UNIVERSITA' DEGLI STUDI DI PADOVA
FACOLTA' DI INGEGNERIA INFORMATICA
CORSO DI LAUREA IN INGEGNERIA INFORMATICA

MONITORAGGIO DI UN'INFRASTRUTTURA INFORMATICA AZIENDALE TRAMITE SOFTWARE OPEN-SOURCE

Laureando: Pudia Domenico
Relatore: Dott. Fantozzi Carlo
Correlatore: Dami Dario
Correlatore: Galvagni Stefano

Data Laurea: 27/09/2010
Anno Accademico: 2009-2010

ABSTRACT

La relazione di tirocinio tratta la realizzazione di un sistema di monitoraggio per un'infrastruttura informatica aziendale attraverso l'utilizzo esclusivo di software open-source. I software che ho scelto di utilizzare sono stati: Nagios, Cacti ed MRTG. Con questi applicativi sono state monitorate tre diverse reti con caratteristiche ed usi differenti. Nagios è stato utilizzato per il monitoraggio degli apparati connessi in rete e per le notifiche; Cacti per la creazione dei grafici degli host e servizi monitorati; MRTG per la generazione dei log sull'utilizzo della banda. Prima dell'implementazione è stata affrontata sia l'analisi e la progettazione del sistema di monitoraggio che la stesura della documentazione, con lo scopo di dare una visone globale dell'infrastruttura di rete. Infine, è stato trattato lo studio e la risoluzione di un problema su un applicativo dedicato al monitoraggio di domini internet.

SOMMARIO

Capitolo 1 – Introduzione....1

Capitolo 2 – Strumenti Utilizzati

2.1 Software....3

2.1.1. Nagios....3

2.1.2. Cacti....8

2.1.3. MRTG....10

2.2 Protocolli....11

2.2.1. Il protocollo TCP/IP....11

2.2.2. Il protocollo UDP....14

2.2.3. Il protocollo SNMP....14

Capitolo 3 – Analisi e progettazione....17

3.1 Analisi della rete....18

3.1.1. Analisi della rete “ufficio”....18

3.1.2. Analisi della rete “sala dati”....25

3.1.3. Analisi della rete “Banca Etica”....28

3.2 Creazione di un modello di massima....30

3.2.1. Ambienti da monitorare....31

3.2.2. Risorse da monitorare....33

3.2.3. Notifiche degli eventi....38

3.2.4. Creazione dei grafici....42

Capitolo 4 – Implementazione....45

4.1 Implementazione ufficio....50

4.2 Implementazione sala dati....54

4.3 Implementazione Banca Etica....57

Capitolo 5 – Modifiche

Simple Faileover....60

Capitolo 6 – Conclusioni....64

Appendice

Bibliografia

Ringraziamenti

Capitolo 1

INTRODUZIONE

Col passare degli anni si sta assistendo sempre di più ad una notevole evoluzione delle reti aziendali e dei relativi servizi ad esse connesse: questo avviene grazie ai sistemi di virtualizzazione, che permettono di creare una versione virtuale di una risorsa normalmente fornita fisicamente. Grazie a questi sistemi qualunque risorsa hardware o software può essere virtualizzata permettendo così una riduzione dei costi hardware e dei consumi energetici con tempi di implementazione estremamente ridotti.

L'aumento delle dimensioni e della complessità di una rete aziendale fa sorgere l'esigenza di monitorare le funzioni "vitali" delle diverse macchine, virtuali e non, insieme ai servizi ospitati su ognuna di esse.

I sistemi di monitoraggio sono ormai diventati uno strumento indispensabile per la gestione di una rete aziendale sia di piccole che di grandi dimensioni.

Ad esempio, un sito internet momentaneamente non raggiungibile può causare dei disservizi e quindi provocare danni, sia a livello di immagine aziendale che a livello economico.

Inoltre un sistema di monitoraggio oltre a controllare i servizi erogati deve anche monitorare le risorse interne della rete, perché un disservizio che colpisce una pagina internet potrebbe essere causato anche da un *crash* della macchina che ospita il *web-server*.

In definitiva il compito di un buon sistema di monitoraggio è di dare una visione globale della rete in tempo reale, controllando lo stato dei servizi e delle risorse interne ed in caso di anomalie inviare delle notifiche di allarme agli amministratori della rete.

Naturalmente oltre all'aspetto puramente funzionale del sistema di monitoraggio è di primaria importanza garantire la sicurezza dei dati che transitano sulla rete: di conseguenza, la macchina che ospita il sistema di management deve essere protetta sia dal lato *front-end* con l'ausilio di protocolli di crittografia, che dal lato *back-end* grazie a politiche di *firewalling*.

Da questi presupposti nasce la mia attività di stagista all'interno dell'azienda Sianet che ha avuto come scopo il monitoraggio di un'infrastruttura informatica aziendale tramite software open-source.

L'azienda ha la necessità di monitorare tutte le risorse ed i servizi dislocati in tre reti diverse tra loro, rispettivamente: ufficio, sala dati e Banca Etica. Ognuna delle reti è destinata ad un utilizzo specifico: l'ufficio è la sede operativa dove lavorano i dipendenti e costituisce il nucleo operativo; la sala dati è il cuore dell'azienda dove sono ospitati i server di produzione; Banca Etica rappresenta un cliente di cui si gestisce la rete ed i relativi servizi ospitati.

L'obiettivo dello stage è stato quello di monitorare tutta l'infrastruttura aziendale sia attraverso controlli attivi che andavano a notificare il problema al verificarsi di determinate condizioni, sia attraverso controlli proattivi che garantissero delle notifiche prima del verificarsi di situazioni di stallo, consentendo quindi di intervenire preventivamente per risolvere un determinato problema.

Il tutto veniva corredato dalla creazione di grafici rappresentanti le performance della rete e dei relativi servizi così da avere una visione globale dell'infrastruttura di rete.

Capitolo 2

STRUMENTI UTILIZZATI

2.1 Software utilizzati

Per la realizzazione del sistema di monitoraggio si è fatto uso di tre software *open-source* che ho appositamente scelto: Nagios, Cacti ed MRTG. Questi software sono stati installati su tre macchine virtuali dedicate unicamente al monitoraggio dell'ufficio, della sala dati e di Banca Etica; queste sono state equipaggiate con sistema operativo Linux, server HTTP Apache, il motore di database MySQL ed il linguaggio PHP. Ognuna delle macchine virtuali è stata creata con le seguenti caratteristiche software:

- Sistema operativo Ubuntu server versione 10.04 a 64bit;
- MySQL versione 5;
- Apache 2;
- PHP versione 5;
- Hardisk minimo 8Gb;
- Ram minima 512Mb;
- Doppia scheda di rete.

2.1.1. NAGIOS

La scelta di utilizzare Nagios come sistema di monitoraggio nasce dall'esigenza di utilizzare un prodotto flessibile, versatile, funzionale e *open-source.* Queste sono le caratteristiche che identificano pienamente questo software.

Nagios nasce nel 2002 dalle ceneri di NetSaint, che è stato uno dei migliori software di monitoraggio di reti a livello *open-source*. Il suo padrino Ethan Galstad già creatore di NetSaint, definì Nagios con l'acronimo "Notices Any Glitch In Our System", che sta a significare "Notifichiamo qualsiasi piccolo problema sul nostro sistema".

Il monitoraggio di un'infrastruttura informatica nasce dall'esigenza di tenere sotto controllo host e servizi di diverso tipo con lo scopo di mantenere efficiente la rete e notificare eventuali problemi. Nagios svolge l'attività di monitoraggio attraverso l'utilizzo di *plugin*, questi si comportano come delle sonde ed in caso di anomalie inviano degli allarmi al sistema centrale, che provvede poi a inviare delle notifiche agli amministratori della rete.

Nagios è stato concepito e sviluppato per lavorare in ambienti *Unix-like*, cioè su sistemi operativi che sono progettati seguendo le specifiche dei sistemi Unix.

Fig. 2.1 – Interfaccia web Nagios

Nagios è dotato di un'interfaccia web che viene gestita attraverso Apache, un *web server open-source* molto diffuso che è in grado di operare sia su sistemi operativi *Unix-like* che *Microsoft*.

Apache oltre a fornire funzioni di trasporto delle informazioni e di collegamento offre anche funzioni di controllo per la sicurezza, infatti nel nostro caso gestisce gli accessi all'interfaccia web Nagios. Il *web server* Apache utilizza inoltre una libreria denominata "*gd library*", che serve per la creazione dinamica di immagini GIF, PNG, BMP e JPG.

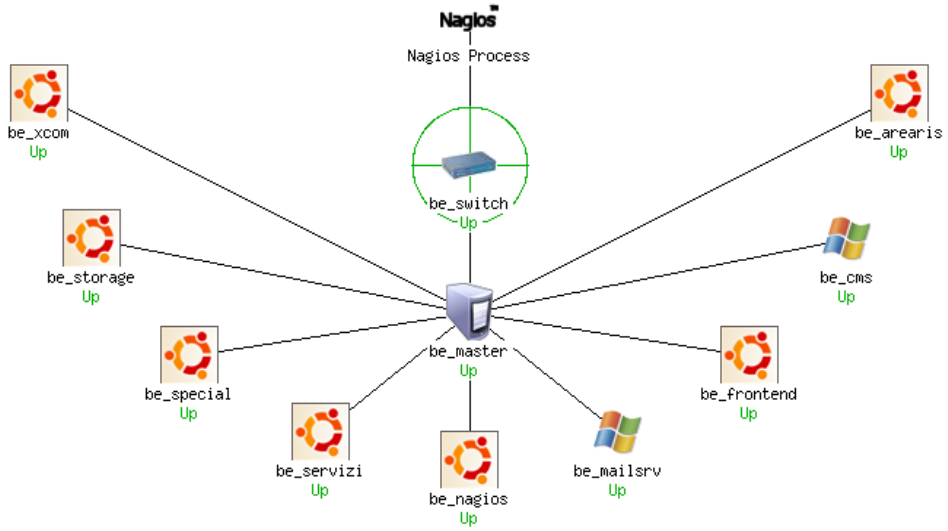


Fig. 2.2 – Mappa generata automaticamente da Nagios

Questa libreria disegna automaticamente immagini complete di linee, archi e testi, come ad esempio quella riportata in Fig. 2.2. Nagios sfrutta la "*gd library*" sia per creare i grafici di raggiungibilità degli host e dei relativi servizi connessi a ciascuno di esso, sia per disegnare le mappe che descrivono la rete.

Il nucleo di Nagios è rappresentato dai cosiddetti plugin che servono a controllare lo stato di un host o di un servizio.

I plugin sono degli eseguibili o script scritti in C, Perl o qualsiasi altro linguaggio di programmazione, Nagios non si interessa dei meccanismi interni dei plugin, richiede però che gli stessi rispettino degli standard.

Di seguito verranno illustrati gli standard più importanti.

Un plugin deve poter essere richiamato tramite linea di comando, quindi deve fare uso dello *Standard Output* (*STDOUT*) che permette di stampare a video le informazioni di ritorno del plugin stesso permettendo così di eseguire test e di verificarne il funzionamento. Tutte le informazione restituite tramite "*STDOUT*" vengono poi automaticamente pubblicate sullo "*Status Information"* del relativo servizio o risorsa monitorata, che è presente sull'interfaccia web. Di seguito le opzioni che un plugin deve gestire quando viene richiamato da riga di comando:

- *"**-V**" o "**--version**"* restituisce la versione del plugin
- *"**-h**" o "**--help**"* restituisce la guida d'uso del plugin
- *"**-v**" o "**--verbose**"* mostra riga per riga le operazioni che effettua il plugin durante la sua esecuzione
- *"**-t**" o "**--timeout**"* permette di impostare il tempo di timeout
- *"**-w**" o "**--warning**"* permette di impostare la soglia di *warning*
- *"**-c**" o "**--critical**"* premette di impostare la soglia *critical*
- *"**-H**" o "**--hostname**"* permette di impostare il nome dell'host

```
benagios@benagios:/usr/local/nagios/libexec$ ./check_ping -h
check_ping v1810 (nagios-plugins 1.4.11)
Copyright (c) 1999 Ethan Galstad <nagios@nagios.org>
Copyright (c) 2000-2006 Nagios Plugin Development Team
      <nagiosplug-devel@lists.sourceforge.net>

Use ping to check connection statistics for a remote host.

Usage:check_ping -H <host_address> -w <wrta>,<wpl>% -c <crta>,<cpl>%
 [-p packets] [-t timeout] [-4|-6]
Options:
 -h, --help
 -V, --version
 -4, --use-ipv4
 -6, --use-ipv6
 -H, --hostname=HOST
 -w, --warning=THRESHOLD
 -c, --critical=THRESHOLD
 -p, --packets=INTEGER
 -t, --timeout=INTEGER
```

Fig. 2.3 – Esecuzione del comando "check_ping -h"

In figura 2.3 è riportato un esempio di interrogazione del plugin "*check_ping*" con l'opzione "*-h*", che stampa a schermo la guida d'uso.

Oltre allo "*STDOUT*" un altro standard da rispettare è quello dei codici di ritorno (*return codes*), infatti ogni plugin deve necessariamente restituire un codice di ritorno che viene associato allo stato del servizio o della risorsa monitorata secondo lo standard "*POSIX*":

Valore Numerico	Stato del Servizio	Descrizione dello Stato
0	OK	Il plugin ha potuto testare il servizio e questo ha funzionato correttamente.
1	Warning	Il plugin ha potuto testare il servizio ma è stata superata la soglia di "warning" oppure sembra che non funzioni correttamente.
2	Critical	E' stata superata la soglia "critical" oppure il plugin non è stato avviato correttamente.
3	Unknown	Argomenti errati da linea di comando o il plugin non ha potuto testare lo stato del dato host/servizio.

Fig. 2.4 –Return codes dei plugin

Il risultato ottenuto dall'esecuzione di un plugin determina l'attuale stato di salute di un host o di un servizio della rete. La flessibilità di Nagios sta appunto nei "plugin". Se nasce l'esigenza di monitorare un nuovo servizio, basta scrivere un apposito script capace di controllare quel tipo di servizio.

Per l'esecuzione dei plugin sugli host, Nagios si basa su due client; "*NRPE*" e "*NSCA*", compatibili solo con sistemi operativi di tipo Linux. Per il monitoraggio degli host Linux e relativi servizi si è scelto di utilizzare il *client NRPE*.

NRPE è l'acronimo di "*Nagios Remote Plugin Executor*", viene utilizzato su un host remoto per il controllo delle risorse e dei servizi che esso ospita.

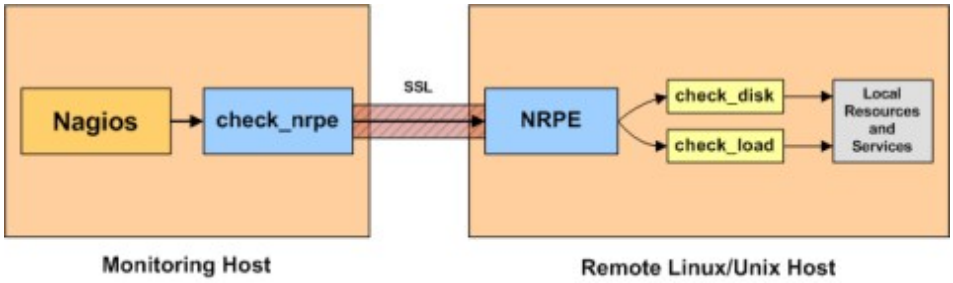


Fig. 2.5 – Funzionamento NRPE

NRPE esegue, sul computer su cui gira, il plugin richiesto da Nagios e invia al sistema di monitoraggio il risultato della sua esecuzione. Il risultato viene mandato attraverso la rete con protocollo *SSL* (*Secure Sockets Layer*), un protocollo di crittografia che permette una comunicazione sicura ed un'integrità dei dati su reti TCP/IP. *NRPE* opera attraverso la porta TCP 5666.

Per il monitoraggio degli host Windows e relativi servizi si è scelto di utilizzare *NSClient++*. Questo client ha il medesimo funzionamento del client *NRPE*.

2.1.2. Cacti

Oltre alla necessità di monitorare le macchine ed i relativi servizi ospitati si è reso necessario realizzare dei grafici sulle diverse performance della rete. Per soddisfare questa necessità si è fatto ricorso ad un altro software *open-source* chiamato Cacti.

Cacti è un software per la realizzazione di grafici tramite l'utilizzo di un database chiamato *RRDtool* (*Round-Robin Database tool*).

Le tre strutture principali sono:

- ***i grafici***: possono essere a barre o composti da linee, possono rappresentare i dati tramite diverse scale o presentare legende e sintesi testuali delle informazioni.
- ***i device***: sono le apparecchiature da controllare, server, stampati o qualsiasi altro dispositivo dotato di supporto SNMP
- ***i data source***: sono le sorgenti dati o ancora meglio l'insieme delle tecniche usate per collezionare le informazioni provenienti dai dispositivi controllati.

I grafici sono dotati di funzioni zoom che consentono di esplorare precisi momenti temporali per un'analisi accurata.

Il database RRDtool con il quale lavora Cacti è un programma *open-source* che permette di memorizzare misurazioni effettuate nel tempo e ricavarne diagrammi riguardanti il traffico della rete, l'utilizzo delle risorse ecc., si basa sul concetto del Round Robin in cui i nuovi elementi vengono aggiunti sovrascrivendo i dati più vecchi. In pratica il database è circolare, quindi una volta raggiunta la fine il puntatore si sposta di nuovo sul primo elemento e inizia a sovrascrivere i dati.

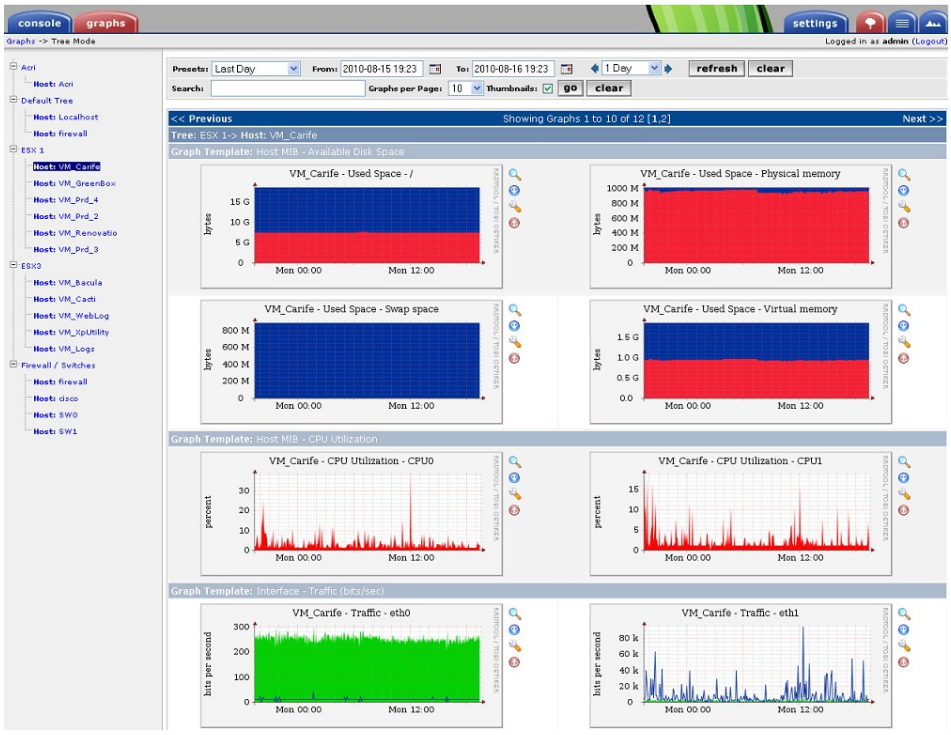


Fig. 2.6 – Grafici Cacti

Il database viene popolato attraverso i dati che provengono dalle interrogazioni inoltrate tramite protocollo SNMP.
Uno dei punti di forza di Cacti è la facilità di configurazione garantendo ugualmente un'elevata flessibilità.
L'applicativo è dotato di un *front-end* con interfaccia web scritta in PHP, inoltre permette di gestire più utenti con viste personalizzate e diversi livelli di permesso per le modifiche.
Cacti permette una vasta gamma di personalizzazioni, che vanno dall'aggiunta di macchine da monitorare, alla definizione di nuove sorgenti di dati, ottenibili tramite SNMP o script ad hoc.

2.1.3. MRTG

MRTG (*Multi Router Traffic Grapher*) è un software open-source per il monitoraggio delle apparecchiature di rete tramite il protocollo SNMP. È stato sviluppato utilizzando i linguaggi di programmazione Perl e C.
Questo applicativo viene utilizzato dai network manager per la sua duttilità e per la facilità di utilizzo nel controllo degli apparati di rete come, ad esempio, i router della *CISCO Systems*, inoltre è in grado di operare sia su sistemi operativi *Unix-like* che *Microsoft*.
Questo applicativo è stato utilizzato per calcolare il traffico generato dalle schede di rete dei vari dispositivi e per produrre i relativi file di *log* necessari al funzionamento di un plugin Nagios chiamato "*check_mrtg_traffic*". Il plugin riceveva come parametri d'ingresso i log prodotti da MRTG che venivano poi elaborati per calcolare il traffico medio e quello istantaneo.

2.2 Protocolli utilizzati

Prima di parlare dei protocolli utilizzati è bene chiarire il concetto di protocollo.

Nel campo delle telecomunicazioni, per protocollo di rete si intende la definizione formale delle regole che devono rispettare due apparecchiature elettroniche collegate per poter instaurare una comunicazione corretta. Queste apparecchiature possono essere host, server, palmari, telefoni, cellulari, monitor, stampanti, sensori ecc. L'aderenza ai protocolli garantisce che due software in esecuzione su diverse macchine possano comunicare efficacemente, anche se sono stati realizzati indipendentemente. È evidente l'importanza della standardizzazione dei protocolli di rete.

Uno dei protocolli più usati per effettuare i monitoraggi di rete è il protocollo TCP/IP, il quale specifica come devono essere inviati i pacchetti all'interno di una rete, sia essa di tipo LAN, Wireless o Internet. Altro protocollo utilizzato è quello SNMP che consente la gestione e la supervisione di apparati collegati in una rete, questo utilizza il protocollo UDP a livello di trasporto delle informazioni.

