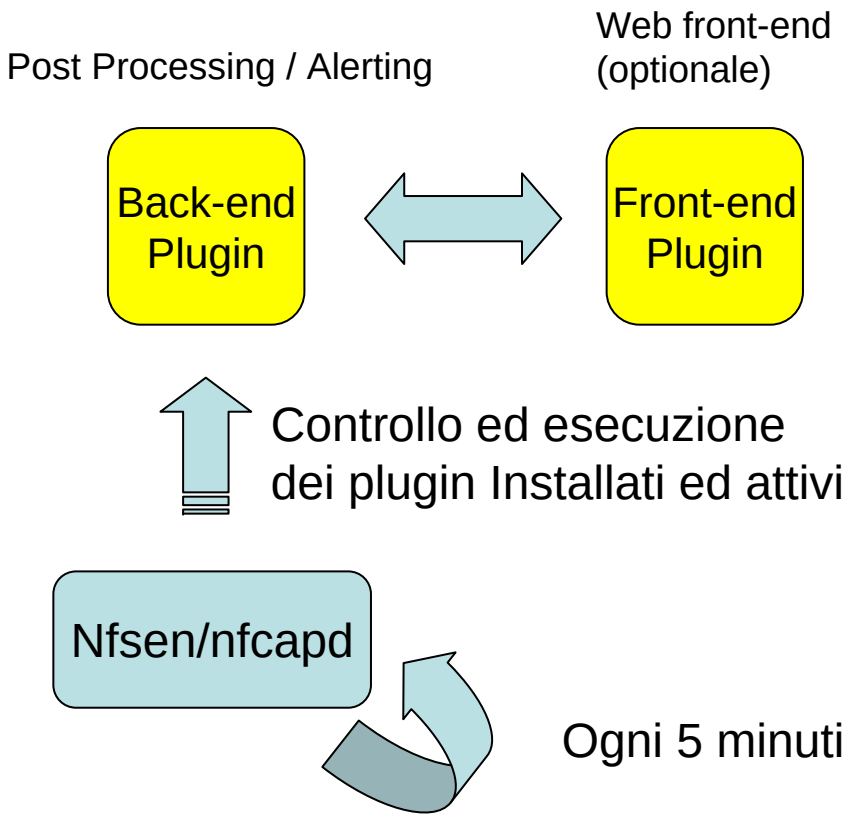
- Nfsen plugin:

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

  Sono composti da tue parti:

  1. Back-end, scritti in linguaggio perl, hanno lo scopo di elaborare i dati
  2. Front-end, scritti in linguaggio php, fanno da interfaccia al Back-end

- I file di un plugin devono avere lo stesso nome:
  - Back-end:<nome_plugin>.pm
  - Front-end:<nome_plugin>.php



Back-end plugin

Funzioni predefinite:

Init

- Questa funzione viene chiamata quando il plugin viene caricato. Lo scopo è di dare al plugin la possibilità di inizializzarsi.
- 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 è di dare la possibilita' al plugin di terminare in maniera corretta.

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

Back-end plugin

Run

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

- è il cuore del plugin.

```
Es.
sub Run {
    my $argref = shift;
    my $profile = $$argref{'profilè};    #il nome del profilo sul quale è installato
    my $profilegroup = $$argref{'profilegroup'}; #gruppo di profili
    my $timeslot = $$argref{'timeslot'}; #il timeslot corrente (es. 200905201015)

    # qui dovrebbe essere inserito il codice del plugin
}
```

Variabili predefinite


```
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
my %profileinfo = NfProfile::ReadProfile($profile, $profilegroup); #un hash table che contiene tutti i dati relativi al profilo in uso:

%profileinfo = (
'description' => [ 'The profile' ],                    # Array of comment lines
'name' => 'live',                                      # name of profile
'group' => '.',                                        # name of profile group
'tbegin' => 1177711200,                                # UNIX time of begin of profile
'tcreate' => 1177711200,                               # UNIX time of create time of profile
'tstart' => 1177711200,                                # UNIX time of start time of profile data
'tend' => 1177969800,                                  # UNIX time of end time of profile
'updated' => 1177969800,                               # UNIX time of last update
'expire' => 24,                                        # Max lifetime of profile data in hours 0 = no expire time
'maxsize' => 0,                                        # Max size of profile in bytes 0 = no limit
'size' => 12354,                                       # Current size of profile in bytes
'type' => 0,                                           # Profile type: 0: life, 1: history profile, 2: continuous profile
'locked' => 0,                                         # profile locked - 0 or 1
'status' => "OK",                                      # status of profile
'version' => 130,                                      # version of profile.dat 130 for version 1.3.0
'channel' => {                                         # hash of all channels in this profile
'               upstream' => {                         # channel name as key
                        'order' => '1',                # display order in graph
                        'sign' => '+',                 # display on + or - side of y-axis
                        'colour' => '#abcdef'          # channel colour
                        'sourcelist' => 'upstream'     # array of ':' separated list of sources
                        },
                }

);
```