2.2.1. Protocollo TCP/IP

Lo standard TCP/IP (*Transmission Control Protocol e Internet Protocol*) è stato sviluppato nella seconda metà degli anni '70 dalla "*DARPA*" (*Defence Advanced Research Project Agency*), allo scopo di permettere la comunicazione tra diversi tipi di computer e di reti di computer. I due protocolli che compongono il TCP/IP si occupano di aspetti diversi delle reti di computer. L'*Internet Protocol*, la parte IP di TCP/IP, è un protocollo senza connessione che tratta solo l'instradamento dei pacchetti di rete usando il datagramma IP come l'unità fondamentale dell'informazione di rete.

I datagrammi IP hanno lunghezza massima di 65536 byte, e sono suddivisi in due parti: l'intestazione (header) che contiene le informazioni necessarie per instradare il pacchetto, il campo dati, che contiene il messaggio da inviare. Esistono diverse versioni di IP, quella attualmente più diffusa è la versione 4, anche se da anni si cerca di sostituirla con la versione 6 che comprende uno spazio di indirizzamento molto più ampio (IPv6 → $2^{128} \approx 3.4 \times 10^{38}$ indirizzi contro IPv4 → $2^{32} \approx 4 \times 10^{9}$ indirizzi).

Ogni macchina all'interno di una rete è individuata univocamente da un indirizzo IP formato da 32 bit.

Ogni indirizzo IP è composto da due parti: un indirizzo di rete (comune a tutti i nodi di una stessa rete) ed un indirizzo di host. Per semplicità vengono scritti con numeri decimali (da 0 a 255), separati tra loro da dei punti.

Esistono tre classi di indirizzi denominate A, B e C.

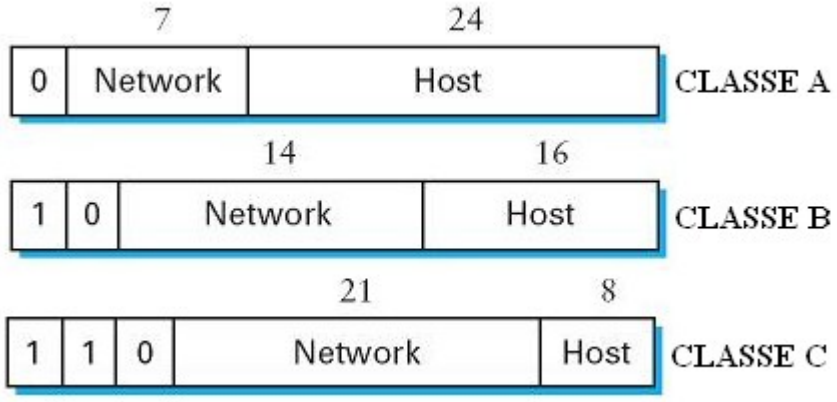


Fig. 2.7 – Classi di indirizzi

Nella figura seguente sono rappresentate le caratteristiche delle tre classi di rete:

Classe	Bit iniziali	Tot numero di reti	Numero max indirizzi host
Classe A	0	128	16777214
Classe B	10	16384	65534
Classe C	110	2097152	254

Fig. 2.8 – Caratteristiche delle classi di rete

Come si può notare le reti di classe A sono poche ma di grandi dimensioni, le reti di classe B sono di medie dimensione e le reti di classe C sono tanti ma di piccole dimensioni.

Per quanto riguarda il *Transmission Control Protocol*, la parte TCP di TCP/IP, nacque negli anni '70 come frutto del lavoro di un gruppo di ricerca del dipartimento di difesa statunitense.

I suoi punti di forza sono l'alta affidabilità e robustezza. La sua popolarità si deve anche grazie ad una sua implementazione diffusa dalla Università di Berkeley, rilasciata senza costi sotto forma di sorgenti. Questo protocollo è stato progettato per utilizzare i servizi del protocollo IP, il quale non offre alcuna garanzia sull'ordine di consegna dei pacchetti, sul ritardo o sulla congestione.

Il TCP garantisce invece che i dati tra le connessioni siano consegnati e che arrivino agli host della rete nello stesso ordine in cui sono stati trasmessi da un altro host della rete. Il canale di comunicazione è costituito da un flusso bidirezionale di byte.

Una delle caratteristiche principali del TCP è la capacità di fornire un servizio di multiplazione delle connessioni su un host, attraverso il meccanismo delle porte.

Una connessione TCP sarà quindi identificata dagli indirizzi IP dei due host e dalle porte utilizzate sui due host.

In questo modo, un server può accettare connessioni da più client contemporaneamente attraverso una o più porte, un client può stabilire più connessioni verso più destinazioni, ed è anche possibile che un client stabilisca contemporaneamente più connessioni indipendenti verso la stessa porta dello stesso server.

2.2.2. Il Protocollo UDP

UDP (*User Datagram Protocol*) è un protocollo di trasporto a pacchetto *stateless*, ovvero non tiene nota dello stato della connessione, per SNMP questo protocollo viene utilizzato a livello di trasporto. A differenza del TCP, l'UDP è un protocollo di tipo *connectionless*, inoltre non gestisce il riordinamento dei pacchetti né la ritrasmissione di quelli persi, ed è perciò generalmente considerato di minore affidabilità. In compenso è molto rapido ed efficiente per le applicazioni "leggere" o *time-sensitive*. Ad esempio, è usato spesso per la trasmissione di informazioni audio o video.

L'UDP fornisce soltanto i servizi basilari del livello di trasporto:

- ***multiplazione delle connessioni***, ottenuta attraverso l'utilizzo delle porte;
- ***verifica degli errori*** attraverso l'utilizzo del checksum.

2.2.3. Il Protocollo SNMP

SNMP (*Simple Network Management Protocol*) è un protocollo che è stato scritto essenzialmente per i gestori dell'ambiente di rete. Grazie ad SNMP è possibile avere diverse informazioni sulle prestazioni di un sistema.

Sostanzialmente un *framework* SNMP è composto da una o più *Stazioni di Gestione* (*Management Station*) e dagli *agenti SNMP* (*SNMP agent*) in funzione sui device di rete.

Le Management Station interrogano gli agent, i quali inviano le informazioni richieste sul device. Solitamente gli agent SNMP di un apparato di rete sono implementati nel *firmware* dello stesso, mentre per quanto riguarda i server, si tratta di servizi software. Gli oggetti gestiti dagli agent, sono raccolti, in ogni device, in un database chiamato "*MIB*" (*Management Information Base*).

Esiste un caso particolare dove l'apparato di rete non viene interrogato, ma è esso stesso a generare il messaggio ed inviarlo alle Management Station.

E' infatti possibile configurare gli agent in modo da inviare un particolare messaggio al verificarsi di determinati eventi ovvero impostare le cosiddette *trap*. Grazie all'impostazione delle trap è possibile ad esempio sapere quando una stampante esaurisce il toner, in quanto al verificarsi dell'errore, l'agent SNMP che esegue il monitoraggio dell'apparato invia alla Management Station un messaggio che identifica il problema, ad esempio "toner esaurito". SNMP utilizza come protocollo di trasmissione UDP in modo da ottenere migliori performance e minore *overhead* della rete. In particolare viene utilizzata la porta UDP 161 per le interrogazioni e le risposte, e la porta UDP 162 come destinazione dei messaggi trap SNMP generate dagli agent SNMP.

L'insieme degli apparati di rete gestiti da SNMP appartengono ad una comunità (*community*). La comunità rappresenta un identificativo che permette di garantire la sicurezza delle interrogazioni SNMP. Un agent SNMP risponde solo alla richieste di informazioni effettuate da una Management Station appartenente alla stessa comunità.

Esistono tre tipi di comunità:

- ***monitor***: permette di lavorare in sola lettura (*Read Only*) , quindi di effettuare solamente interrogazioni agli agent ;
- ***control***: permette tramite gli agent SNMP di effettuare delle operazioni in lettura/scrittura (*Read/Write*) sul dispositivo;
- ***trap***: permette ad un agent di inviare un messaggio trap SNMP alla management station secondo la propria configurazione.

Spesso i nomi delle community di default sono "*public*" per le community di tipo *Read Only* (RO) e "*private*" per le community di tipo *Read/Write* (RW). E' bene modificare queste impostazioni di default per motivi di sicurezza.

Capitolo 3

ANALISI E PROGETTAZIONE

Dopo aver ricercato e descritto il funzionamento dei software che ho scelto di utilizzare per il monitoraggio dell'infrastruttura informatica aziendale, si è passati alla fase di analisi e progettazione.

In questo capitolo viene dapprima affrontata la parte di analisi delle tre reti oggetto del monitoraggio, attraverso questa fase si sono apprese le conoscenze sui dispositivi connessi in rete e sulle loro configurazioni. Successivamente alla fase di analisi è stata prodotta della documentazione con lo scopo di dare una visone globale dell'infrastruttura informatica aziendale, rappresentando i diversi device insieme alle loro configurazioni e collocazioni all'interno delle reti.

Successivamente alla fase di analisi si è passati alla progettazione di un modello di massima con l'obiettivo di minimizzare i tempi ed i costi di implementazione.

Nel successivo capitolo verrà illustrata l'implementazione del sistema di monitoraggio dell'infrastruttura informatica aziendale, fornendo anche una panoramica sulle tecniche di implementazione adottate.

All'interno del capitolo riguardante le modifiche, verrà affrontato lo studio e la risoluzione di un problema in merito ad un applicativo per il monitoraggio dei domini internet.

3.1 Analisi della rete

Uno dei passi fondamentali nel monitoraggio di un'infrastruttura informatica è l'analisi della rete che permette di avere una visone globale dell'infrastruttura da monitorare. Prima di procedere alla stesura della documentazione relativa alle tre diverse reti (ufficio, sala dati e Banca Etica) si è reso necessario catalogare tutti gli apparati connessi in rete. Sono stati raccolti dati in merito alle macchine, alla loro configurazione all'interno della rete e alle relative caratteristiche sia hardware che software. L'utilità di redarre questo tipo di documentazione è fondamentale perché oltre ad avere una visione globale della rete permette sia di ottimizzarla sia di trovare eventuali soluzioni in tempo celere nel caso di malfunzionamenti o di altri disservizi.

3.1.1. Analisi della rete "ufficio"

Per la rete "ufficio" è stata prodotta la seguente documentazione.

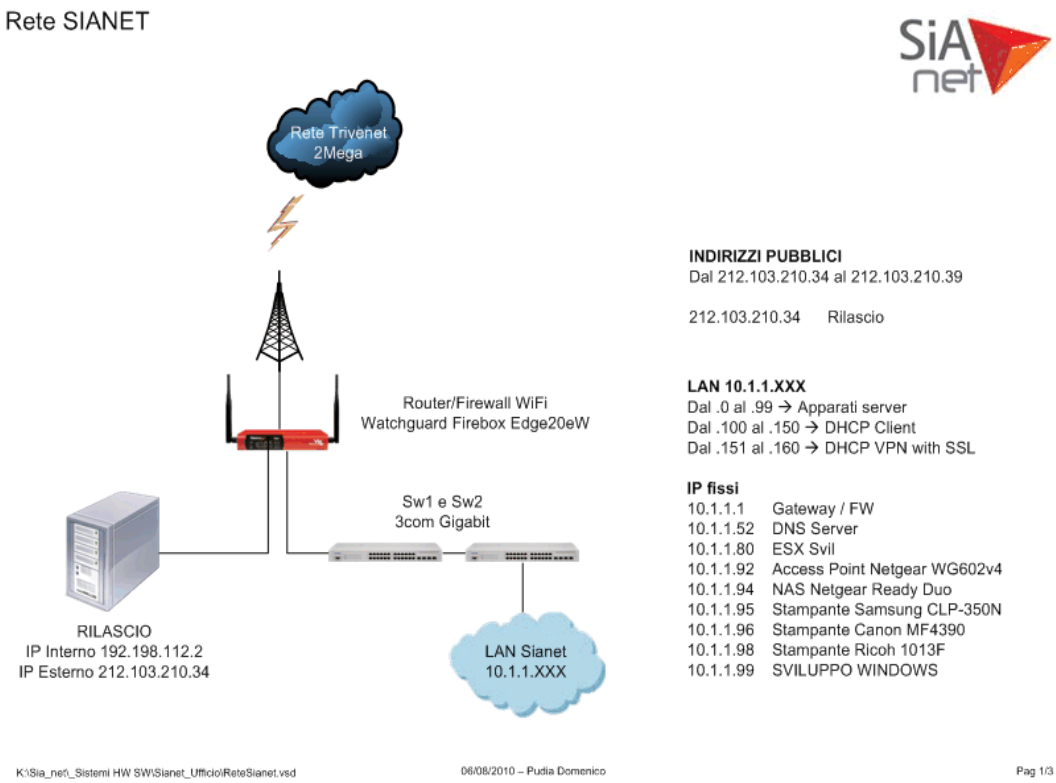


Fig. 3.1 – Rete Sianet *(appendice 1)*

Nella figura 3.1 è rappresenta la rete “ufficio”. In alto è rappresentato l' *Internet Service Provider* (*ISP*), il fornitore che tramite un ponte radio eroga il servizio per la connessione internet. Tutto il traffico da e verso l'esterno passa attraverso un firewall il quale è stato impostato per dividere la rete in due zone:

- ***la rete DMZ*** (D*emilitarized Zone*), dove è collegato l'host denominato “Rilascio”, che è una sottorete raggiungibile sia da reti interne che esterne, permettendo però connessioni esclusivamente dall'interno verso l'esterno;
- ***la rete LAN*** denominata “*LAN Sianet*” realizzata attraverso l'utilizzo di due switch *Gigabit* da 16 porte.

Nel successivo schema è stato rappresentato lo schema dei cavi, i quali sono stati prima catalogati, rilevati e successivamente inseriti nello schema. Con questo tipo di schema è possibile conoscere sia l'ubicazione che il relativo cavo di connessione di ogni apparato di rete.

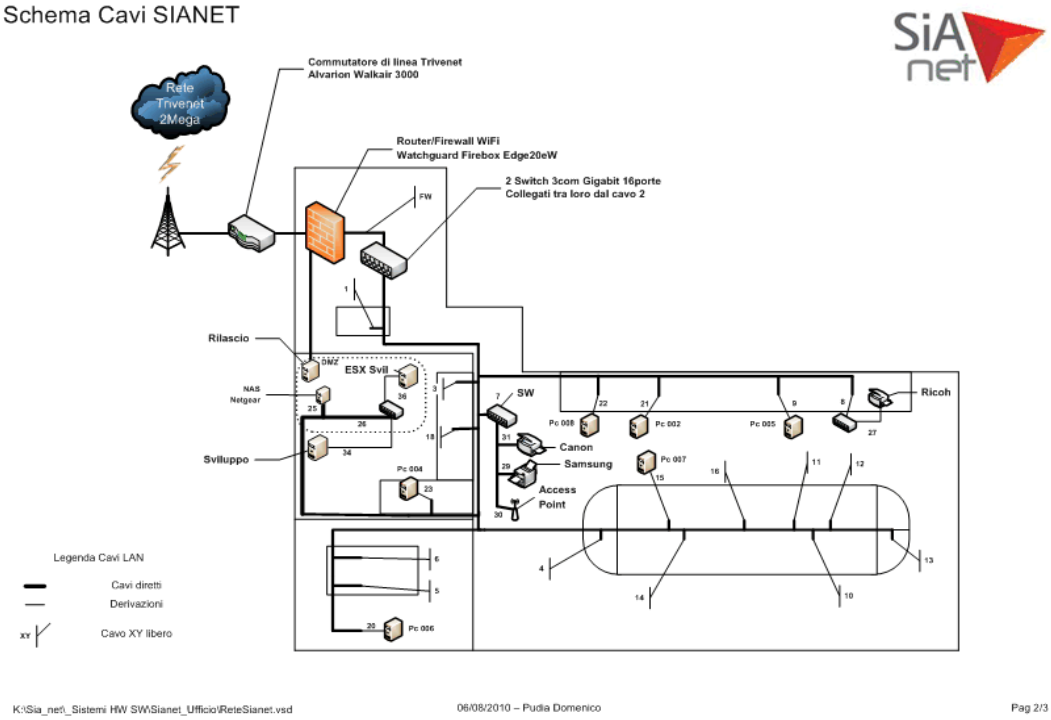


Fig. 3.2 – Schema cavi Sianet *(appendice 1)*

Nel seguente schema è stata riprodotta la connessione dei cavi LAN con annessa etichettatura e descrizione del dispositivo connesso sui due switch Gigabit 16 porte.

Switch 3com Gigabit - Sw2 (superiore)								
Note	Scrivania ingresso	Cavo Uplink/Downlink Switch1	Ufficio - isola centrale	Switch Nilox 5 porte	Ufficio - Scrivania a muro Pc005	Ufficio - isola centrale	Ufficio - isola centrale	Ufficio - isola centrale
Nome Cavo	1	2	4	8	9	10	11	12
Porte Switch	1	2	3	4	5	6	7	8
	9	10	11	12	13	14	15	16
Nome Cavo	13	14	15	16	21	22		
Note	Ufficio - isola centrale	Ufficio - isola centrale	Ufficio - isola centrale Pc007		Ufficio - Scrivania a muro Pc002	Ufficio - Scrivania a muro Pc008		

Switch 3com Gigabit - Sw1 (inferiore)								
Note	Armadio ingresso - Firewall Firebox WatchGuard	Cavo Uplink/Downlink Switch2	Sala Server Switch Nilox 5 porte	Sala Server NAS Netgear Ready Duo	Sala Riunioni angolo LCD	Sala Riunioni - scrivania	Sala Riunioni - scrivania	
Nome Cavo	FW	2	26	25	20	6	5	18
Porte Switch	1	2	3	4	5	6	7	8
	9	10	11	12	13	14	15	16
Nome Cavo	23	3	7	19	17			
Note	Sala Server - scrivania vicino porta	Sala Server - Switch D-Link 2 PC amm.	Switch mobiletto stampante	Cavo a pavimento	Cavo a pavimento			

Fig. 3.3 – Schema connessione cavi sugli switch Gigabit *(appendice 1)*

Nella tabella successiva vi è un riassumendo degli indirizzi IP di classe A, associati ai relativi servizi e dispositivi di rete.

Web	IP	Cavo
Indirizzi Pubblici		
Dal 212.103.210.34 al 212.103.210.39		
Rilascio	212.103.210.34	
LAN 10.1.1.XXX		
Apparati server	Dal .0 al .99	
DHCP Client	Dal .100 al .150	
DHCP VPN with SSL	Dal .151 al .160	
IP Fissi		
Gateway / FW	10.1.1.1	FW
DNS Server	10.1.1.52	
Server ESX Svil	10.1.1.80	36
Access Point Netgear	10.1.1.92	30
NAS Netgear Ready Duo	10.1.1.94	25
Stampante Samsung CLP-350N	10.1.1.95	29
Stampante Canon MF4349	10.1.1.96	31
Stampante Ricoh 1013F	10.1.1.98	27
SVILUPPO WINDOWS	10.1.1.99	35
IP Interno Rilascio	192.168.112.2	DMZ

Configurazione WAN	IP
DNS Primario	212.103.192.10
DNS Alternativo	212.103.195.195
GATEWAY	212.103.210.33

Fig. 3.4 – Tabella riassuntiva degli indirizzi IP della rete "ufficio" (appendice 1)

Oltre agli schemi inerenti alla rete "ufficio" sono stati redatti degli schemi in merito alle macchine virtuali presenti in rete con le relative caratteristiche hardware, software e di rete insieme alle caratteristiche del relativo server che le ospita chiamato "ESX SVILUPPO". In questo server risiedono tutte le macchine virtuali presenti nell'infrastruttura "ufficio":

- **le *macchine operative***;
- **le *macchine di test***;
- **le *macchine template*** che sono delle macchine virtuali "vuote" usate esclusivamente per essere clonate ottimizzando così i tempi di creazione di nuove macchine virtuali;

- **le *macchine ZZZ*** chiamate con questo prefisso perché identificano le macchine poco utilizzate e prossime alla dismissione.

Di seguito lo schema delle caratteristiche hardware e software del server "ESX SVILUPPO":

CAMPO	VALORE
Nome PC	ESX SVILUPPO
Case	PowerEdge T410 Tower Chassis for Up to 6x 3.5" Cabled HDDs with Quad-Pack LED Diagnostics
Product Number	GJLCP4J
Network Adapter	Broadcom NetExtreme II BCM5716 1000Base-T
Storage Adapter	Dell Perc H700 Adapter
Numero Processori	4
Scheda Madre	DELL
Bios	Dell Inc. System BIOS 1.3.8 26/02/2010
RAM	12GB (3x4GB RDIMM dual rank) 1.066MHz
Hard Disk 0	450GB SAS 6Gb/s 15.000rpm 3,5"
Hard Disk 1	450GB SAS 6Gb/s 15.000rpm 3,5"
Hard Disk 2	500 GB Samsung RAID 5
Hard Disk 3	500 GB Samsung RAID 5
Hard Disk 4	500 GB Samsung RAID 5
Sistema Operativo	ESX 4
Accessori	iDRAC6 Embedded BMC, Redundant Power Supply (2 PSU) 580W
Lettore DVD	16X DVD-ROM Drive SATA with SATA Cable for Win2K8 R2

Fig. 3.5 – Caratteristiche hardware server "ESX SVILUPPO" (appendice 2)

Nello schema sottostante sono rappresenti gli indirizzi IP utilizzati dal server "ESX SVILUPPO":

IP	DESCRIZIONE
10.1.1.80	IP interno
10.1.1.81	DRAC

Fig. 3.6 – Indirizzi IP associati a "ESX SVILUPPO" (appendice 2)

come si può notare oltre all'indirizzo IP denominato "*IP interno*" è presente un indirizzo denominato "*DRAC*" (*Dell Remote Access Controller*).

Il DRAC è una soluzione hardware/software di gestione del sistema che offre funzioni di gestione remota, ripristino del sistema da errori e funzioni di controllo dell'alimentazione. E' dotata di microprocessore e memoria ed è alimentata dal sistema nel quale è installata.

Insieme agli schemi inerenti il server è stata prodotta una documentazione in merito alle macchine virtuali ospitati sul server di sviluppo.

Di seguito le caratteristiche hardware delle singole macchine virtuali:

VM	NUM CPU	RAM	HD 0	HD 1	HD 2	HD 3
AAA_ORA11SVIL	1	2Gb	10 Gb	20 Gb		
AAA_ORA9SVIL	1	512 Mb	30Gb			
AAA_UB_10.04_SERVIZI	1	1Gb	50Gb			
AAA_UB_10.04_64_MYSQL	1	1Gb	24Gb			
AAA_UB10.04_64_NAGIOS	1	512Mb	8Gb			
AAA_vSphere Management	1	512Mb	5Gb			
AAA_W2K3R2_IIS	1	1Gb	50Gb	80Gb	80Gb	80Gb
AAA_W2K3R2_SQL2005	1	1Gb	50Gb			
AAA_W2K3R2_SQL2008R2	1	1Gb	50Gb			
AAA_W2K8R2_SERVIZI	1	1Gb	40Gb			
AAA_W2008R2_TFS2010	1	2Gb	40Gb			
AAA_WXP_VSNET	2	1Gb	20Gb			
TPL_W2K8R2_VUOTA	1	2Gb	40Gb			
AAA_WXP_UTILITY	1	256Mb	124Gb			
TST_UB_10.04_BEAREARIS	1	1Gb	50Gb			
TST_UB_10.04_BESPECIAL	1	1Gb	50Gb			
TST_UB_9.04_CLUSTER_MASTER	1	1Gb	8Gb	5Gb		
TST_UB_9.04_CLUSTER_SLAVE	1	1Gb	8Gb	5Gb		
TST_UB9.10_64_Nagios	1	1Gb	15Gb			
TST_W2K3_MOSS2007	2	1Gb	32Gb			
TST_W7UlN_64	1	2Gb	40Gb			
ZZZ_UB_8.10_BECMS	1	512Mb	24Gb			
ZZZ_UB_9.04_BEORG	1	512Mb	8Gb			
ZZZ_UB_9.04_BECMS_PROXY	2	512Mb	8Gb			
ZZZ_W2K3_BECMS	1	512Mb	10Gb	8Gb		

Fig. 3.7 – Caratteristiche hardware macchine virtuali su "ESX SVILUPPO" (appendice 2)

Il nome dato a ciascuna macchina virtuale rispetta delle convenzioni: il prefisso può essere *AAA* (macchine operative) o *TPL* (macchine template) o *TST* (macchine di test) oppure *ZZZ* (macchine poco utilizzate); la parte centrale del nome identifica il sistema operativo della macchina ; la parte finale il nome dato alla macchina. Esempio: *AAA_UB10.04_64_NAGIOS* identifica una macchina operativa (AAA) con sistema operativo Ubuntu versione 10.04 a 64 bit (UB_10.04_64) chiamata "NAGIOS".

IP primario	IP secondario	Proxy	DESCRIZIONE
10.1.1.51		10.1.1.52	AAA_ORA11SVIL
10.1.1.52			AAA_UB_10.04_SERVIZI
10.1.1.53	10.1.2.53	10.1.1.52	TST_UB_10.04_BEAREARIS
10.1.1.54		10.1.1.52	TST_UB_10.04_BESPECIAL
10.1.1.55		10.1.1.52	AAA_vSphere Management
10.1.1.56		10.1.1.52	TST_UB9.10_64_Nagios
10.1.1.57	10.1.2.57	10.1.1.52	AAA_ORA9SVIL
10.1.1.58			
10.1.1.59			
10.1.1.60		10.1.1.52	AAA_W2008R2_TFS2010
10.1.1.61	10.1.2.61	10.1.1.52	AAA_WXP_VSNET
10.1.1.62		10.1.1.52	TST_W7UIN_64
10.1.1.63		10.1.1.52	AAA_W2K3R2_IIS
10.1.1.64		10.1.1.52	AAA_W2K3R2_SQL2008R2
10.1.1.65		10.1.1.52	AAA_W2K3R2_SQL2005
10.1.1.66		10.1.1.52	AAA_UB_10.04_64_MYSQL
10.1.1.67		10.1.1.52	TST_W2K3_MOSS2007
10.1.1.68		10.1.1.52	AAA_UB10.04_64_NAGIOS
10.1.1.69			
10.1.1.70		10.1.1.52	ZZZ_UB_9.04_BECMS_PROXY
10.1.1.71			
10.1.1.72		10.1.1.52	AAA_WXP_UTILITY
10.1.1.73		10.1.1.52	TPL_WXP_VUOTA
Dinamico		10.1.1.52	TPL_W2K8R2_VUOTA
Dinamico		10.1.1.52	TST_UB_9.04_CLUSTER_SLAVE
Dinamico		10.1.1.52	ZZZ_UB_8.10_BECMS
Dinamico		10.1.1.52	ZZZ_UB_9.04_BEORG
Dinamico		10.1.1.52	TST_UB_9.04_CLUSTER_MASTER
Dinamico		10.1.1.52	ZZZ_W2K3_BECMS

Fig. 3.8 – Caratteristiche di rete macchine virtuali su "ESX SVILUPPO" (appendice 2)

Oltre allo schema che rappresenta le caratteristiche hardware di ciascuna macchina virtuale, è stato redatto uno schema che riassume le rispettive caratteristiche di rete.

Dallo schema si può notare che la macchina denominata "*AAA_UB_10.04_SERVIZI*" con indirizzo IP 10.1.1.52 oltre ad essere il *server DNS* svolge anche il ruolo di *proxy,* cioè si interpone tra un client e un server inoltrando le richieste e le risposte dall'uno all'altro.

Il client si collega al proxy invece che al server, e gli invia delle richieste. Il proxy a sua volta si collega al server e inoltra la richiesta del client, riceve la risposta e la inoltra al client. Inoltre implementa dei meccanismi di *cache* per diminuire il traffico di rete.

3.1.2. Analisi rete "sala dati"

Per la rete "sala dati" è stata prodotto lo schema in Fig. 3.9.

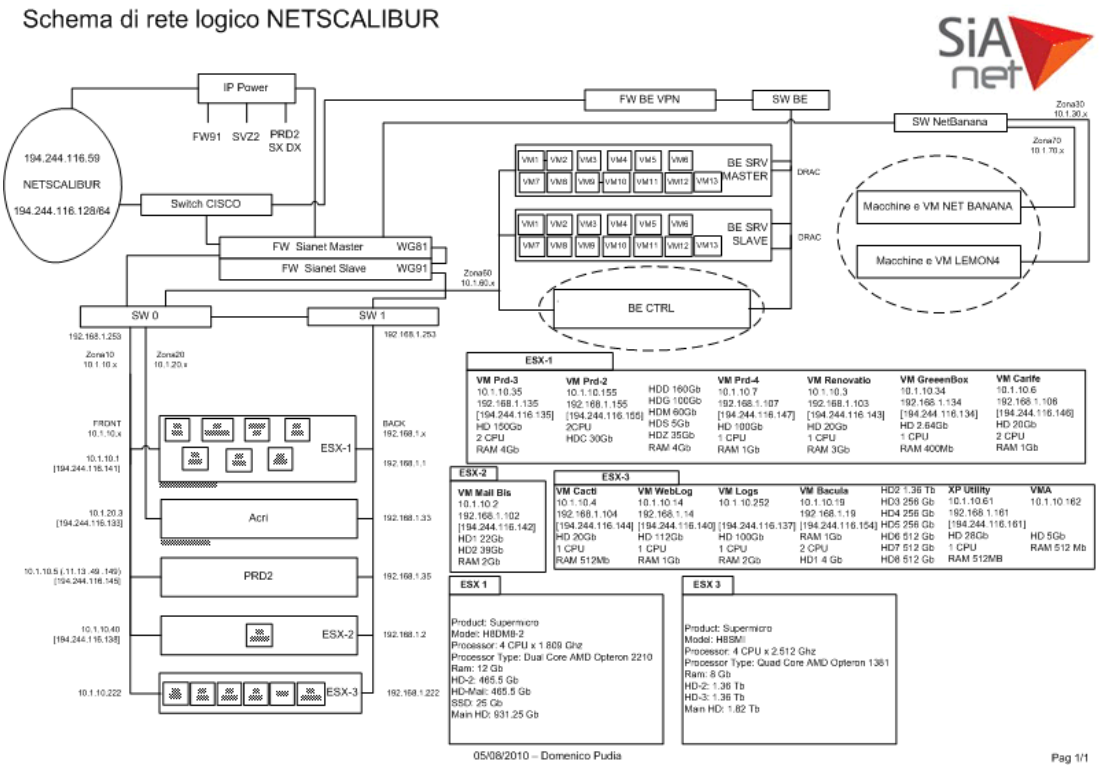


Fig. 3.9 – Schema rete logico "sala dati" *(appendice 3)*

Partendo da sinistra, dove è rappresentato l' *ISP "Netscalibur*", si può notare che sono presenti due dispositivi: un *IP Power,* che è un dispositivo comandato attraverso un browser web che permette di dare o togliere corrente ai dispositivi a esso collegati e si comporta quindi come un interruttore con gestione remota; uno switch *Cisco 2900 XL a cui* sono collegati gli apparati di "*Banca Etica*" e due firewall in *HA* (*High Availability*). Questo tipo di connessione permette in caso di malfunzionamento del firewall primario l'intervento del firewall secondario senza avere discontinuità di servizio.

Dallo schema in Fig. 3.9 si può notare che la rete "sala dati" è stata suddivisa nelle seguenti cinque zone:

- ***zona 10*** rappresentata dagli indirizzi IP del tipo 10.1.10.x;
- ***zona 20*** rappresentata dagli indirizzi IP del tipo 10.1.20.x;
- ***zona 30*** rappresentata dagli indirizzi IP del tipo 10.1.30.x;
- ***zona 60*** rappresentata dagli indirizzi IP del tipo 10.1.60.x;
- ***zona 70*** rappresentata dagli indirizzi IP del tipo 10.1.70.x.

Tutte e cinque le zone sono gestite attraverso i due firewall connessi tra loro in *HA*.

Dal monitoraggio della "sala dati" sono state escluse le seguenti zone: zona 30 e 70 perché dall'azienda viene erogato solo un servizio di hosting verso il cliente che occupa tali zone; zona 60 che rappresenta la parte di rete dedicata a "Banca Etica" che verrà monitorata separatamente.

Oltre a gestire le diverse zone, i firewall gestiscono anche i due switch *Linksys 24 port*e, con supporto SNMP, denominati *SW0* ed *SW1* e collegati tra loro in modalità *HA*.

Sullo "*switch0*" (*SW0*) sono configurate le zone 10 e 20 chiamate anche "*FRONT*" dove sono collegati tutti i server di produzione e dove passa tutto il traffico generato dai servizi erogati, quindi servizi web, ftp, posta ecc.

Sullo "*switch1*" (*SW1*) è configurata una zona denominata "*BACK*" identificata dagli indirizzi IP del tipo 192.168.1.x dove sono collegate tutte le seconde schede di rete dei server di produzione e dove passa tutto il traffico generato tra i server, quindi backup, aggiornamenti interni, copie files ecc. Inoltre, questa zona visto il tipo di utilizzo non dispone di un *gateway* quindi non comunica con l'esterno.

Nella "*zona di produzione*" sono presenti cinque server:

- ***server "ESX1"*** che ospita sette macchine virtuali, delle quali tre dedicate all'hosting (*VM Renovatio*, *VM Carife*, *Greenbox*), tre dedicate ad ospitare domini internet e relativi servizi web (*VM Prd-2*, *VM Prd-3*, *Vm Prd-4*), ed una *(VM Mail)* pronta ad accogliere la virtualizzazione della macchina virtuale "*VM Mail Bis*" attualmente sita su *ESX2*;
- ***server "Acri"*** cliente di cui si gestisce solo la parte di hosting e relativo monitoraggio ;
- ***server "PRD2"*** che è in fase di dismissione e quindi escluso dal monitoraggio;
- ***server "ESX2"*** di cui non è stato richiesto il monitoraggio perché in fase di dimissione con lo scopo di essere virtualizzato sul server *ESX1* attraverso la macchina "*VM Mail*";
- ***server "ESX3"*** che ospita sei macchine virtuali dedicate ai servizi interni della sala dati, delle quali due macchine dedicate alla raccolta dei log provenienti sia dalle applicazione web che dal firewall (*VM WebLog* e *VM Logs*), una macchina adibita al monitoraggio della sala dati (*VM Cacti*), una macchina alle utility (*XP Utility*), una macchina alla gestione del server *ESX3* (*VMA*) ed una dedicata ai backup (*VM Bacula*).

Oltre alle caratteristiche di rete dei singoli dispositivi, nello schema sono state rappresentate le caratteristiche hardware dei server "*ESX*" e delle relative macchine virtuali ospitate su ognuno di essi.

3.1.3. Analisi della rete “Banca Etica”

Per quanto riguarda al rete chiamata “B*anca Etica*” è stata prodotto lo schema presentato in Fig. 3.10.

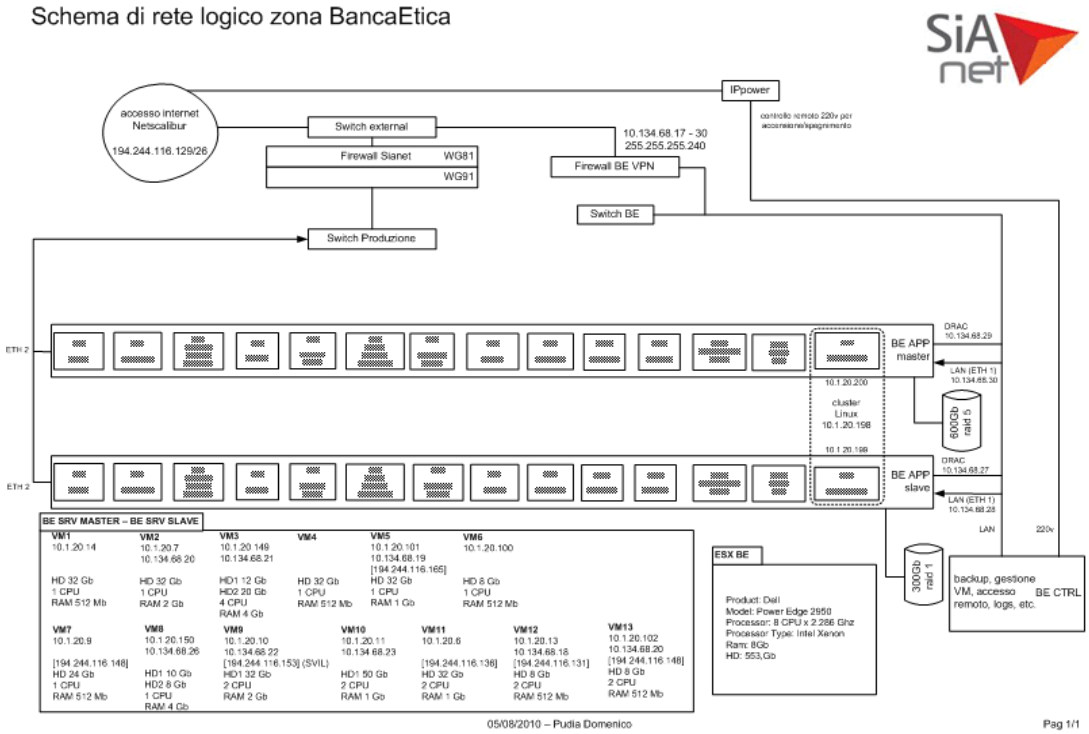


Fig. 3.10 – Schema rete logico "Banca Etica" *(appendice 4)*

La rete “*Banca Etica*” dispone dello stesso *ISP* della rete “*sala dati*”. Questa infrastruttura di rete è stata creata seguendo il modello dell'alta affidabilità (*HA*): sono quindi presenti due server uno principale denominato “*BE APP master*”, e uno secondario denominato “*BE APP slave*”. I due server sono speculari uno all'altro (stesse macchine virtuali ma caratteristiche di rete differenti) per garantire l'alta affidabilità e quindi la continuità dei servizi erogati.

Il sistema *HA* viene gestito tramite l'utilizzo di "*Heartbeat",* uno strumento software del progetto *Linux-HA* che consente di monitorare il funzionamento di uno o più nodi di un "*cluster HA*" (tipologia di cluster progettata per garantire continuità dei servizi informatici erogati) e stabilire comportamenti da adottare in caso di malfunzionamento di un nodo.

In caso di malfunzionamento di una o più macchine, *Heartbeat* provvede a trasferire le risorse dalla "*macchina master*" alla relativa "*macchina slave*" e settare le configurazioni di rete necessarie per rendere operativa la *macchina slave* senza avere interruzioni di servizio.

Le macchine virtuali ospitate sui "*server HA*" insistono su due reti: una rete pubblica da cui proviene il traffico generato dai servizi web (http, https); una rete di gestione dell'infrastruttura stessa di tipo *VPN* (*Virtual Private Network*), cioè una rete privata instaurata tra soggetti che utilizzano un sistema di trasmissione pubblico e condiviso (internet) che necessita di autenticazione per garantire la sicurezza dei dati scambiati su tale rete.

Tra le macchine virtuali è di particolare importanza la macchina chiamata "*VM0*". Questa è un cluster con *filesystem ridondato*, al cui interno è presente un database dove le altre macchine virtuali presenti sul server effettuano lo *storage* dei propri dati. Il filesystem vine ridondato attraverso l'utilizzo di un software chiamo "*DRBD*" (*Distributed Replicated Block Device*) il funzionamento ricorda semanticamente un RAID 1 (*mirror*) tra due dispositivi di rete. L'utilità di questo cluster Linux risiede nel fatto che se una o più *macchine master* vengono spente le relative *macchine slave* posseggono gli stessi "elaborati".

Nello schema sono state riportate anche le relative caratteristiche hardware e di rete sia delle macchine virtuali che dei server che le ospitano.

3.2 Creazione di un modello di massima

Dopo aver analizzato le tre reti si è passato alla creazione di un modello di massima, cioè alla creazione e definizione di uno standard per il monitoraggio dei diversi ambienti .

La necessità di creare un modello di massima nasce con l'obiettivo di minimizzare i tempi e quindi anche i costi di implementazione del sistema di monitoraggio.

E' stato creato un modello per ogni ambiente da monitorare, quindi un modello per la rete “ufficio”, uno per la “sala dati” ed un altro per “Banca Etica”.

Successivamente sono state definite le risorse “standard” da monitorare su tutti i dispositivi connessi in ognuna delle tre reti.

Per quanto riguarda i server con sistema operativo “*VMware ESX*” non è stato possibile impostare un monitoraggio adeguato attraverso Nagios perché le risorse hardware e di rete del server non erano accessibili attraverso questo sistema operativo, anche perché gli sviluppatori *Vmware* hanno implementato un sistema di monitoraggio di tipo proprietario.

Successivamente si è passato a studiare le modalità di notifica degli eventi, quindi tutta la parte di “*alert*” del sistema di monitoraggio.

Oltre alla parte strettamente di monitoraggio si è deciso di corredare il tutto con la creazione di grafici su scala temporale, in modo da avere una visone cronologica sull'andamento sia dei parametri di rete, sia dei parametri hardware di ogni dispositivo connesso alle tre reti analizzate.

3.2.1. Ambienti da monitorare

Il monitoraggio, come già specificato anche in precedenza è stato realizzato per controllare le tre reti: “ufficio”, “sala dati” e “Banca Etica”.

Ognuna delle reti in questione presenta caratteristiche ed esigenze di monitoraggio diverse.

La rete “ufficio” rappresenta un ambiente di sviluppo. In questa rete sono presenti: postazioni di lavoro; stampanti; fax; dispositivi di *storage* (*NAS*); server con le relative macchine virtuali.

Per quanto riguarda le macchine virtuali sono presenti attualmente tredici macchine virtuali operative, di cui dieci con sistema operativo *Microsoft Windows* e le restanti tre con sistemi operativi *Unix-like*. Delle macchine virtuali con sistema operativo *Microsoft Windows Server*, alcune hanno installato al loro interno “*IIS*” (*Internet Information Services*) un complesso di servizi server internet per sistemi operativi *Microsoft Windows*, due rispettivamente con “*SQL server 2005*” e “*SQL server 2008*”, un “*RDBMS*” (*Relational Database Management System*). Altre due hanno installato rispettivamente “*Oracle 9*” ed “*Oracle 11*” altro “*RDBMS*”. Altra è dotata di “*vSphere Management*”, un applicativo volto alla gestione dei server di tipo “*ESX*”, mentre le rimanenti macchine sono state adibite ad altri servizi. Le tre macchine virtuali con sistemi operativi *Unix-like* lato server sono impiegate come segue: una ospita i servizi di *DNS*, *proxy*, backup; una ha installato sia “*MYSQL*” (un “*RDBMS*”) che il web-server Apache; l'ultima ospita il sistema di monitoraggio.

Della rete ufficio vengono monitorate le macchine virtuali operative, le postazioni di lavoro e i diversi dispositivi di rete connessi e redatti i relativi grafici tramite l'applicativo Cacti.

La rete “sala dati” rappresenta come già anticipato l'ambiente di produzione dell'azienda, infatti è caratterizzata dalla presenza esclusiva di server, macchine virtuali e dispositivi di rete come firewall e switch. In questo ambiente è stato di vitale importanza implementare un sistema di monitoraggio capace di controllare in tempo reale la banda utilizzata dai vari servizi ospitati nelle diverse “zone” per prevenire situazioni di saturazione, causando così fenomeni di latenza sugli altri servizi. Il sistema di monitoraggio è stato implementato per tenere sotto controllo, oltre ai servizi di rete, tutti i parametri hardware sia dei server che delle macchine virtuali presenti in rete, per garantire in caso di malfunzionamenti una rapida notifica e un successivo intervento senza pregiudicare l'erogazione dei servizi e l'operatività delle varie macchine. Le macchine virtuali presenti in questa rete si possono suddividere in due categorie: le macchine che erogano servizi, ospitate su “*ESX1*”, dedicate all'hosting e ai servizi web; le macchine di gestione, ospitate su “*ESX3*”, dedicate ai servizi interni della sala dati, tra cui due macchine dedicate alla raccolta dei “*log*” provenienti sia dalle applicazioni web che dal firewall, una macchina adibita al monitoraggio della sala dati, una macchina per le “*utility*”, una macchina alla gestione del server *ESX3* ed una dedicata ai *backup*. Insieme al monitoraggio della rete vengono creati i relativi grafici.

La terza rete che è stata monitorata è quella chiamata “Banca Etica” dal nome del cliente a cui si presta il servizio di hosting, di gestione della rete e dei servizi web. Questa rete, come già detto, è stata creata secondo i protocolli dell'alta affidabilità. La parte di maggior rilevanza che è stata oggetto del monitoraggio sono i due server denominati rispettivamente “*BE APP master*” (principale) e “*BE APP slave*” (secondario).

Questi due server sono speculari tra loro, possedendo le stesse macchine virtuali ma con caratteristiche di rete differenti (indirizzi IP differenti). Per questa rete è stata monitorata tutta la banda impegnata e i due server con le relative macchine virtuali presenti su ognuno di esso; il tutto è stato correlato dalla generazione di grafici con l'ausilio di Cacti.