Front-end plugin (opzionale)

- Funzioni predefinite:

  <nome_plugin>_ParseInput (obbligatoria)

  - è 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 è selezionato il tab dei Plugin
  - Puo' essere utilizzata per ritornare messaggi

```
Es.
function plugindiprova_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
}
```

Comunicazione tra back-end e front-end (1/2)

- Nel front-end:
  - **Richieste al back-end:**

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

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

  - **Risposte del back-end:**

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

Comunicazione tra back-end e front-end (2/2)

- Nel back-end

Al fine di potere esequire una funzione del plugin di back-end da quello di front-end, occorre definire alcune variabili.

- Gestione delle richieste del front-end:

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

- **Ritorno dei risultati al front-end:**

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

Esempio completo

Front-end:

// Richieste al back-end:

```
$opts = array();

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

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

// chiamata alla funzione in back-end
$out_list = nfsend_query("Pluginname::try", $opts);
```

//Risposte del back-end:

```
// if $out_list == FALSE – it's an error
if ( !is_array($out_list) ) {
    SetMessage('error', "Errore del plugin di back-end");
    return FALSE;
}
$string = $out_list['string'];
$othercolours = $out_list['othercolours'];

print “Il back-end ha risposto: <b>$string</b><br>\n";
```

back-end:

#Gestione delle richieste del front-end:

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

# Lettura dati del front-end
#due scalari
my $colour1 = $$opts{'colour1'};
my $colour2 = $$opts{'colour2'};

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

#Ritorno dei risultati al front-end:

```
# 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 plugind di back-end."
$args{'othercolours'} = \@othercolours;

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

Alert Condition

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.
- I flussi "matchati" dal filtro vengono registrati in una directory apposita ed il nome del file è passato alla subrutine nella variabile ***$$argref{alertfile}***

```
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;
}
```

Alert Action

- 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 è un array che contiene tanti elementi quanti sono gli elementi di $_POST che si vogliono parsare.

Ad ogni elemento è 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 è il puntatore al valore da validare
$opts è 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 gli indirizzi IP della macchina GARR coinvolta nel DoS e l'orario in cui è avvenuto

Esempio: Plugin DoS

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

Esempio: plugin DoS

- Back-end plugin DOS.pm:

```
package DOS;
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/DOS_creo_tabella 2>
                          /mnt/nfsen/plugins/appoggio.noc")
```

Esempio: plugin DoS

- Alert plugin: DOS_alert.pm:

```
sub alert_action {
    ......
     package DOS_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_DOS $timeslot \
     2> /mnt/nfsen/plugins/log_DOS");
     .....
}
```

Esempio: plugin DoS

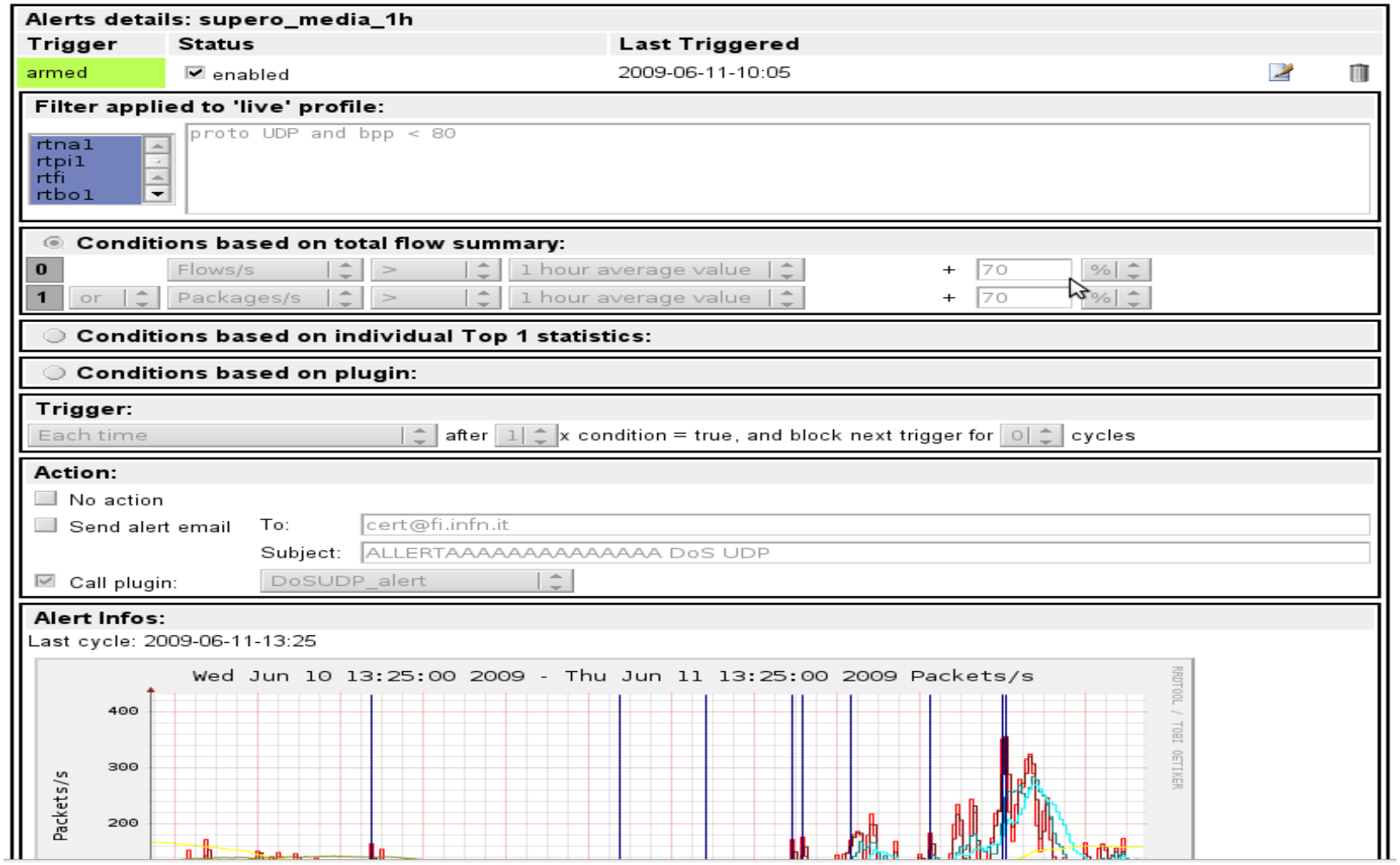
- Front-end plugin DOS.php:

```
<?php
 /* DOS plugin*/
      function DOS_ParseInput( $plugin_id ) {}

      function DOS_Run( $plugin_id ) {
            $VARDIR = "/mnt/nfsen/www/plugins";
            print "<h3>DOS Logfile</h3>\n";
            $logfile = "$VARDIR/DOS_output.log";
            echo htmlspecialchars($logfile) ;
            include $logfile;
            print "</pre>\n";
      } // End of DOS_Run
?>
```

Esempio: plugin DoS

- Configurazione alert:



Esempio: plugin DoS

- RISULTATO:

Home | Graphs | Details | Alerts | Stats | Plugins | live | Bookmark URL | Profile: | live

NOC

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

Appendici

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

Esempi di configurazione: JunOS (v5)

- 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*

Esempi di configurazione:Cisco (v5)

- 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)

- 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
```

Esempi di configurazione:Cisco(v9)

- 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(v9)

- 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

Configurazione switch

- 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
```

Riferimenti

- Nfdump: http://nfdump.sourceforge.net
- Nfsen: http://nfsen.sourceforge.net
- Cisco Netflow: https://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html
- Juniper: http://www.juniper.net/techpubs/software/junos/junos94/swconfig-services/swconfig-services-TOC.html
- Nprobe: http://www.ntop.org/nProbe.html

Contatti

- Per informazioni potete scrivere a:
  - Nino.Ciurleo@garr.it
  - Alessandro.Inzerilli@garr.it
  - Simona.Venuti@garr.it