3.2.2. Risorse da monitorare

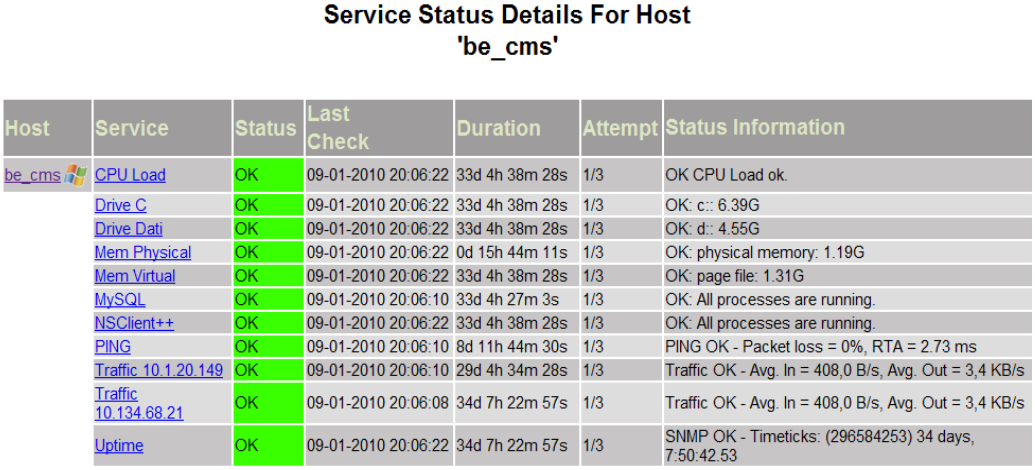
Per monitorare lo stato di una risorsa o un servizio attraverso Nagios è necessario l'utilizzo di un plugin specifico per la risorsa o servizio che si intende monitorare. Nagios dispone di una serie di plugin di base che permettono di controllare sia i servizi di rete più comuni che la parte hardware della macchina che ospita il sistema di monitoraggio. Grazie alla sua flessibilità Nagios permette di eseguire una miriade di plugin; essendo anche un software open-source molto utilizzato è possibile trovare delle community dove si possono scaricare e condividere liberamente numerosi plugin. Una delle community più famose è "*MonitoringEXCHANGE*", da cui si è attinto per la ricerca di plugin e informazioni necessarie alla realizzazione di questo sistema di monitoraggio.

Per ogni macchina virtuale o fisica si sono stabilite delle risorse "standard" da monitorare attraverso l'uso di specifici plugin, sia per quelle con sistema operativo *Windows* sia per quelle con sistemi operativi *Unix-like* (**nix*).

Per le macchine *Windows*, come già detto, si è fatto uso del client *NSClient++* per svolgere l'attività di monitoraggio. Questo software dispone già al momento della sua installazione sulla *macchina client* di una libreria di plugin standard di cui sono stati usati quelli per monitorare le risorse hardware.

Di seguito i plugin utilizzati per il monitoraggio hardware e di rete:

- ***CheckDriveSize*** controlla lo spazio disponibile su un disco generando un allarme in caso di superamento delle soglie di *warning* (10% spazio libero) o *critical* (5% spazio libero);
- ***Checkcpu*** calcola una media di utilizzo della CPU per un determinato periodo di tempo "*t*" (30s) generando un allarme in caso di superamento delle soglie di *warning* (85% utilizzo) o *critical* (90% utilizzo);
- ***CheckMEM*** controlla l'utilizzo sia della memoria fisica (*type=phisical*) che virtuale (*type=virtual*) generando un allarme in caso di superamento delle soglie di *warning* (80% spazio utilizzato) o *critical* (90% spazio utilizzato);
- ***CheckProcState*** controlla lo stato di uno o più processi del sistema e genera uno stato *critical* se qualsiasi processo non è nello stato richiesto (*procstate=started/stopped*);
- ***check_ping*** calcola attraverso l'uso del comando "*ping*" la percentuale di pacchetti persi (*pl*) o la media del round trip time (*RTT*) generando un allarme in caso di superamento delle soglie di *warning* (RTT=60ms o pl=60%) o *critical* (RTT=70ms o pl=80%);
- ***check_mrtg_traffic*** calcola attraverso i log generati da *MRTG* una media dell'utilizzo di banda su una determinata scheda di rete, generando un allarme in caso di superamento delle soglie di warning (300 kb/s) o critical (500 kb/s);
- ***check_snmp_uptime*** calcola attraverso l'utilizzo del protocollo SNMP da quanto tempo la macchina non viene riavviata.



Service Status Details For Host 'be_cms'

Host	Service	Status	Last Check	Duration	Attempt	Status Information
be_cms	CPU Load	OK	09-01-2010 20:06:22	33d 4h 38m 28s	1/3	OK CPU Load ok.
	Drive C	OK	09-01-2010 20:06:22	33d 4h 38m 28s	1/3	OK: c:: 6.39G
	Drive Dati	OK	09-01-2010 20:06:22	33d 4h 38m 28s	1/3	OK: d:: 4.55G
	Mem Physical	OK	09-01-2010 20:06:22	0d 15h 44m 11s	1/3	OK: physical memory: 1.19G
	Mem Virtual	OK	09-01-2010 20:06:22	33d 4h 38m 28s	1/3	OK: page file: 1.31G
	MySQL	OK	09-01-2010 20:06:10	33d 4h 27m 3s	1/3	OK: All processes are running.
	NSClient++	OK	09-01-2010 20:06:22	33d 4h 38m 28s	1/3	OK: All processes are running.
	PING	OK	09-01-2010 20:06:10	8d 11h 44m 30s	1/3	PING OK - Packet loss = 0%, RTA = 2.73 ms
	Traffic 10.1.20.149	OK	09-01-2010 20:06:10	29d 4h 34m 28s	1/3	Traffic OK - Avg. In = 408,0 B/s, Avg. Out = 3,4 KB/s
	Traffic 10.134.68.21	OK	09-01-2010 20:06:08	34d 7h 22m 57s	1/3	Traffic OK - Avg. In = 408,0 B/s, Avg. Out = 3,4 KB/s
	Uptime	OK	09-01-2010 20:06:22	34d 7h 22m 57s	1/3	SNMP OK - Timeticks: (296584253) 34 days, 7:50:42.53

Fig. 3.11 – Servizi e risorse monitorate su un host Windows

Per le macchine con sistema operativo ***nix*, si è fatto uso del client *NRPE* per svolgere l'attività di monitoraggio. Questo software dispone già al momento della sua installazione sulla *macchina client* della stessa libreria di plugin standard della macchina Nagios. Di seguito i plugin utilizzati per il monitoraggio hardware e di rete:

- ***check_disk*** controlla lo spazio disponibile su un disco generando un allarme in caso di superamento delle soglie di *warning* (10% spazio libero) o *critical* (5% spazio libero);
- ***check_swap*** controlla lo spazio disponibile sul disco "*swap*" generando un allarme in caso di superamento delle soglie di *warning* (10% spazio libero) o *critical* (5% spazio libero).
- ***check_load*** calcola una media di utilizzo della CPU per un determinato periodo di tempo "*t*" (30s) generando un allarme in caso di superamento delle soglie di *warning* (85% utilizzo) o *critical* (90% utilizzo);
- ***check_ram*** controlla l'utilizzo della memoria fisica generando un allarme in caso di superamento delle soglie di *warning* (80% spazio utilizzato) o *critical* (90% spazio utilizzato);
- ***check_procs*** controlla il numero di processi attivi sulla macchina generando un allarme in caso di superamento delle soglie di *warning* (90 processi attivi) o *critical* (110 processi attivi);
- ***check_users*** controlla il numero degli utenti collegati sulla macchina generando un allarme in caso di superamento delle soglie di *warning* (5 utenti) o *critical* (10 utenti);
- ***check_ssh*** prova a connettersi ad un server SSH attraverso un indirizzo e una porta specificati generando un allarme di tipo *critical* in caso di connessione fallita;
- ***check_ping*** stesso plugin utilizzato per le macchine Windows*;*
- ***check_mrtg_traffic*** stesso plugin utilizzato per le macchine Windows;
- ***check_snmp_uptime*** stesso plugin utilizzato per le macchine Windows.

Service Status Details For Host 'be_nagios'

Host	Service	Status	Last Check	Duration	Attempt	Status Information
be_nagios	Current Load	OK	09-01-2010 20:11:46	34d 20h 49m 30s	1/3	OK - load average: 0.35, 0.42, 0.34
	Current Users	OK	09-01-2010 20:11:45	34d 20h 49m 30s	1/3	USERS OK - 1 users currently logged in
	PING	OK	09-01-2010 20:11:21	34d 20h 49m 30s	1/3	PING OK - Packet loss = 0%, RTA = 0.04 ms
	PROCS	OK	09-01-2010 20:10:39	34d 20h 49m 30s	1/3	PROCS OK: 79 processes with STATE = RSZDT
	RAM	OK	09-01-2010 20:11:34	34d 20h 49m 30s	1/3	RAM OK: 330MB ram free
	SSH	OK	09-01-2010 20:10:39	34d 20h 49m 30s	1/3	SSH OK - OpenSSH_5.3p1 Debian-3ubuntu4 (protocol 2.0)
	Swap Usage	OK	09-01-2010 20:10:54	34d 20h 49m 30s	1/3	SWAP OK - 92% free (356 MB out of 387 MB)
	Traffic 10.1.20.102	OK	09-01-2010 20:11:06	34d 10h 49m 0s	1/3	Traffic OK - Avg. In = 718,0 B/s, Avg. Out = 597,0 B/s
	Traffic 10.134.68.20	OK	09-01-2010 20:11:47	34d 8h 22m 57s	1/3	Traffic OK - Avg. In = 720,0 B/s, Avg. Out = 599,0 B/s
	Uptime	OK	09-01-2010 20:12:10	34d 20h 49m 30s	1/3	SNMP OK - Timeticks: (304075555) 35 days, 4:39:15.55

Fig. 3.12 – Servizi e risorse monitorate su un host Ubuntu

Per quanto riguarda i server con sistema operativo *VMware ESX* non è stato possibile monitorare in maniera soddisfacente con Nagios le risorse hardware e di rete, ma ci si è successivamente avvalsi di un applicativo specifico chiamato "*VSphere server*". Il monitoraggio tramite l'applicativo *VSphere server* dei server con sistema operativo *VMware ESX* non è stato affrontato durante lo svolgimento dello stage. L'unico servizio che è stato monitorato con Nagios è il ping sul server e sulla scheda *DRAC*, così da verificare verificare la raggiungibilità del server.

Service Status Details For Host 'be_master'

Host	Service	Status	Last Check	Duration	Attempt	Status Information
be_master	DRAC	OK	09-02-2010 12:20:10	33d 20h 11m 6s	1/3	PING OK - Packet loss = 0%, RTA = 0.98 ms
	PING	OK	09-02-2010 12:20:10	33d 20h 12m 37s	1/3	PING OK - Packet loss = 0%, RTA = 0.58 ms

Fig. 3.13 – Servizi monitorati su server "ESX"

Per le stampanti connesse in rete sono stati utilizzati i seguenti plugin:

- ***check_ping*** stesso plugin utilizzato per le macchine Windows*;*
- ***check_pagecount*** visualizza il "*pagecount*" della stampante;
- ***check_paper*** genera un allarme di tipo *critical* quando il vassoio principale o il secondario sono vuoti;
- ***check_model*** visualizza marca e modello della stampante e versione firmware.

Service Status Details For Host 'canon'

Host	Service	Status	Last Check	Duration	Attempt	Status Information
canon	Model	OK	09-03-2010 12:25:38	29d 22h 32m 43s	1/3	Canon MF4360-4390
	PING	OK	09-03-2010 12:21:39	0d 1h 29m 51s	1/3	PING OK - Packet loss = 0%, RTA = 1.91 ms
	Pagecount	OK	09-03-2010 12:29:28	29d 22h 31m 46s	1/3	OK Pagecount is 4825
	Uptime	OK	09-03-2010 12:30:53	0d 4h 25m 37s	1/3	SNMP OK - Timeticks: (1592468) 4:25:24.68
	Vassoio Principale	OK	09-03-2010 12:27:40	0d 4h 23m 50s	1/3	TRAY2 is OK

Fig. 3.14 – Servizi monitorati su una stampante

I plugin appena illustrati sono stati utilizzati su stampanti con supporto SNMP, mentre su quelle senza tale supporto è stato possibile controllare solo il "*ping*" necessario per verificare la raggiungibilità del dispositivo all'interno della rete.

Per i dispositivi come switch, firewall e access point con supporto SNMP sono stati utilizzati i seguenti plugin:

- ***check_ping*** stesso plugin utilizzato per le macchine Windows*;*
- ***check_mrtg_traffic*** stesso plugin utilizzato per le macchine Windows*;*
- ***check_snmp_uptime*** stesso plugin utilizzato per le macchine Windows*;*

Per quanto riguarda gli switch senza il supporto SNMP è stato possibile monitorare solo il "*ping*" necessario per verificare la latenza e la raggiungibilità del dispositivo.

Host	Service	Status	Last Check	Duration	Attempt	Status Information
sw1	PING	CRITICAL	09-03-2010 15:34:14	0d 0h 0m 55s	1/3	CRITICAL - Host Unreachable (192.168.1.252)
	Traffic ch1	OK	08-24-2010 17:19:37	11d 0h 15m 32s	1/3	Traffic OK - Avg. In = 188,0 KB/s, Avg. Out = 26,9 KB/s
	Traffic g1	OK	08-24-2010 17:18:39	38d 20h 46m 30s	1/3	Traffic OK - Avg. In = 16,9 KB/s, Avg. Out = 174,3 KB/s
	Traffic g10	OK	08-24-2010 17:19:41	38d 20h 45m 28s	1/3	Traffic OK - Avg. In = 0,0 B/s, Avg. Out = 56,0 B/s
	Traffic g11	OK	08-24-2010 17:21:49	11d 0h 23m 20s	1/3	Traffic OK - Avg. In = 41,9 KB/s, Avg. Out = 21,5 KB/s
	Traffic g12	WARNING	09-03-2010 15:34:09	0d 0h 5m 0s	3/3	MRTG data has expired (14399 minutes old)
	Traffic g13	WARNING	09-03-2010 15:33:11	0d 0h 3m 58s	2/3	MRTG data has expired (14398 minutes old)
	Traffic g15	WARNING	09-03-2010 15:34:13	0d 0h 2m 56s	2/3	MRTG data has expired (14399 minutes old)
	Traffic g18	WARNING	09-03-2010 15:33:15	0d 0h 1m 54s	1/3	MRTG data has expired (14398 minutes old)
	Traffic g2	OK	09-03-2010 15:34:17	38d 20h 48m 31s	1/3	Traffic OK - Avg. In = 6.9 KB/s, Avg. Out = 39.9 KB/s
	Traffic g20	OK	08-24-2010 17:16:52	12d 11h 18m 17s	1/3	Traffic OK - Avg. In = 45,0 B/s, Avg. Out = 112,0 B/s
	Traffic g22	OK	08-24-2010 17:20:42	15d 23h 34m 27s	1/3	Traffic OK - Avg. In = 0,0 B/s, Avg. Out = 78,0 B/s
	Traffic g24	OK	08-24-2010 17:17:44	10d 10h 27m 25s	1/3	Traffic OK - Avg. In = 119,0 B/s, Avg. Out = 229,0 B/s
	Traffic g3	OK	08-24-2010 17:20:52	38d 20h 44m 22s	1/3	Traffic OK - Avg. In = 0,0 B/s, Avg. Out = 32,0 B/s
	Traffic g4	OK	09-03-2010 15:32:12	0d 0h 2m 57s	1/3	Traffic OK - Avg. In = 1.2 KB/s, Avg. Out = 589.0 B/s
	Traffic g5	OK	09-03-2010 15:31:14	11d 1h 24m 25s	1/3	Traffic OK - Avg. In = 38.0 B/s, Avg. Out = 82.0 B/s
	Traffic g6	OK	09-03-2010 15:32:16	38d 20h 50m 32s	1/3	Traffic OK - Avg. In = 0.0 B/s, Avg. Out = 64.0 B/s
	Traffic g7	OK	09-03-2010 15:33:18	38d 20h 49m 30s	1/3	Traffic OK - Avg. In = 8.0 B/s, Avg. Out = 443.0 B/s
	Traffic g8	WARNING	09-03-2010 15:34:20	0d 0h 0m 49s	1/3	MRTG data has expired (14399 minutes old)
	Uptime	OK	08-24-2010 17:19:51	38d 20h 47m 26s	1/3	SNMP OK - Timeticks: (79754352) 9 days 5:32:23.52

Fig. 3.15 – Servizi monitorati su uno switch

3.2.3. Notifiche degli eventi

Un buon sistema di monitoraggio, oltre a tenere sotto controllo una rete insieme ai dispositivi connessi, per essere efficiente deve anche poter notificare eventuali problemi.

Le notifiche attualmente vengono inviate tramite mail. E' stato previsto anche l'invio di SMS; il relativo modulo è stato implementato ma non ancora attivato per una politica interna aziendale. Oltre ai metodi sopracitati è possibile notificare gli eventi in diversi modi come dice anche la guida ufficiale Nagios: *<< You can have Nagios notify you of problems and recoveries pretty much anyway you want: cellphone, e-mail, instant message, audio alert, electric shocker, etc.* **>>**, che tradotto significa *<< E' possibile ricevere notifiche Nagios su problemi e "recovery" in qualsiasi modo si desideri: cellulare, e-mail, messaggi istantanei, allarmi audio, scossa elettrica, etc.>>*. Da ciò si capisce ancora una volta come Nagios sia un software molto versatile o come si dice in gergo "*customizzabile*".

Oltre a decidere tramite quali mezzi notificare gli eventi, è stata decisa la politica in base alla quale un evento deve essere notificato.

Viene inoltrata una notifica quando un host o un servizio:

- ***cambia stato***, generando quindi un allarme di tipo *warning* o *critical* in base alle soglie impostate sui relativi plugin;
- ***ritorna allo stato di "OK"***, cioè ritorna nei valori si soglia impostati nel relativo plugin **(**questa viene chiamata notifica di "*recovery*");
- ***persevera nello stato "non-OK"***, quindi viene mandata una notifica ogni intervallo di tempo specificato nel "*notifcation interval*" di quel determinato servizio/host monitorato.
-

Per le notifiche degli host è stato configurato un comando Nagios chiamato "*notify-host-by-email*" così costituito:

```
# 'notify-host-by-email' command definition
define command{
        command_name    notify-host-by-email
        command_line    /usr/bin/php -f /$USER$/mailsender_host.php "Notification Type:
                        $NOTIFICATIONTYPE$" "Host: $HOSTNAME$" "State: $HOSTSTATE$"
                }
```

Fig. 3.16 – Comando "notify-host-by-email"

Quando viene invocato questo comando il sistema Nagios invia una e-mail agli amministratori della rete tramite lo script "*mailsender_host.php*".

Inoltre si può personalizzare il sistema di notifica in modo da mandare una e-mail quando l'host è in un particolare stato, impostando i seguenti parametri:

- ***-d*** host "*down*" (non risponde);
- ***-u*** host "*unreachable*" (irraggiungibile);
- ***-r*** "*recovery*" l'host ritorna nello stato "*OK*" dopo essere stato "*down*" o "*unreachable*".

Per le notifiche dei servizi è stato configurato un comando Nagios chiamato "*notify-service-by-email*" così costituito:

```
# 'notify-service-by-email' command definition
define command{
        command_name    notify-service-by-email
      command_line      /usr/bin/php -f /$USER1$/mailsender_service.php "Notification Type:
                        $NOTIFICATIONTYPE$" "Service: $SERVICEDESC$" "Host:$HOSTALIAS$"
                    }
```

Fig. 3.17 – Comando "notify-service-by-email"

Quando viene invocato questo comando il sistema Nagios invia una e-mail agli amministratori della rete tramite lo script "*mailsender_service.php*".

Inoltre si può personalizzare il sistema di notifica in modo da mandare una e-mail quando il servizio è in un particolare stato, impostando i seguenti parametri:

- ***-w*** il servizio è nello stato "*warning*";
- ***-u*** il servizio è nello stato "*unreachable*";
- ***-c*** il servizio è nello stato "*critical*";
- ***-r*** (*recovery*) il servizio ritorna nello stato "*OK*" dopo essere stato "*critical*" o "*unreachable*".

E' stato creato l'indirizzo mail alert@sianet.biz dove il sistema di notifica di Nagios invia le e-mail.

Di seguito una e-mail di *alert* inviata dal sistema di notifica:

Subject: SALA DATI – Host: vm_renovatio Service: RAM State: WARNING
From: Nagios
To: alert@sianet.biz
Reply-To: alert@sianet.biz
Date: Sat,4 Sep 2010 07:16:55

NotificationType: PROBLEM
Service: RAM
Host: vm_renovatio
Address: 10.1.10.3
State: WARNING
Date/Time: Sat Sept 4 09:16:55 CEST 2010
Additional Info: RAM WARNING: 18% ram free (2766/3024 MB used)

Fig. 3.18 – E-mail di notifica

Per quanto riguarda le notifiche tramite SMS sono stati creati degli appositi comandi che hanno una struttura simile ai comandi di notifica precedentemente illustrati.

Per le notifiche degli host è stato configurato un comando Nagios chiamato "*notify-host-by-sms*" così costituito:

```
# 'notify-host-by-sms' command definition
define command{
        command_name    notify-host-by-sms
        command_line    /usr/bin/php -f /$USER$/sms_sender_host.php "Notification Type:
                        $NOTIFICATIONTYPE$" "Host: $HOSTNAME$" "State: $HOSTSTATE$"
                }
```

Fig. 3.19 – Comando "notify-host-by-sms"

Quando viene invocato questo comando il sistema Nagios invia un SMS agli amministratori della rete tramite lo script "sms*_sender_host.php*".

Anche per questo comando valgono le opzioni di notifica precedentemente illustrate per il comando *notify-host-by-email.*

Per le notifiche dei servizi è stato configurato un comando Nagios chiamato "*notify-service-by-sms*" così costituito:

```
# 'notify-service-by-sms' command definition
define command{
        command_name    notify-service-by-sms
       command_line       /usr/bin/php -f /$USER1$/sms_sender_service.php "Notification Type:
                          $NOTIFICATIONTYPE$" "Service: $SERVICEDESC$" "Host:$HOSTALIAS$"
                       }
```

Fig. 3.20 – Comando "notify-service-by-sms"

Quando viene invocato questo comando il sistema Nagios invia un SMS agli amministratori della rete tramite lo script *"sms_sender_service.php"*.

Anche per questo comando valgono le opzioni di notifica precedentemente illustrate per il comando *notify-service-by-email.*

3.2.4. Creazione dei grafici

Come già anticipato, oltre alla parte strettamente di monitoraggio sono stati creati dei grafici attraverso l'uso dell'applicativo Cacti.

Con questa applicazione sono stati generati grafici per ogni servizio e host monitorato attraverso Nagios.

Cacti per la generazione dei grafici utilizza dei *template* facilmente reperibili sul sito ufficiale www.cacti.net. Inoltre, per interrogare i dispositivi di rete e popolare il database utilizza il protocollo SNMP.

Per una maggiore semplicità di consultazione i grafici dei servizi e degli host monitorati sono stati raggruppati utilizzando un modello ad albero. Di seguito uno "*screen-shot*" proveniente dalla rete "*sala dati*".

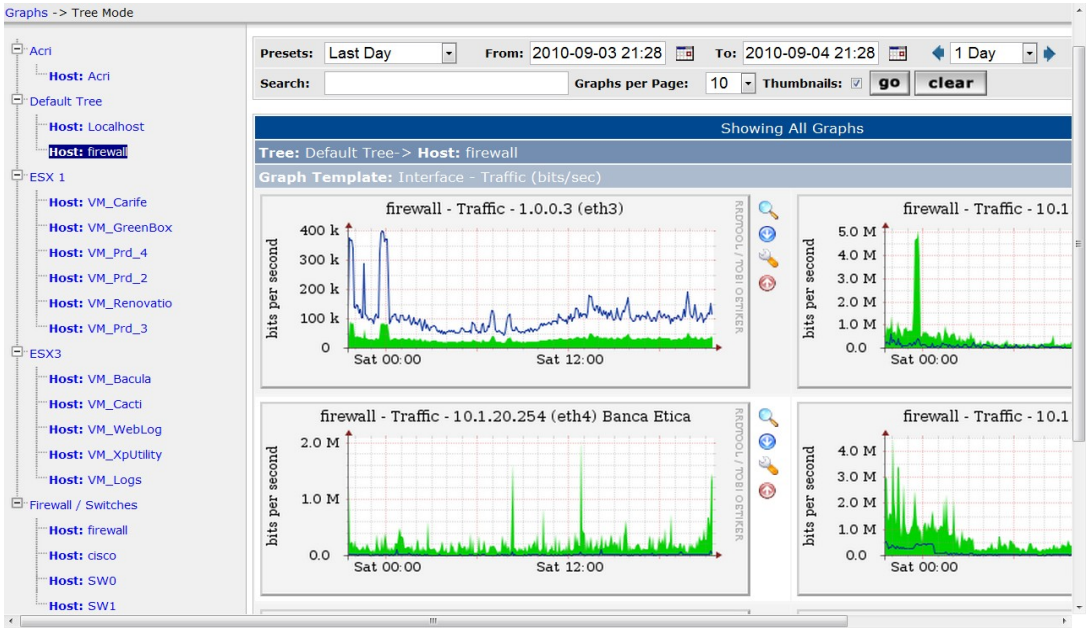


Fig. 3.21 – Screen-shot Cacti "sala dati"

Per ogni servizio monitorato vengono generati quattro grafici con i seguenti *range* temporali:

- ***Daily*** grafico con visione giornaliera calcolato sulla media dei dati ogni 5 minuti;
- ***Weekly*** grafico con visione settimanale calcolato sulla media dei dati ogni 30 minuti;
- ***Monthly*** grafico con visione mensile calcolato sulla media dei dati ogni 2 ore;
- ***Yearly*** grafico con visione annuale calcolato sulla media dei dati ogni giorno;

Su ogni grafico è possibile usare la funzione zoom per inquadrare i particolari oppure andare ad analizzare un preciso arco temporale con la funzione cronologia.

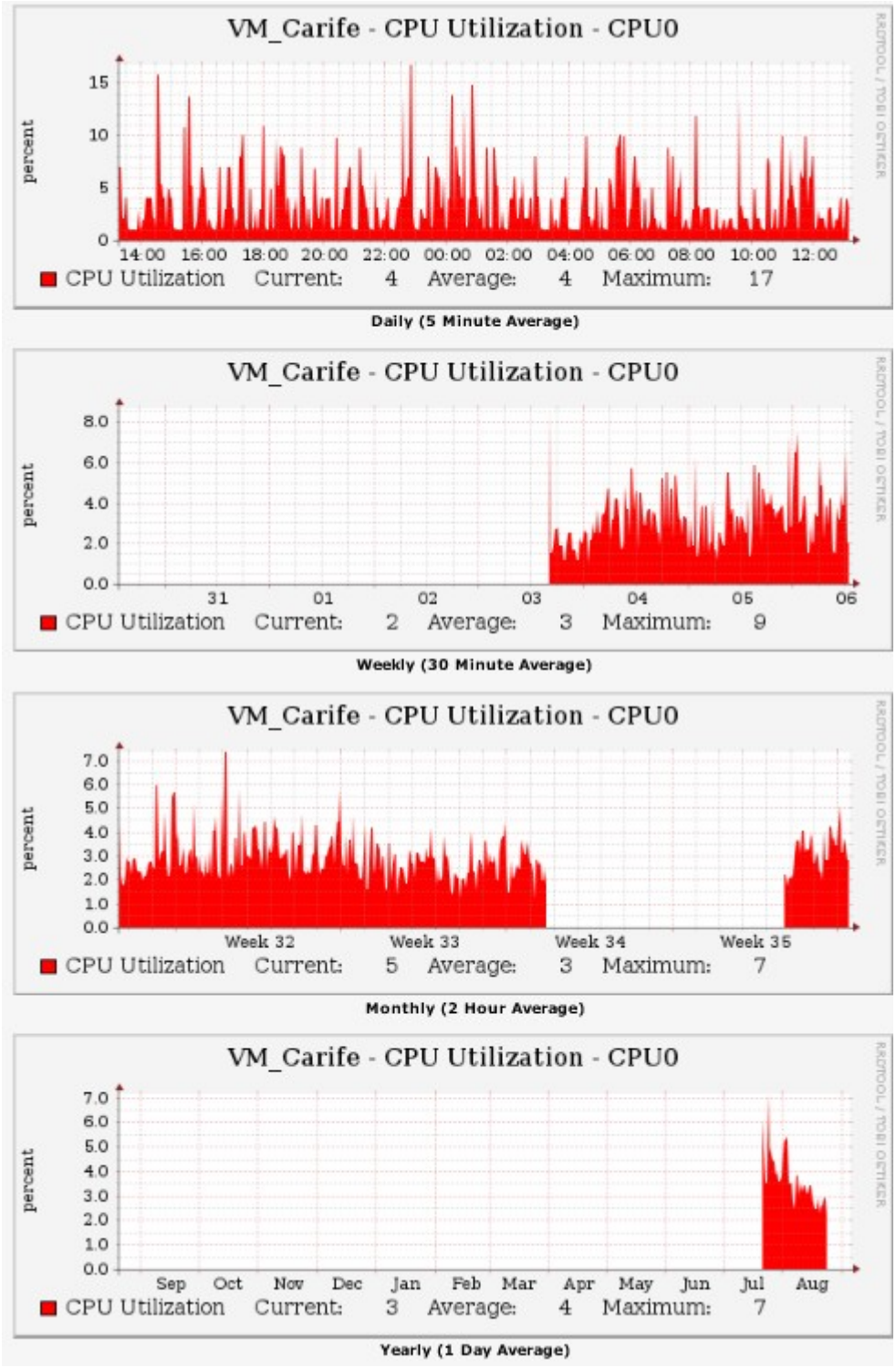


Fig. 3.22 – Grafici Cacti per il servizio CPU Utilization

Capitolo 4

IMPLEMENTAZIONE

Dopo aver studiato un modello di massima si è passati all'implementazione del sistema di monitoraggio.

Prima di passare alla descrizione dell'implementazione delle tre reti oggetto del monitoraggio è opportuno fare una panoramica sulle parti che si andranno a realizzare. Per quanto riguarda Nagios, i file di configurazione si trovano tutti nel seguente path: */usr/local/nagios/etc*.

Il più importante file di configurazione Nagios è quello chiamato "*nagios.cfg*" che contiene tutte le direttive e le opzioni necessarie al funzionamento dell'applicativo in questione.

Altro file di configurazione importante è quello chiamato "*cgi.cfg*" che gestisce tutta la parte dell'interfaccia web e gli accessi dei vari utenti alla stessa.

Per quanto riguarda le configurazioni degli host e dei servizi, questi utilizzano dei "*template*" nei loro file di configurazione, nei quali sono contenute diverse funzioni ed opzioni.

Tutti i template sono contenuti in un unico file chiamato "*template.cfg*" che si trova nel seguente path: */usr/local/nagios/etc/objects*.

I file di configurazione degli host si basano sull'utilizzo di un template chiamato "*generic-host*" il cui contenuto è il seguente:

```
# Generic host definition template - This is NOT a real host, just a template!

define host{
        name                            generic-host    ; The name of this host template
        notifications_enabled           1               ; Host notifications are enabled
        event_handler_enabled           1               ; Host event handler is enabled
        flap_detection_enabled          1               ; Flap detection is enabled
        failure_prediction_enabled      1               ; Failure prediction is enabled
        process_perf_data               1               ; Process performance data
        retain_status_information       1               ; Retain status information across program restarts
        retain_nonstatus_information    1               ; Retain non-status information across program restarts
        notification_period             24x7            ; Send host notifications at any time
        register                        0               ; DONT REGISTER THIS DEFINITION - ITS NOT
                                                          A REAL HOST, JUST A TEMPLATE

        }
```

Fig. 4.1 – Template "generic-host"

si può notare come il template sia formato da una serie di opzioni. Da notare l'opzione chiamata "*flap_detenction_enable*" che posta a "*1*" risulta essere abilitata: questa opzione disabilita le notifiche in caso di un continuo cambio di stato dell'host monitorato (valore di default 20%). Cosa importante che distingue i template è l'opzione "*register*" che è posta a "*0*", citando il relativo commento: << *DONT REGISTER THIS DEFINITION - ITS NOT A REAL HOST, JUST A TEMPLATE* >>, che tradotto significa: << *Non registrare questa definizione – Non è un host reale, ma solo un template* >>.

Dopo aver visto come è costituito il template *generic-host* è più facile comprendere il file di configurazione dell'host reale "*vm_cacti*" riportato in seguito:

```
define host{
        use                     generic-host            ; Name of host template to use
        host_name               vm_cacti
        alias                   VM Cacti
        address                 10.1.10.4
        icon_image              ubuntu2.png
        parents                 esx3
        statusmap_image         ubuntu2.gd2

}
```

Fig. 4.2 – File di configurazione host "vm_cacti"

l'host in questione usa il template precedentemente illustrato.

Notare la voce "*parents*" che identifica il dispositivo a cui l'host è collegato: tramite questa funzione Nagios disegna automaticamente la mappa di rete. Per quanto riguarda i servizi monitorati il funzionamento è analogo agli host, cioè ogni servizio utilizza un template. Di seguito il template chiamato "*generic-service*":

```
# Generic service definition template - This is NOT a real service, just a template!

define service{
        name                            generic-service         ; The 'name' of this service template
        active_checks_enabled           1                       ; Active service checks are enabled
        passive_checks_enabled          1                       ; Passive service checks are enabled/accepted
        parallelize_check               1                       ; Active service checks should be parallelized
        obsess_over_service             1                       ; We should obsess over this service
        check_freshness                 0                       ; Default is to NOT check service 'freshness'
        notifications_enabled           1                       ; Service notifications are enabled
        event_handler_enabled           1                       ; Service event handler is enabled
        flap_detection_enabled          1                       ; Flap detection is enabled
        failure_prediction_enabled      1                       ; Failure prediction is enabled
        process_perf_data               1                       ; Process performance data
        retain_status_information       1                       ; Retain status information across
                                                                  program restarts
        retain_nonstatus_information    1                       ; Retain non-status information across
                                                                  program restarts
        is_volatile                     0                       ; The service is not volatile
        check_period                    24x7                    ; The service can be checked at any time of the day
        max_check_attempts              3                       ; Re-check the service up to 3 times in
                                                                  order to determine its final (hard) state
        normal_check_interval           10                      ; Check the service every 10 minutes
                                                                  under normal conditions
        retry_check_interval            2                       ; Re-check the service every two minutes
                                                                  until a hard state can be determined
        contact_groups                  admins                  ; Notifications get sent out to everyone
                                                                  in the 'admins' group
        notification_options            w,u,c,r                 ; Send notifications about warning,
                                                                  unknown, critical, and recovery events
        notification_interval           0                       ; Re-notify about service problems every 0
        notification_period             24x7                    ; Notifications can be sent out at any time
        register                        0                       ; DONT REGISTER THIS DEFINITION -
                                                                ITS NOT A REAL SERVICE, JUST A TEMPLATE!
        }
```

Fig. 4.3 – Template "generic-service"

In questo template comune a tutti i servizi che si andranno ad implementare si può notare la presenza sia degli "*active_checks_enable*" sia dei "*passive_checks_enable*" entrambi abilitati perché posti ad "*1*". La differenza tra "*active checks*" e "*passive checks*" sta nel fatto che i primi sono controlli che esegue direttamente Nagios, gli altri sono controlli effettuati da

applicazioni esterne, ad esempio uno script che è in esecuzione su un host ed invia durante il suo funzionamento i risultati a Nagios. Nel template sono presenti anche le funzioni "*normal_check_interval*" e "*max_check_attempts*": la prima impostata a "*10*", indica che Nagios effettuerà i *check* ogni dieci minuti, mentre la seconda, impostata a "*3*", indica che solo dopo tre check con egual esito il sistema provvederà ad inviare la relativa notifica, così da evitare falsi allarmi. Di seguito è riportata la configurazione del servizio chiamato "*Current Users*" della macchina *vm_cacti*:

```
define service{
        use                             generic-service         ; Name of service template to use
        host_name                       vm_cacti
        service_description             Current Users
        check_command                   check_users!5!10
        }
```

Fig. 4.4 – Servizio Current Users

si può notare come il servizio estenda il template *generic-service* e i relativi valori di soglia del plugin *"check_users"* sono posti a "*5*" per lo stato di *warning* e a "*10*" per quello di *critical*.

```
# Linux host definition template

define host{
        name                    linux-server    ; The name of this host template
        use                     generic-host    ; This template inherits other values from the generic-host
        check_period            24x7            ; By default, Linux hosts are checked round the clock
        check_interval          2               ; Actively check the host every 2 minutes
        retry_interval          1               ; Schedule host check retries at 1 minute intervals
        max_check_attempts      3               ; Check each Linux host 3 times (max)
        check_command           check-host-alive ; Default command to check Linux hosts
        notification_period     24x7            ; Send notification out at any time - day or night
        notification_interval   0               ; Resend notifications every 0 minutes
        notification_options    d,u,r           ; Only send notifications for specific host states
        contact_groups          admins          ; Notifications get sent to the admins by default
        register                0
        }
```

Fig. 4.5 – Template "linux-server"

E' da precisare che è possibile creare ulteriori template che estendano al loro interno altri template; in Fig 4.5 un esempio:

il template "*linux-server*" estende il template *generic-host*.

Per quel che riguarda Cacti di seguito un esempio di configurazione di un "*device*" per la generazione di grafici per i servizi monitorati:

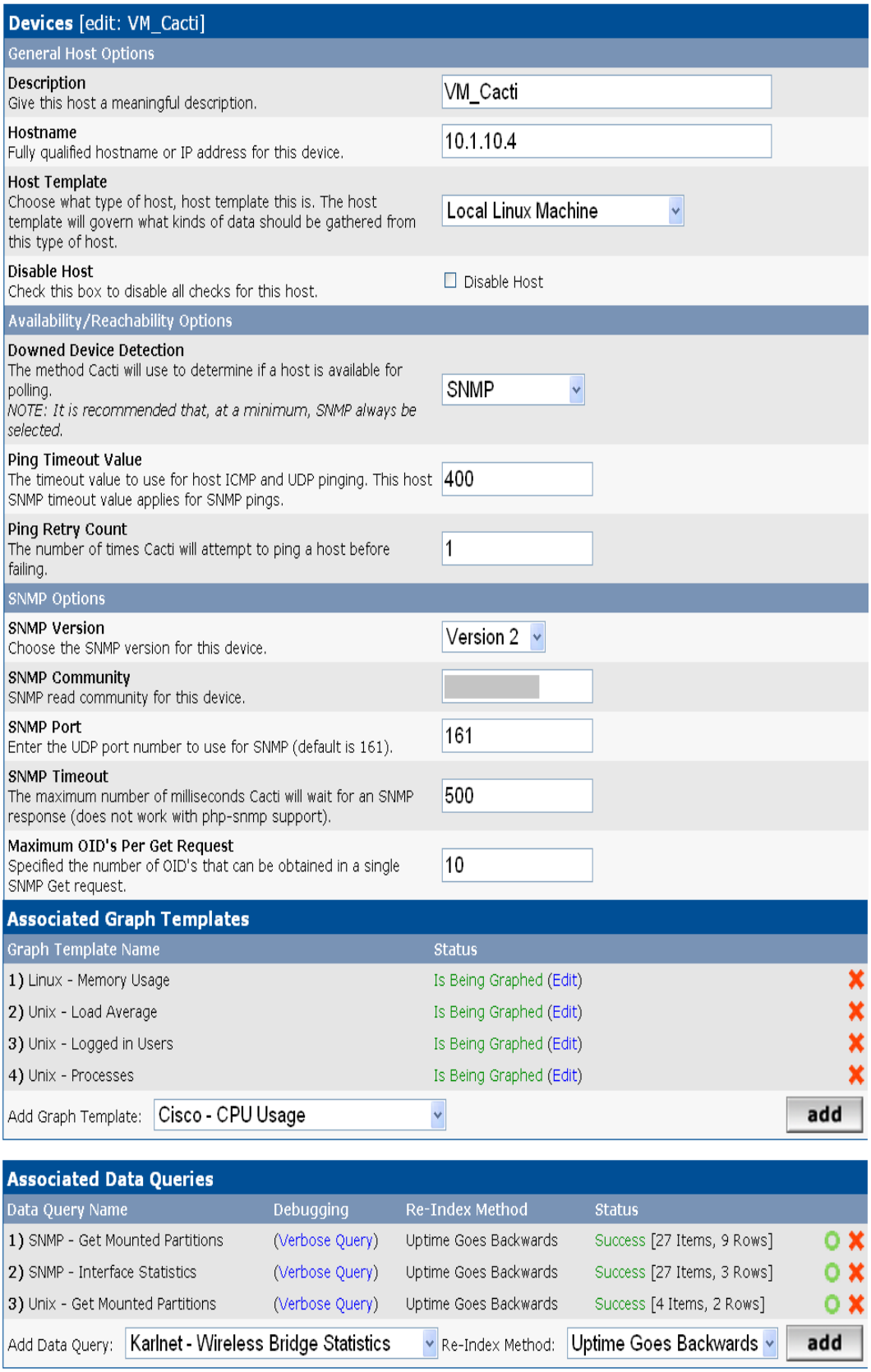


Fig. 4.6 – Configurazione dell' host "Vm_Cacti" attraverso interfaccia web

Come si può notare dalla figura soprastante, la configurazione di un *device* di rete attraverso Cacti è molto semplice ed intuitiva. Si tratta, infatti, di compilare la relativa interfaccia con il nome dell'host, il relativo IP, il template da utilizzare in base al device da monitorare (Linux Machine, Windows Machine, Router, Firewall etc), abilitare il protocollo SNMP e completare i relativi campi con la versione del protocollo (version 1, version 2 o version 3) e la relativa "*community string*". In automatico il sistema propone i diversi tipi di grafici che si possono creare in base ai servizi rilevati tramite interrogazione SNMP; per creare i grafici basta poi fare click sul tipo di servizio da "graficare".
A differenza di Nagios, Cacti è molto più facile da configurare e da implementare perché tutte le operazioni necessarie vengono svolte tramite l'utilizzo dell'interfaccia web.

4.1 Implementazione ufficio

Per il sistema di monitoraggio della rete ufficio è stata utilizzata una macchina virtuale dedicata ospitata sul server "*ESX SVILUPPO*" chiamata "*AAA_UB10.04_64_NAGIOS*", in cui sono stati installati i tre applicativi utilizzati per svolgere il monitoraggio dell'infrastruttura di rete. La macchina virtuale per motivi di sicurezza non dispone di un indirizzo pubblico, difatti possiede l'indirizzo 10.1.1.68 che appartiene al range degli indirizzi delle rete interna "*LAN Sianet*", quindi è possibile accedervi solo dalla rete interna oppure tramite VPN se si è al di fuori.
Per la rete ufficio Nagios controlla tutti gli apparati connessi e ne notifica gli eventi tramite l'invio di e-mail.
Per quanto riguarda i dispositivi connessi sono stati utilizzati i plugin standard illustrati nel capitolo precedente durante lo studio del modello di massima.

Di seguito la mappa di rete che è stata generata automaticamente da Nagios tramite l'utilizzo della "*gd library*" e alla funzione "*parent*" contenuta in ognuno dei file di configurazione dei dispositivi di rete.

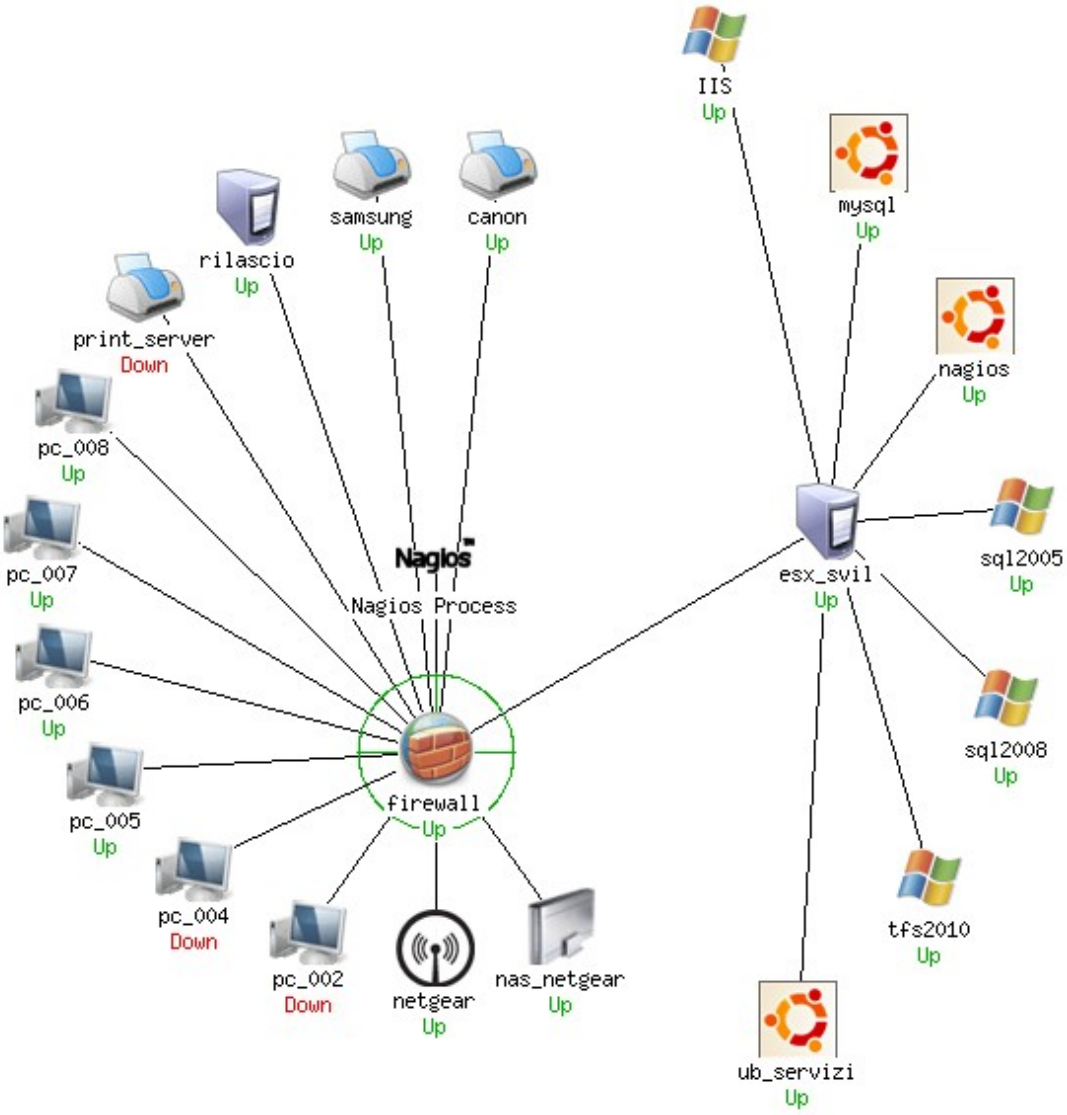


Fig. 4.7 – Mappa della rete "ufficio"

Per avere una visione globale degli apparati e dei relativi servizi, all'interno dell'interfaccia web di Nagios sono stati creati degli "*hostgroup*" che si possono paragonare a degli insiemi. Nel caso della rete ufficio sono stati creati i seguenti *hostgroup*:

- ***ESX SVIL*** che raggruppa al suo interno il server *ESX SVILUPPO* e le relative macchine virtuali ospitate ;

- ***network-printers*** che raggruppa al suo interno tutte le stampanti di rete;
- ***pc-ufficio*** che raggruppa la suo interno tutte le postazioni di lavoro *desktop* presenti nella rete;
- ***switches*** che raggruppa al suo interno il firewall e gli switch presenti in rete.

Nella seguente immagine sono rappresentati i quattro hostgroup con i relativi dispositivi:

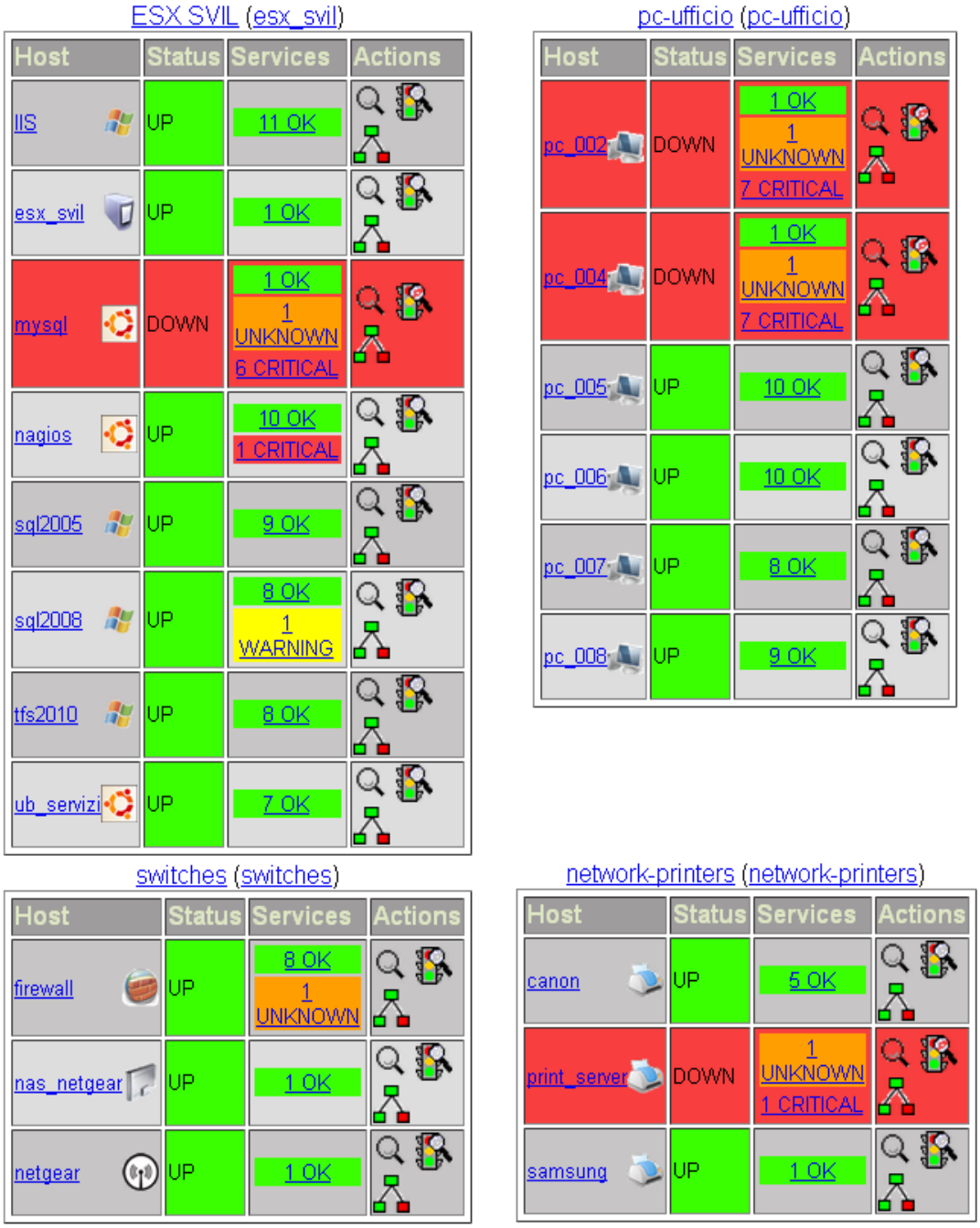


Fig. 4.8 – Visione degli hostgroup per la rete ufficio

Grazie alla creazione degli hostgroup è possibile avere una panoramica dei vari tipi di apparati presenti e del loro relativo stato. In merito agli host che fanno parte dei gruppi "*pc-ufficio*" e "*ESX SVIL*" sono stati creati due script chiamati "*rmtctrl_enable*" (*remote control enable*) ed "*rmtctrl_disable*" (*remote control disable*) presenti su ognuna delle macchine sia fisiche che virtuali appartenenti ad ognuno dei due hostgroup sopracitati. Il compito di questi due script è di abilitare (*rmtctrl_enable*) o disabilitare (*rmtctrl_disable*) i check Nagios appartenenti alla macchina che ha eseguito tali script nella fase rispettivamente di avvio o di spegnimento. L'utilità di questi due script sta nell'evitare che Nagios invii delle notifiche quando un host viene volontariamente spento e successivamente avviato.

Per quanto riguarda i template degli host (*generic-host*) e dei servizi (*generic-service*) sono state utilizzate le seguenti opzioni:

	HOST	**SERVIZI**
Check Period	24x7	24x7
Check Interval	3	3
Max Check Attempts	3	3
Notification Period	24x7	24x7
Notification Interval	0	0
Notification Option	d,u,r	w,u,c,r

Per consultare l'interfaccia web di Nagios è necessario visitare il seguente indirizzo: http://10.1.1.68/nagios e successivamente autenticarsi per accedere all'interfaccia web.

Per quanto riguarda i grafici generati da Cacti, la consultazione è possibile accedendo al seguente indirizzo: http://10.1.1.68/cacti e successivamente autenticandosi per accedere all'interfaccia web.

4.2 Implementazione sala dati

Anche per la rete sala dati è stata utilizzata una macchina virtuale dedicata dove sono stati installati i tre applicativi richiesti per svolgere l'attività di monitoraggio.

La macchina chiamata "*VM_Cacti*" è stata dotata di sistema operativo *Ubuntu Server 10.04* a 64 bit ospitata sul server "*ESX-3*".

La *VM_Cacti* dispone di tre schede di rete con i seguenti indirizzi IP: 10.1.10.4 che appartiene alla zona "*FRONT*"; 192.168.1.104 che appartiene alla zona "*BACK*"; 194.244.116.144 che è l'indirizzo pubblico della macchina.

Per questa macchina si è scelto di utilizzare un indirizzo pubblico per accedere al sistema di monitoraggio senza però pregiudicare la sicurezza, infatti sono state aggiunte delle opportune regole sul firewall tali per cui è possibile accadervi solo da determinati indirizzi IP.

Per consultare l'interfaccia web di Nagios è necessario visitare il seguente indirizzo: http://194.244.116.144/nagios e successivamente autenticarsi per accedere all'interfaccia web.

Per quanto riguarda i grafici generati da Cacti, la consultazione è possibile accedendo al seguente indirizzo: http://194.244.116.144/cacti e successivamente autenticandosi per accedere all'interfaccia web.

Nagios tiene sotto controllo tutti gli apparati connessi alla rete; per l'implementazione di ogni dispositivo è stato applicato il relativo modello di massima con i relativi standard.

Le notifiche per questa rete vengono inviate tramite e-mail agli amministratori della rete. Di seguito la mappa di rete che è stata generata automaticamente da Nagios tramite l'utilizzo della "*gd library*" e alla funzione parent contenuta in ognuno dei file di configurazione dei dispositivi di rete.

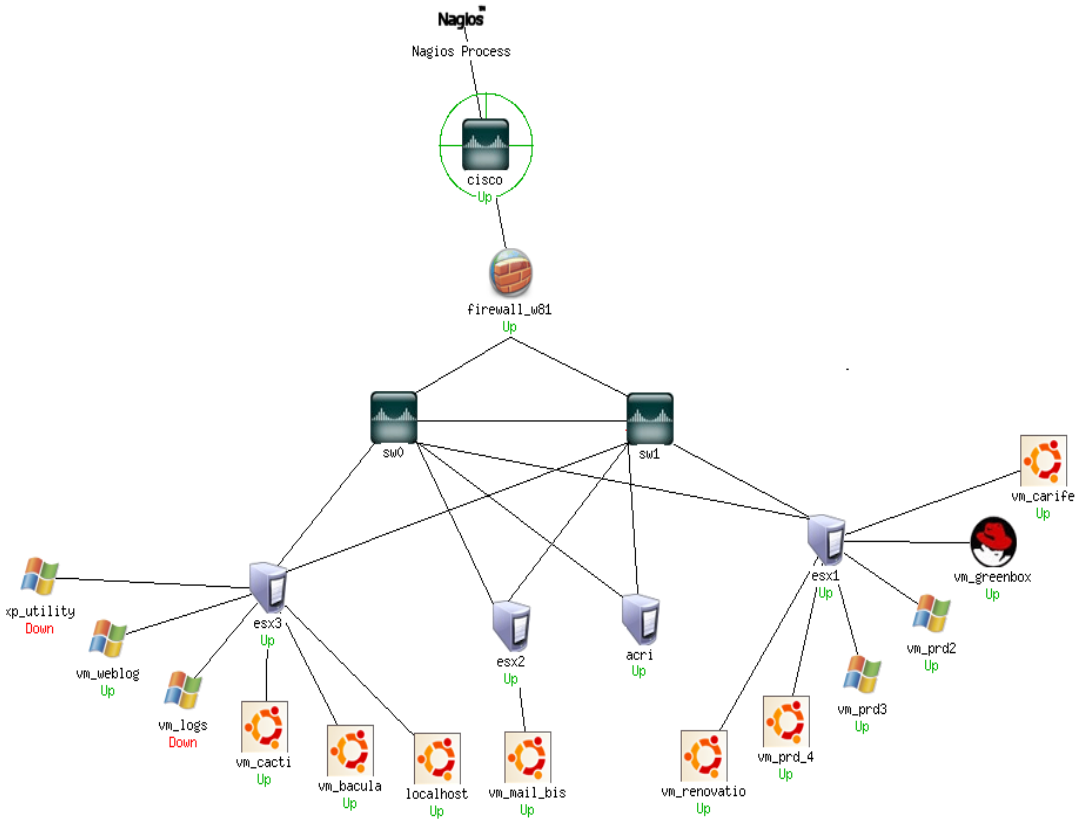


Fig. 4.9 – Mappa della rete "sala dati"

Per avere una visione globale delle macchine e relativi servizi monitorati, oltre alla mappa di rete sono stati creati i seguenti quattro hostgroup:

- ***server acri*** di cui fa parte il relativo server;
- ***server esx1*** che raggruppa al suo interno il server *ESX-1* e le relative macchine virtuali ospitate;
- ***server esx2*** che raggruppa al suo interno il server *ESX-2* e le relative macchine virtuali ospitate;
- ***server esx3*** che raggruppa al suo interno il server *ESX-3* e le relative macchine virtuali ospitate.

Nella seguente immagine sono rappresentati i quattro hostgroup con i relativi dispositivi:

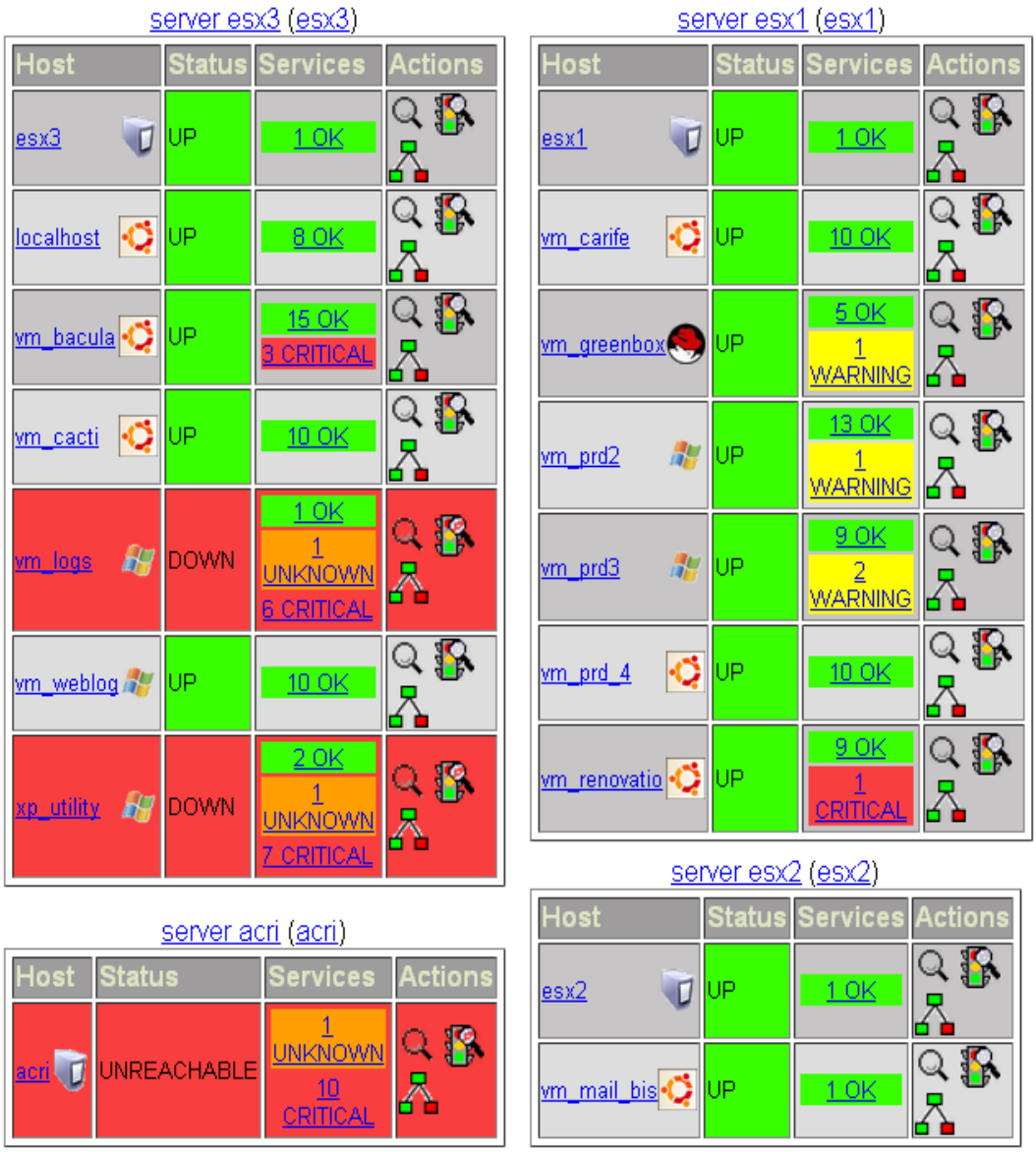


Fig. 4.10 – Visione degli hostgroup per la rete sala dati

Per quanto riguarda i template degli host (*generic-host*) e dei servizi (*generic-service*) sono state utilizzate le seguenti opzioni:

	HOST	**SERVIZI**
Check Period	24x7	24x7
Check Interval	1	1
Max Check Attempts	3	3
Notification Period	24x7	24x7
Notification Interval	60	60
Notification Option	d,u,r	w,u,c,r

Come si può notare le opzioni utilizzate nei template di questa rete sono diverse rispetto alla rete ufficio, infatti in questa rete il Check Interval è stato portato da "*3*" ad "*1*": così facendo Nagios esegue i check ogni minuto per avere una maggiore efficienza nella rilevazione di eventuali errori. Inoltre il Notification Interval è stato portato da "0" a "60" così Nagios invierà un'ulteriore notifica a distanza di un'ora dalla precedente per evidenziare che un problema persiste. Con l'aumentare della frequenza dei check aumenta anche il carico di lavoro della macchina, infatti è stato tenuto conto di questo fattore durante la progettazione della macchina stessa.

4.3 Implementazione Banca Etica

Come per le altre due reti, anche in questa è stata utilizzata una macchina dedicata per il monitoraggio. Trattandosi di una infrastruttura creata secondo gli standard dell'alta affidabilità, la macchina per il monitoraggio è stata creata sul server *BE APP master* e successivamente copiata specularmente sul server *BE APP slave*. La macchina che ospita il sistema di monitoraggio è stata chiamata "*VM_13*" ed è dotata di sistema operativo *Ubuntu Server 10.04* a 64 bit. La *VM_13* possiede, oltre a due indirizzi interni alla rete, un terzo indirizzo di tipo pubblico. L'indirizzo pubblico viene utilizzato per accedere al sistema di monitoraggio; questo è stato reso sicuro tramite l'impiego del protocollo di crittografia "*https*", che oltre ad essere utilizzato per gestire l'accesso al sistema di monitoraggio gestisce anche altri accessi per diversi servizi erogati attraverso questa rete.

Per consultare l'interfaccia web di Nagios è necessario visitare il seguente indirizzo: https://www.bancaetica.com/nagios/ e successivamente autenticarsi per accedere all'interfaccia web.

I grafici generati da Cacti possono essere consultati all'indirizzo: https://www.bancaetica.com/cacti/, autenticandosi per accedere

all'interfaccia web.

Nagios tiene sotto controllo il server *BE APP master* e le relative macchine virtuali presenti su esso, per l'implementazione del server e relative macchine virtuali sono stati applicati i modelli di massima con i relativi standard.

Analogo è stato il lavoro svolto sul server *BE APP slave*.

Le notifiche per questa rete vengono inviate tramite e-mail agli amministratori della rete.

Di seguito la mappa di rete che è stata generata automaticamente da Nagios tramite l'utilizzo della "*gd library*" e alla funzione "*parent*" contenuta in ognuno dei file di configurazione dei dispositivi di rete.

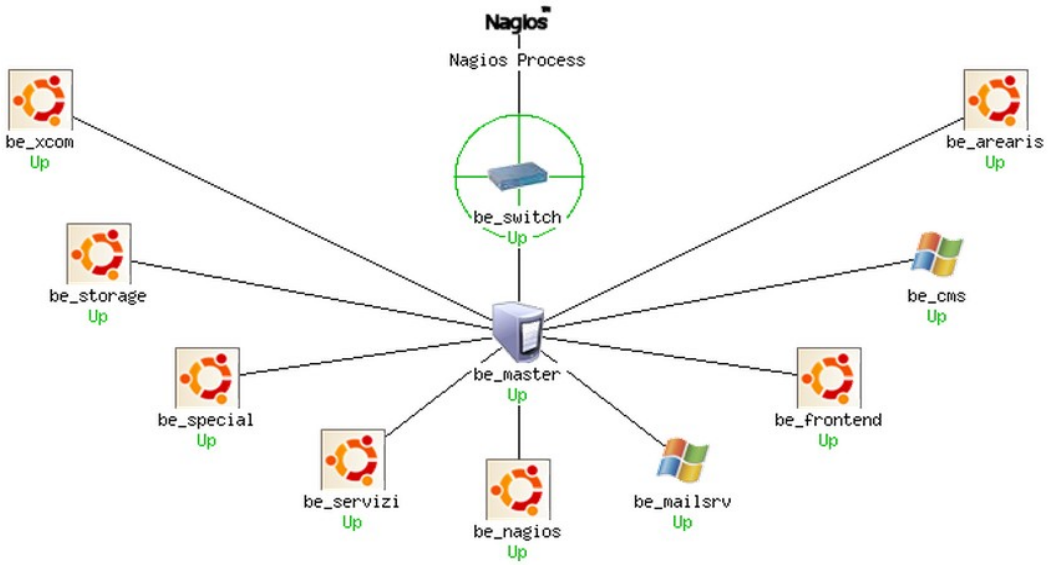


Fig. 4.11 – Mappa della rete "Banca Etica"

Per avere una visione globale delle macchine e relativi servizi monitorati, oltre alla mappa di rete è stato creato un solo hostgroup: ***BE MASTER*** che raggruppa al suo interno il server be_master e le relative macchine virtuali ospitate.

Di seguito è rappresentato l'unico hostgroup con le relative macchine virtuali:

Fig. 4.12 – Visione dell'hostgroup per la rete Banca Etica

Per quanto riguarda i template degli host (*generic-host*) e dei servizi (*generic-service*) sono state utilizzate le seguenti opzioni:

	HOST	**SERVIZI**
Check Period	24x7	24x7
Check Interval	1	1
Max Check Attempts	3	3
Notification Period	24x7	24x7
Notification Interval	60	60
Notification Option	d,u,r	w,u,c,r

Anche in questa rete, come in quella della sala dati, sono stati utilizzati gli stessi valori sulle opzioni "Check Interval" e "Notification Interval" essendo anche questa rete un ambiente di produzione.

Capitolo 5

MODIFICHE

Durante lo svolgimento dello stage è stato necessario risolvere un problema di falsi allarmi legato ad un applicativo che monitorava la raggiungibilità dei siti internet.

L'applicativo su cui si è svolto il lavoro si chiama "*Simple Faileover*", un un prodotto commerciale che verifica attraverso delle richieste di tipo "http", la raggiungibilità dei domini.

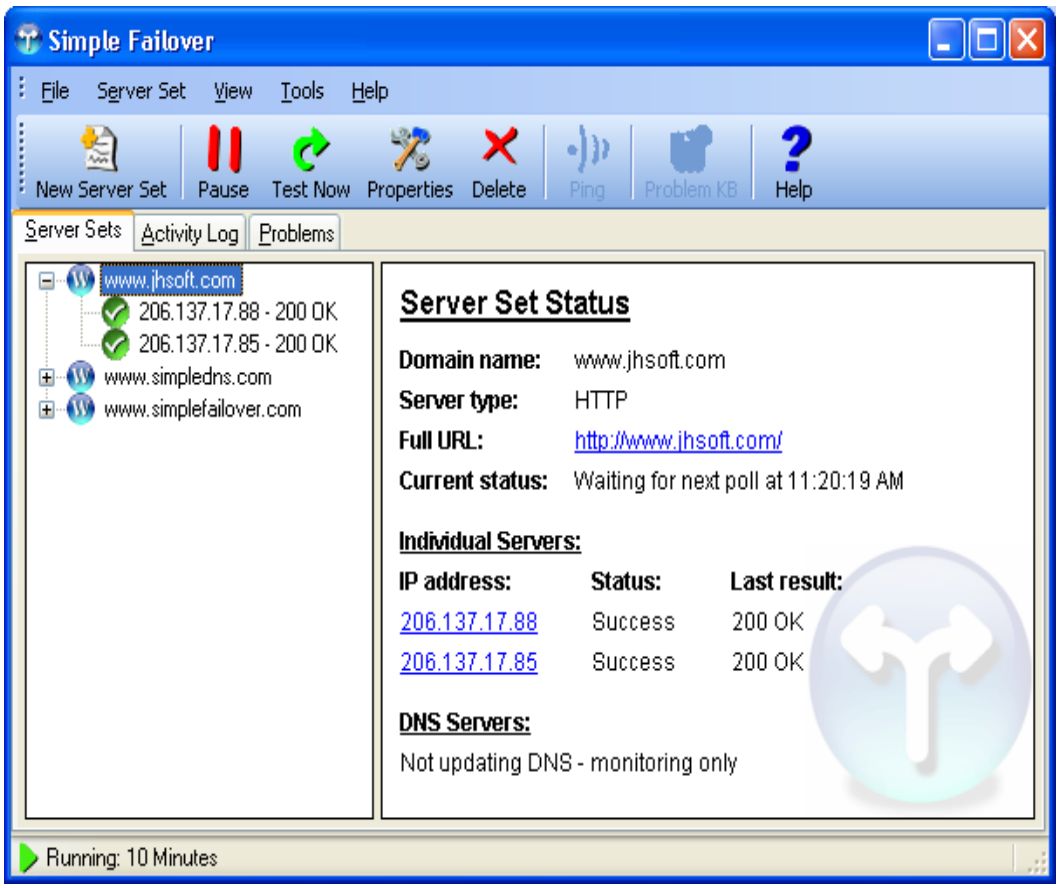


Fig. 5.1 – Interfaccia principale "SFO"

L'azienda aveva acquistato un hosting ad Atlanta su cui aveva installato questo software, la scelta della locazione geografica nasce dal fatto che se i domini risultavano essere raggiungibili dagli Stati Uniti a maggior ragione saranno stati raggiungibili dal territorio italiano.

Il funzionamento di *SFO* (*Simple Faileover*) è semplice: i domini da controllare sono contenuti in un file "*xml*", dove per ogni dominio viene specificato l'indirizzo IP a cui puntano. Il programma esegue i check in base al "*check interval*" specificato nell'opzioni di settaggio, e quando rileva un errore di raggiungibilità invia una e-mail di alert agli amministratori del sistema.

Il funzionamento normale dell'applicativo è stato modificato da parte del sistemista aziendale con lo scopo di personalizzare il funzionamento e manipolare i dati prodotti: è stato quindi creato un database chiamato "*db_SFO*" che raccoglie al suo interno tutti i domini e le specifiche contenute nel file "*xml*" di configurazione insieme ad una serie di altre opzioni, su cui non si entrerà nei particolari perché non necessarie alla risoluzione del problema assegnato.

Sul database operano tre script che hanno le seguenti funzioni: "*StartUpdateTbl*" e "*StartDeleteTbl*" hanno il compito di tenere allineati i record dell'applicativo con il database *db_SFO*; "*check*" ha il compito di effettuare le verifiche sul database e gestire la parte di *alert,* ed è lo script su cui si è lavorato per risolvere il problema.

La creazione del database e relativi script che operano su di esso è stata necessaria per poter quindi sfruttare i risultati prodotti da *SFO,* che è stato utilizzato solo per compiere i check sui vari domini e memorizzare i risultati sul database creato.

Prima di cercare di risolvere il problema delle false notifiche vi è stata una fase di analisi e monitoraggio durata circa una settimana ed eseguita attraverso l'utilizzo di Nagios e Cacti.
Dall'analisi è emerso che il *virtual host* chiamato "*Atlanta server*", su cui era in esecuzione l'applicativo SFO, aveva sia problemi legati alla connettività che alla latenza, andando quindi a causare i falsi allarmi durante le fasi di check.
Dopo aver delineato i due problemi principali si è passato alla ricerca della soluzione che è stata quasi immediata.
Per i problemi legati alla connettività si è inserito nello script "check" il controllo chiamato "*ServerTest*" che veniva eseguito come prima istruzione. Questo controllo andava ad effettuare il ping su indirizzi IP e domini contenuti nel database chiamato "*db_SrvTest*" e così costituito:

IP	**Dominio**	**EU**
74.125.19.99	www.google.com	0
72.30.2.43	www.yahoo.com	0
66.211.181.11	www.ebay.com	0
213.119.21.11	www.msn.it	1
195.210.91.83	www.libero.it	1
212.48.10.150	www.virgilio.it	1

da notare il campo chiamato "*EU*" che se posto ad "*1*" sta a significare che il relativo dominio si trova su territorio europeo viceversa impostato a "*0*".
Se dopo l'esecuzione del controllo *ServerTest* non risultano raggiungibili almeno due domini su territorio europeo e due su territorio non europeo lo script viene interrotto, producendo inoltre una riga di log contenente l'errore di connettività.

Il problema legato alla latenza dell'host è stato risolto prendendo spunto dalla funzione Nagios chiamata "*Max Check Attempts*" che nel sistema di monitoraggio è stata posta a "*3*": ciò significa che Nagios invia una notifica solo dopo tre check con lo stesso esito, così da prevenire l'insorgere di falsi allarmi.

Per realizzare tale funzione sono stati inseriti tre campi nel database *db_SFO* chiamati "*Check 1*", "*Check 2*" e "*Check 3*" andando così a costituire il seguente database:

IP	Dominio	Check 1	Check 2	Check 3	Opzione 1	Opzione n
xxx.xxx.xxx.xxx	dominio1.it	0	0	0		
yyy.yyy.yyy.yyy	dominio2.net	112	112	0		
zzz.zzz.zzz.zzz	dominio3.com	0	0	112		

i codici di errori contenuti all'interno dei campi "*Check*" possono essere di due tipi:

- ***0*** indica che il dominio in questione è raggiungibile;
- ***112*** indica che il dominio non è raggiungibile.

Durante l'esecuzione dello script "*check*" se per un determinato dominio si ha lo stesso codice di errore nel campo "Check 1" e "Check 2", mentre il codice contenuto nel campo "Check 3" è diverso allora il sistema provvede ad inviare una notifica che può essere di due tipi:

- ***notifica 112*** indica che il dominio in questione è irraggiungibile per esempio il dominio chiamato "*dominio2.net*" contenuto nella tabella precedente ha due codici "*112*" nei primi due campi "*Check*" e un codice "*0*" nell'ultimo campo;
- ***notifica 0*** indica che il dominio in questione è tornato online per esempio il dominio chiamato "*dominio3.com*" contenuto nella tabella precedente ha due codici "*0*" nei primi due campi "*Check*" e un codice "*112*" nell'ultimo campo.

Grazie alle due tecniche precedentemente descritte e adottate si è riusciti a prevenire situazioni di falso allarme.

Capitolo 5

CONCLUSIONI

Per quanto riguarda il sistema di monitoraggio implementato sulle tre reti aziendali è importante sottolineare l'utilizzo di software open-source, secondo me prodotti molto validi perfino al di sopra di molti prodotti commerciali.

Quello che a mio parere caratterizza l'elevata qualità dei software che ho scelto per realizzare questo sistema è che dietro ogni applicativo utilizzato c'è la cooperazione libera di molti sviluppatori e programmatori, che realizzano applicativi secondo le proprie esperienze ed ognuno può collaborare per elevare ancora di più gli standard di questi prodotti.

Altro punto di forza di questi applicativi open-source è di avere a disposizione dei forum ufficiali dove potersi confrontare con altre persone per scambiarsi idee, affrontare e risolvere nuovi problemi. Durante lo svolgimento di questo stage mi sono confrontato diverse volte con persone che utilizzavano Nagios e Cacti sia per usi "*domestici*" che aziendali riuscendo anche grazie all'esperienza acquisita durante lo stage a proporre delle soluzioni.

Ovviamente come tutti i programmi anche Nagios possiede alcuni bug che verranno sicuramente risolti con le prossime "*release*": per esempio non è stato possibile monitorare i parametri hardware e di rete dei server con sistema operativo "*VMware ESX*", però sempre attraverso i forum si sta lavorando allo sviluppo di plugin per

assolvere a tali esigenze ed anche io sto cercando di dare il mio contributo.

In merito all'esperienza acquisita durante lo stage mi ritengo molto soddisfatto, perché, prima di entrare in questa nuova realtà ero solamente uno studente di ingegneria informatica con un buon bagaglio teorico ma poca "esperienza sul campo".

All'inizio dello stage ho dovuto lavorare molto per acquisire le conoscenze necessarie per lavorare sulle reti senza pregiudicarne il funzionamento, inoltre cosa importante ho imparato a lavorare all'interno di un team.

Lavorando all'interno di un team si capisce che l'individualismo non è necessario per raggiungere l'obiettivo ma è fondamentale la cooperazione.

Inoltre durante questa esperienza sono stato affiancato da un ottimo tutor che mi ha seguito e mi ha dato giusti "input" per svolgere la mia attività in maniera positiva.

Grazie a questa esperienza mi sono appassionato ancora di più alle reti informatiche e alla loro gestione e monitoraggio, infatti sono riuscito a capire ed utilizzare molti strumenti e tecniche che durante i miei studi avevo potuto conoscere solo dal lato teorico, per esempio alcune tecniche sulla sicurezza informatica correlate alle politiche di firewalling, l'utilità delle VPN ecc.

Soddisfacente è stato l'ambiente in cui ho lavorato in quanto costituito da persone giovani che nonostante ciò hanno alle spalle un'ottima formazione ed esperienza sul campo, fattore questo molto importante.

In azienda il clima era molto sereno, lontano dalle mie concezioni classiche di azienda in cui il lavoratore è messo costantemente sotto pressione da parte dei suoi "superiori"; ovviamente ciò non pregiudica il normale svolgimento delle attività lavorative, nè la professionalità impiegata.

Il sistema che ho implementato viene tuttora utilizzato con ottimi risultati, inoltre l'azienda è rimasta soddisfatta del mio operato tanto che si è parlato di estendere questo tipo di attività ad altri clienti, passando quindi da un'attività per uso interno ad un servizio erogato e quindi ad un'attività remunerativa dal punto di vista aziendale.

Per concludere mi sento di consigliare agli studenti come me, questo tipo di esperienza rispetto all'usuale tesina. In questo modo lo studente ha l'opportunità di realizzare con la propria conoscenza e capacità, progetti concreti, trarne esperienza pratica e cosa ancora più importante (come è successo al sottoscritto) sentirsi molto soddisfatti.

APPENDICE

In questa sezione sono contenuti tutti gli schemi redatti durante lo svolgimento dello stage e contenuti in questa tesi.

INDICE APPENDICE

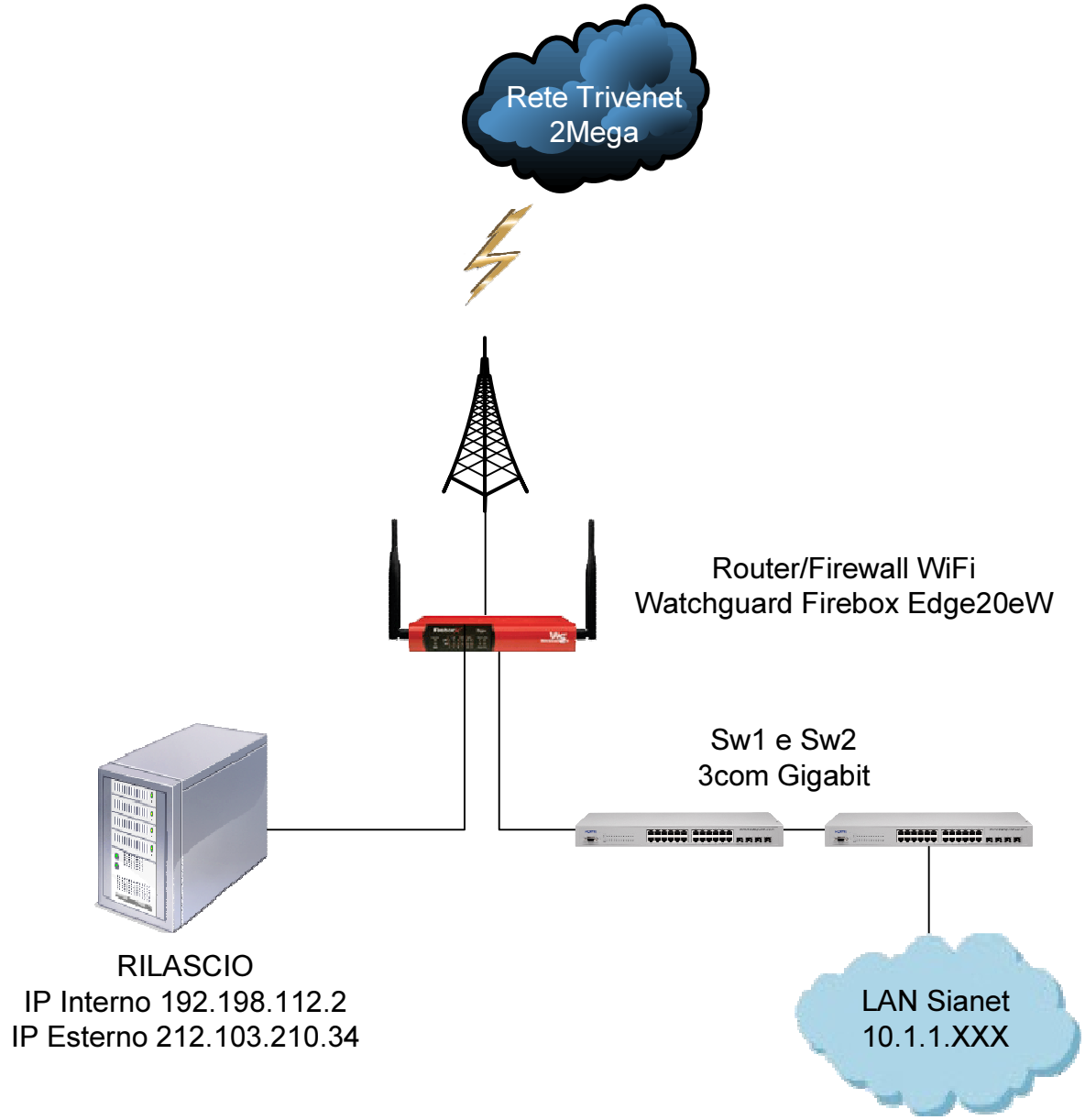
Appendice 1 – Rete Sianet

Appendice 2 – ESX SVILUPPO

Appendice 3 – Schema di rete logico Netscalibur

Appendice 4 – Schema di rete logico zona Banca Etica

Rete SIANET



INDIRIZZI PUBBLICI
Dal 212.103.210.34 al 212.103.210.39

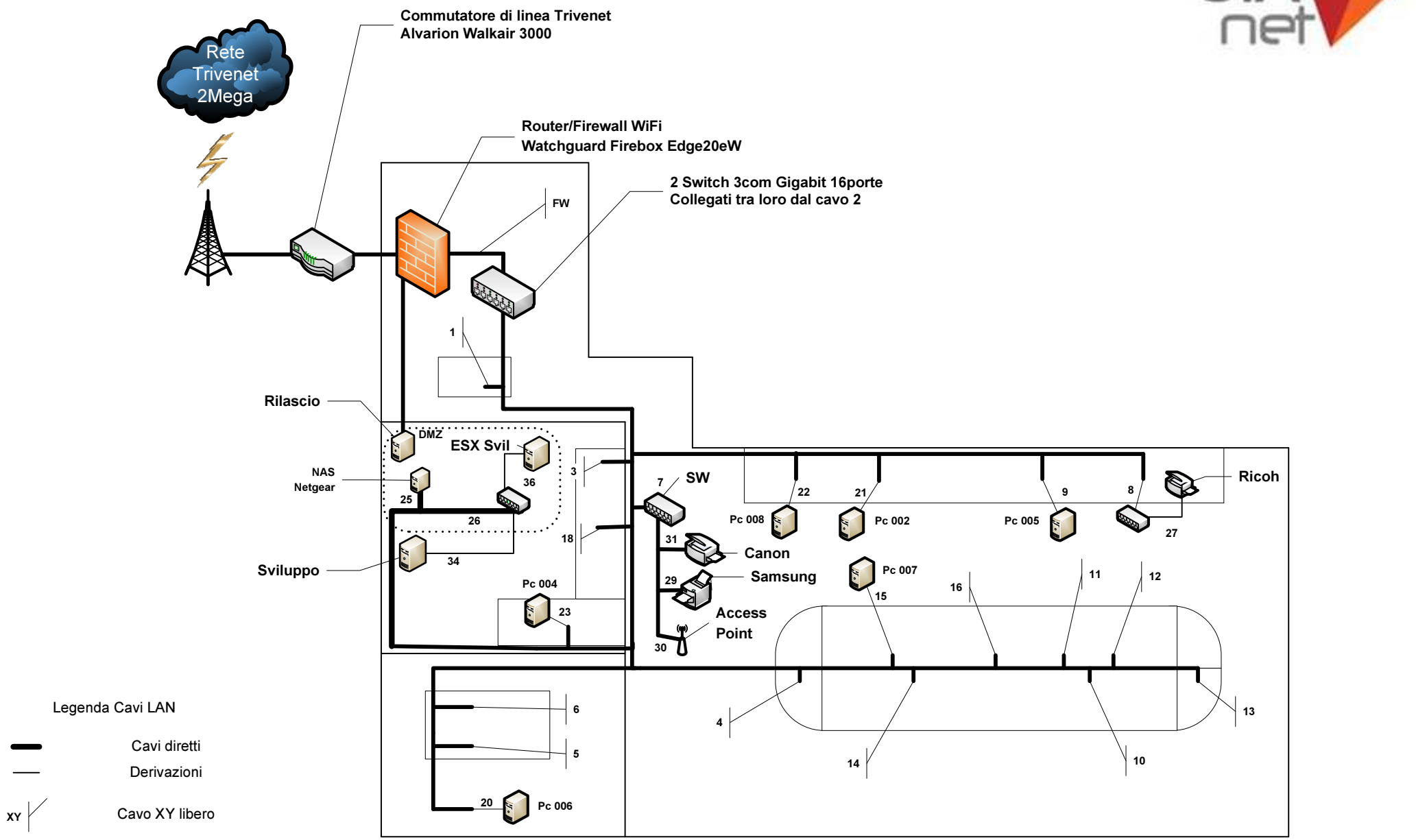
212.103.210.34 Rilascio

LAN 10.1.1.XXX
Dal .0 al .99 → Apparati server
Dal .100 al .150 → DHCP Client
Dal .151 al .160 → DHCP VPN with SSL

IP fissi

10.1.1.1	Gateway / FW
10.1.1.52	DNS Server
10.1.1.80	ESX Svil
10.1.1.92	Access Point Netgear WG602v4
10.1.1.94	NAS Netgear Ready Duo
10.1.1.95	Stampante Samsung CLP-350N
10.1.1.96	Stampante Canon MF4390
10.1.1.98	Stampante Ricoh 1013F
10.1.1.99	SVILUPPO WINDOWS

Schema Cavi SIANET



Connessione Cavi SIANET

Web	IP	CAVO
Indirizzi Pubblici		
Dal 212.103.210.34 al 212.103.210.39		
Rilascio	212.103.210.34	
LAN 10.1.1.XXX		
DHCP Client	Dal .100 al .150	
DHCP VPN with SSL	Dal .151 al .160	
Apparati server	Dal .0 al .99	
IP Fissi		
Gateway / FW	10.1.1.1	FW
Server ESX Svil	10.1.1.80	36
Access Point Netgear (Sianet2)	10.1.1.92	30
NAS Netgear Ready Duo	10.1.1.94	25
Stampante Samsung CLP-350N	10.1.1.95	29
Stampante Canon MF4349	10.1.1.96	31
Stampante Ricoh 1013F	10.1.1.98	27
SVILUPPO WINDOWS	10.1.1.99	35
DNS Server	10.1.1.52	
IP Interno Rilascio	192.168.112.2	DMZ

Configurazione WAN	IP
GATEWAY	212.103.210.33
DNS Primario	212.103.192.10
DNS Alternativo	212.103.195.195

Switch 3com Gigabit - Sw2 (superiore)								
Note	Scrivania ingresso	Cavo Uplink/Downlink Switch1	Ufficio - isola centrale	Switch Nilox 5 porte	Ufficio - Scrivania a muro Pc005	Ufficio - isola centrale	Ufficio - isola centrale	Ufficio - isola centrale
Nome Cavo	1	2	4	8	9	10	11	12
Porte Switch	1	2	3	4	5	6	7	8
	9	10	11	12	13	14	15	16
Nome Cavo	13	14	15	16	21	22		
Note	Ufficio - isola centrale	Ufficio - isola centrale	Ufficio - isola centrale Pc007		Ufficio - Scrivania a muro Pc002	Ufficio - Scrivania a muro Pc008		

Switch 3com Gigabit - Sw1 (inferiore)								
Note	Armadio ingresso - Firewall Firebox WatchGuard	Cavo Uplink/Downlink Switch2	Sala Server Switch Nilox 5 porte	Sala Server NAS Netgear Ready Duo	Sala Riunioni angolo LCD	Sala Riunioni - scrivania	Sala Riunioni - scrivania	
Nome Cavo	FW	2	26	25	20	6	5	18
Porte Switch	1	2	3	4	5	6	7	8
	9	10	11	12	13	14	15	16
Nome Cavo	23	3	7	19	17			
Note	Sala Server - scrivania vicino porta	Sala Server - Switch D-Link 2 PC amm.	Switch mobiletto stampante	Cavo a pavimento	Cavo a pavimento			

ESX SVILUPPO

Caratteristiche Hardware

CAMPO	VALORE
Nome PC	ESX SVILUPPO
Case	PowerEdge T410 Tower Chassis for Up to 6x 3.5" Cabled HDDs with Quad-Pack LED Diagnostics
Product Number	GJLCP4J
Network Adapter	Broadcom NetExtreme II BCM5716 1000Base-T
Storage Adapter	Dell Perc H700 Adapter
Numero Processori	4
Scheda Madre	DELL
Bios	Dell Inc. System BIOS 1.3.8 26/02/2010
RAM	12GB (3x4GB RDIMM dual rank) 1.066MHz
Hard Disk 0	450GB SAS 6Gb/s 15.000rpm 3,5"
Hard Disk 1	450GB SAS 6Gb/s 15.000rpm 3,5"
Hard Disk 2	500 GB Samsung RAID 5
Hard Disk 3	500 GB Samsung RAID 5
Hard Disk 4	500 GB Samsung RAID 5
Sistema Operativo	ESX 4
Accessori	iDRAC6 Embedded BMC, Redundant Power Supply (2 PSU) 580W
Lettore DVD	16X DVD-ROM Drive SATA with SATA Cable for Win2K8 R2

IP ESX SVILUPPO

IP	DESCRIZIONE
10.1.1.80	IP interno
10.1.1.81	DRAC

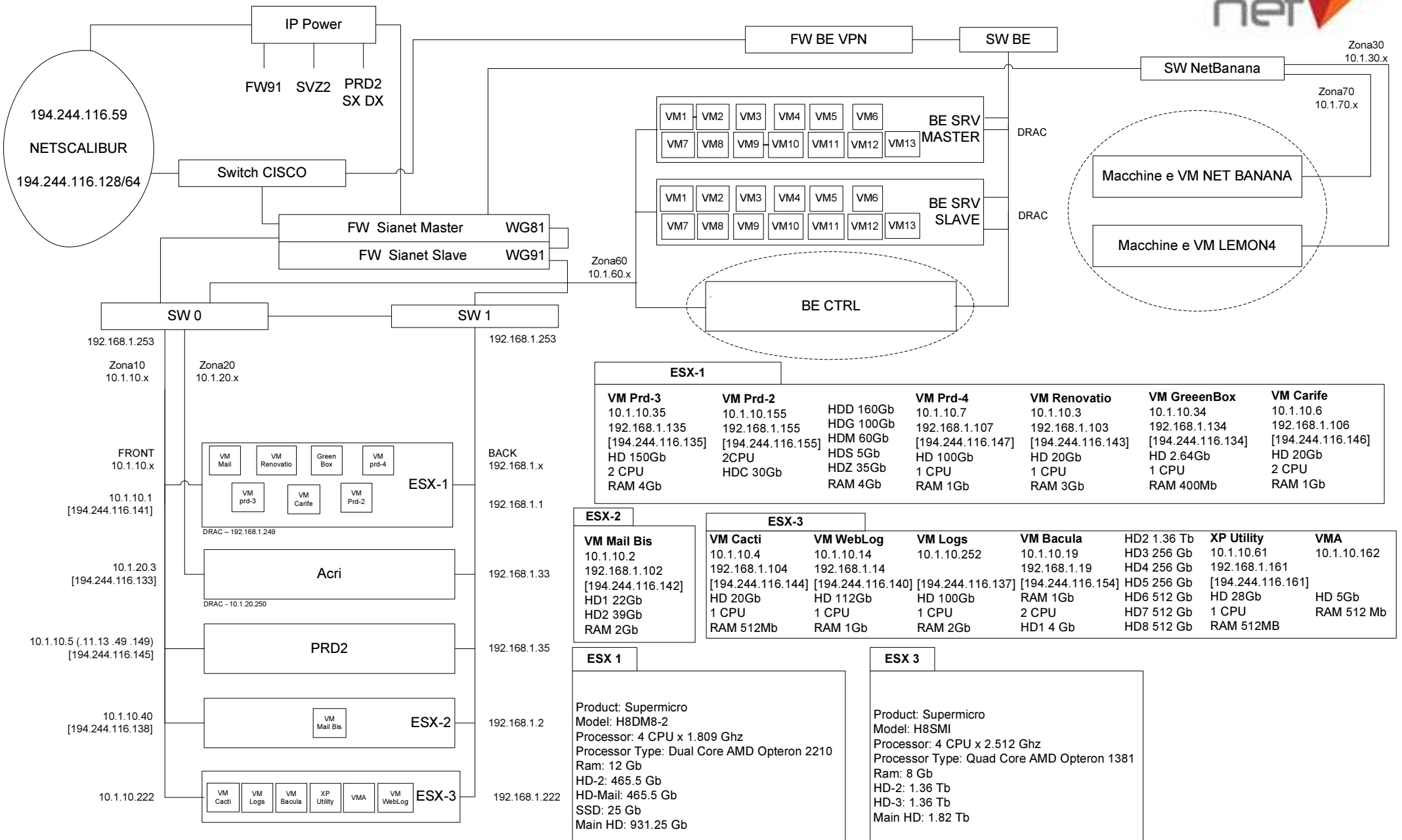
IP Macchine Virtuali

IP primario	IP secondario	Proxy	DESCRIZIONE
10.1.1.51		10.1.1.52	AAA_ORA11SVIL
10.1.1.52			AAA_UB_10.04_SERVIZI
10.1.1.53	10.1.2.53	10.1.1.52	TST_UB_10.04_BEAREARIS
10.1.1.54		10.1.1.52	TST_UB_10.04_BESPECIAL
10.1.1.55		10.1.1.52	AAA_vSphere Management
10.1.1.56		10.1.1.52	TST_UB9.10_64_Nagios
10.1.1.57	10.1.2.57	10.1.1.52	AAA_ORA9SVIL
10.1.1.58			
10.1.1.59			
10.1.1.60		10.1.1.52	AAA_W2008R2_TFS2010
10.1.1.61	10.1.2.61	10.1.1.52	AAA_WXP_VSNET
10.1.1.62		10.1.1.52	TST_W7UIN_64
10.1.1.63		10.1.1.52	AAA_W2K3R2_IIS
10.1.1.64		10.1.1.52	AAA_W2K3R2_SQL2008R2
10.1.1.65		10.1.1.52	AAA_W2K3R2_SQL2005
10.1.1.66		10.1.1.52	AAA_UB_10.04_64_MYSQL
10.1.1.67		10.1.1.52	TST_W2K3_MOSS2007
10.1.1.68		10.1.1.52	AAA_UB10.04_64_NAGIOS
10.1.1.69			
10.1.1.70		10.1.1.52	ZZZ_UB_9.04_BECMS_PROXY
10.1.1.71			
10.1.1.72		10.1.1.52	AAA_WXP_UTILITY
10.1.1.73		10.1.1.52	TPL_WXP_VUOTA
Dinamico		10.1.1.52	TPL_W2K8R2_VUOTA
Dinamico		10.1.1.52	TST_UB_9.04_CLUSTER_SLAVE
Dinamico		10.1.1.52	ZZZ_UB_8.10_BECMS
Dinamico		10.1.1.52	ZZZ_UB_9.04_BEORG
Dinamico		10.1.1.52	TST_UB_9.04_CLUSTER_MASTER
Dinamico		10.1.1.52	ZZZ_W2K3_BECMS

Caratteristiche Hardware Macchine Virtuali

VM	NUM CPU	RAM	HD 0	HD 1	HD 2	HD 3
AAA_ORA11SVIL	1	2Gb	10 Gb	20 Gb		
AAA_ORA9SVIL	1	512 Mb	30Gb			
AAA_UB_10.04_SERVIZI	1	1Gb	50Gb			
AAA_UB_10.04_64_MYSQL	1	1Gb	24Gb			
AAA_UB10.04_64_NAGIOS	1	512Mb	8Gb			
AAA_vSphere Management	1	512Mb	5Gb			
AAA_W2K3R2_IIS	1	1Gb	50Gb	80Gb	80Gb	80Gb
AAA_W2K3R2_SQL2005	1	1Gb	50Gb			
AAA_W2K3R2_SQL2008R2	1	1Gb	50Gb			
AAA_W2K8R2_SERVIZI	1	1Gb	40Gb			
AAA_W2008R2_TFS2010	1	2Gb	40Gb			
AAA_WXP_VSNET	2	1Gb	20Gb			
TPL_W2K8R2_VUOTA	1	2Gb	40Gb			
AAA_WXP_UTILITY	1	256Mb	124Gb			
TST_UB_10.04_BEAREARIS	1	1Gb	50Gb			
TST_UB_10.04_BESPECIAL	1	1Gb	50Gb			
TST_UB_9.04_CLUSTER_MASTER	1	1Gb	8Gb	5Gb		
TST_UB_9.04_CLUSTER_SLAVE	1	1Gb	8Gb	5Gb		
TST_UB9.10_64_Nagios	1	1Gb	15Gb			
TST_W2K3_MOSS2007	2	1Gb	32Gb			
TST_W7UIN_64	1	2Gb	40Gb			
ZZZ_UB_8.10_BECMS	1	512Mb	24Gb			
ZZZ_UB_9.04_BEORG	1	512Mb	8Gb			
ZZZ_UB_9.04_BECMS_PROXY	2	512Mb	8Gb			
ZZZ_W2K3_BECMS	1	512Mb	10Gb	8Gb		

Schema di rete logico NETSCALIBUR



ESX-1

VM Prd-3	VM Prd-2		VM Prd-4	VM Renovatio	VM GreeenBox	VM Carife
10.1.10.35	10.1.10.155	HDD 160Gb	10.1.10.7	10.1.10.3	10.1.10.34	10.1.10.6
192.168.1.135	192.168.1.155	HDG 100Gb	192.168.1.107	192.168.1.103	192.168.1.134	192.168.1.106
[194.244.116.135]	[194.244.116.155]	HDM 60Gb	[194.244.116.147]	[194.244.116.143]	[194.244.116.134]	[194.244.116.146]
HD 150Gb	2CPU	HDS 5Gb	HD 100Gb	HD 20Gb	HD 2.64Gb	HD 20Gb
2 CPU	HDC 30Gb	HDZ 35Gb	1 CPU	1 CPU	1 CPU	2 CPU
RAM 4Gb		RAM 4Gb	RAM 1Gb	RAM 3Gb	RAM 400Mb	RAM 1Gb

ESX-2

VM Mail Bis
10.1.10.2
192.168.1.102
[194.244.116.142]
HD1 22Gb
HD2 39Gb
RAM 2Gb

ESX-3

VM Cacti	VM WebLog	VM Logs	VM Bacula	HD2 1.36 Tb	XP Utility	VMA
10.1.10.4	10.1.10.14	10.1.10.252	10.1.10.19	HD3 256 Gb	10.1.10.61	10.1.10.162
192.168.1.104	192.168.1.14		192.168.1.19	HD4 256 Gb	192.168.1.161	
[194.244.116.144]	[194.244.116.140]	[194.244.116.137]	[194.244.116.154]	HD5 256 Gb	[194.244.116.161]	
HD 20Gb	HD 112Gb	HD 100Gb	RAM 1Gb	HD6 512 Gb	HD 28Gb	HD 5Gb
1 CPU	1 CPU	1 CPU	2 CPU	HD7 512 Gb	1 CPU	RAM 512 Mb
RAM 512Mb	RAM 1Gb	RAM 2Gb	HD1 4 Gb	HD8 512 Gb	RAM 512MB	

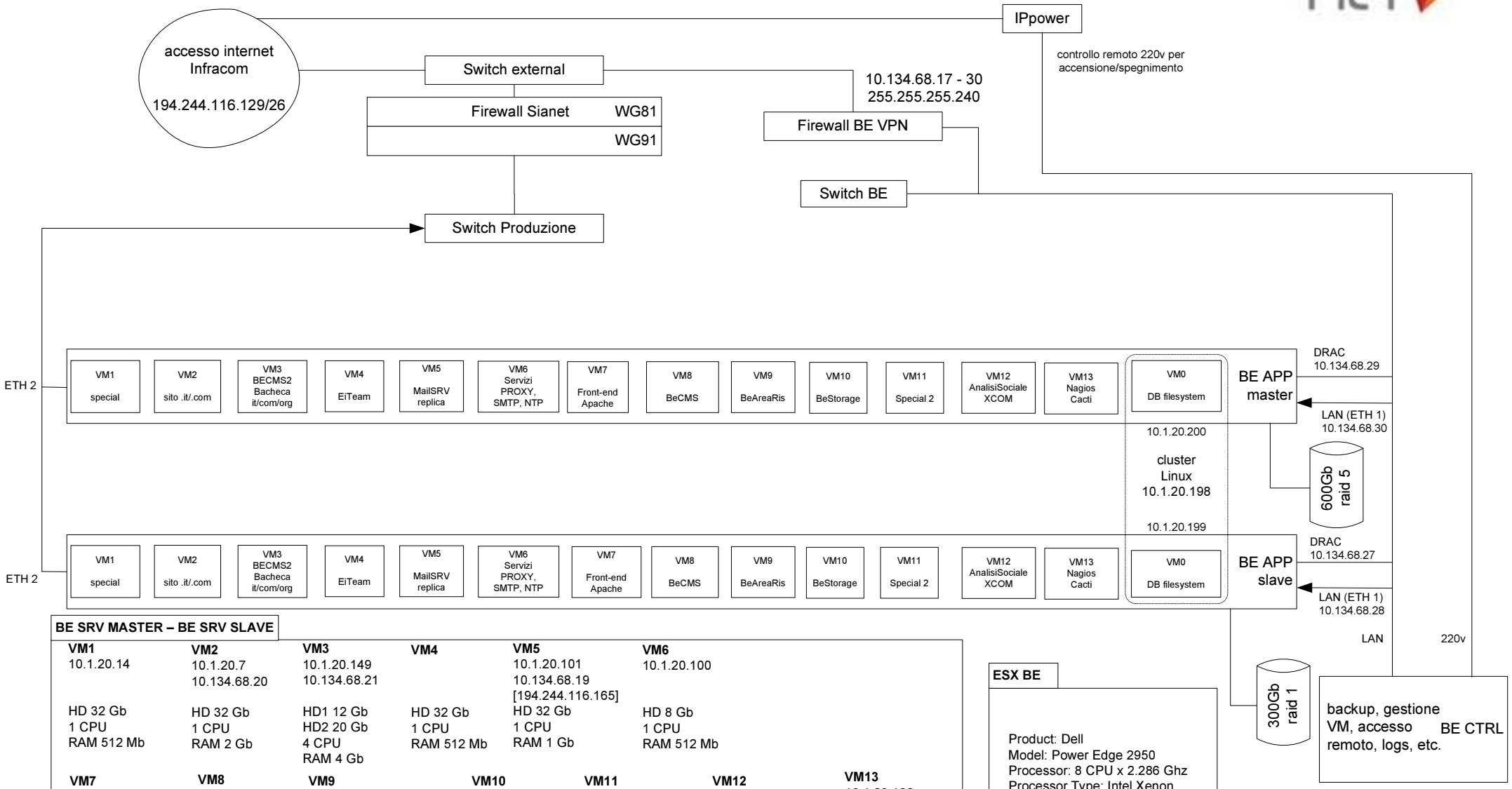
ESX 1

Product: Supermicro
Model: H8DM8-2
Processor: 4 CPU x 1.809 Ghz
Processor Type: Dual Core AMD Opteron 2210
Ram: 12 Gb
HD-2: 465.5 Gb
HD-Mail: 465.5 Gb
SSD: 25 Gb
Main HD: 931.25 Gb

ESX 3

Product: Supermicro
Model: H8SMI
Processor: 4 CPU x 2.512 Ghz
Processor Type: Quad Core AMD Opteron 1381
Ram: 8 Gb
HD-2: 1.36 Tb
HD-3: 1.36 Tb
Main HD: 1.82 Tb

Schema di rete logico Infracom – zona BancaEtica



BE SRV MASTER – BE SRV SLAVE

VM1 10.1.20.14 HD 32 Gb 1 CPU RAM 512 Mb	**VM2** 10.1.20.7 10.134.68.20 HD 32 Gb 1 CPU RAM 2 Gb	**VM3** 10.1.20.149 10.134.68.21 HD1 12 Gb HD2 20 Gb 4 CPU RAM 4 Gb	**VM4** HD 32 Gb 1 CPU RAM 512 Mb	**VM5** 10.1.20.101 10.134.68.19 [194.244.116.165] HD 32 Gb 1 CPU RAM 1 Gb	**VM6** 10.1.20.100 HD 8 Gb 1 CPU RAM 512 Mb

VM7 10.1.20.9 [194.244.116.148] HD 24 Gb 1 CPU RAM 512 Mb	**VM8** 10.1.20.150 10.134.68.26 HD1 10 Gb HD2 8 Gb 1 CPU RAM 4 Gb	**VM9** 10.1.20.10 10.134.68.22 [194.244.116.153] (SVIL) HD1 32 Gb 2 CPU RAM 2 Gb	**VM10** 10.1.20.11 10.134.68.23 HD1 50 Gb 2 CPU RAM 1 Gb	**VM11** 10.1.20.6 [194.244.116.136] HD 32 Gb 2 CPU RAM 1 Gb	**VM12** 10.1.20.13 10.134.68.18 [194.244.116.131] HD 8 Gb 2 CPU RAM 512 Mb	**VM13** 10.1.20.102 10.134.68.20 [194.244.116.148] HD 8 Gb 2 CPU RAM 512 Mb

ESX BE

Product: Dell
Model: Power Edge 2950
Processor: 8 CPU x 2.286 Ghz
Processor Type: Intel Xenon
Ram: 8Gb
HD: 553,Gb

BIBLIOGRAFIA

Documentazione ufficiale Nagios

http://nagios.org/

Documentazione ufficiale Cacti

http://www.cacti.net/

Documentazione ufficiale MRTG

http://oss.oetiker.ch/mrtg/

Community Nagios

http://wiki.nagios.org/index.php/Forums

http://www.monitoringexchange.org/

Documentazione ufficiale NSClient ++

http://nsclient.org/nscp/

Documentazione ufficiale Simple Fileover

http://www.simplefailover.com/

Informazioni generali

http://it.wikipedia.org/

Appunti corso di reti e calcolatori

RINGRAZIAMENTI

Voglio dedicare questo giorno, per me molto importante, a due persone speciali: nonno Domenico e Gianni, che hanno sempre vegliato su di me dandomi la forza necessaria per portare a termine gli studi. Inoltre, continuano a darmi le energie per affrontare i numerosi ostacoli che la vita mi riserva.

Questo giorno lo voglio anche dedicare a nonno Andrea, nonna Assunta e nonna Vittoria; che a causa dell'età e degli acciacchi sono rimasti nella mia amata terra, la Calabria.

Voglio dire GRAZIE alla mia famiglia e alla mia dolce metà Anna che, durante questi LUNGHI anni di studio, mi hanno supportato e SOPPORTATO e continuano a farlo (almeno credo?!).

Ringrazio anche la sorella (Eliana), il Doc (Alessio), Tipo (Fausto), Jack (Giacomo) e gli altri amici per essermi stati sempre vicino nei momenti di SOFFERENZA (lo studio) ma sopratutto nei momenti di "CAZZEGGIO" (scusate il termine).

Un grazie particolare va a Dario (Drom), il mio mitico tutor, che mi ha dato veramente una grossa mano a risolvere (e creare) numerosi problemi e darmi gli giusti insegnamenti per svolgere la mia attività di stagista (sembra quasi vero a leggerlo....). Altro grazie va ad Andrea (Finot) per le informazioni ed il supporto tecnico in merito alla rete di Banca Etica.

Ovviamente, devo ringraziare il Dott. Carlo Fantozzi, il mio relatore, che è stato molto disponibile durante tutto lo svolgimento dello stage.

Per i pochi o spero nulli errori mi sento in dovere di ringraziare: il Dott. Carlo Fantozzi, la mamma (Rosella) e Sofia (è riuscita a trovare cinque errori in due pagine mahh....).

Cosa importante è che per il momento mi sento libero da una famosa raccomandazione che mi faceva spesso mia mamma e mio papà (Raffaele detto anche Baffo): << Mi raccomando STUDIA!!!!!>> ed io da bravo ragazzo rispondevo prontamente: << SI SI TRANQUILLI!!!!>>.

A parte gli scherzi, non si finisce MAI di studiare.

Ora basta ringraziamenti perché se no rischio di essere troppo buono.....