

Università degli Studi di Camerino

Scuola di Scienze e Tecnologie

Corso di Laurea in Informatica Classe L-31



Protocollo SNMP per il Monitoraggio di Reti. Case Study: Zabbix

Studente:

Spinaci Marco

Matr. 082837

Relatore:

Prof. Marcantoni Fausto

Anno Accademico 2013 – 2014

SOMMARIO

1	Introduzione: Obiettivo Tesi	1
2	Funzioni ed Utilità del Monitoraggio delle Reti	2
3	SNMP.....	4
3.1	Informazioni e Caratteristiche generali sul Protocollo	4
3.2	Evoluzione del Protocollo	6
3.2.1	SNMP v1	6
3.2.2	SNMPv2	8
3.2.3	SNMPv3	9
3.3	SNMP nel Dettaglio	10
3.3.1	Formato dei Messaggi	10
3.3.2	Oggetti SNMP	11
3.4	MIB	12
3.5	Applicativi e Tool Grafici.....	14
3.5.1	Net-SNMP	14
3.5.2	SNMP MIB Browser	15
3.5.3	OpenNMS	15
3.5.4	Ganglia.....	15
3.5.5	Nagios	16
3.5.6	Tabella Riassuntiva Software di Monitoraggio	17
4	Zabbix: Monitoring Tool.....	19
4.1	Cosa è Zabbix.....	19
4.2	Funzionalità Generali	19
4.3	Architettura	21
4.4	Data Flow	22
4.5	Concetti Base Necessari	22
4.6	Installazione	24
4.6.1	Prerequisiti	24
4.6.2	Installazione e Configurazione LAMP	24
4.6.3	Installazione e Configurazione Zabbix Server (compilato da sorgenti)	26
4.6.4	Installazione e Configurazione Zabbix Agent su varie macchine (da pacchetti).....	28
4.7	Analisi Interfaccia Utente	30
4.7.1	DashBoard	30
4.7.2	Overview.....	31
4.8	Configurazione Generale.....	32
4.8.1	Configurazione Host con Agent	32
4.8.2	Configurazione Host con SNMP.....	35
4.8.3	Configurazione Item	36

4.8.4	Configurazione dei Trigger	41
4.8.5	Configurazione Ricezione Notifiche a Problemi	43
4.8.6	Configurazione Template	46
4.9	Analisi dei Risultati	48
4.9.1	Implementazione e studio dei Grafici.....	48
4.10	Acknowledgment degli Eventi.....	53
4.11	Discovery	53
4.11.1	Network Discovery o Automatic Discovery	53
4.11.2	Active Agent Auto-Registration	55
4.12	API e Plugins	55
5	Conclusioni.....	57
6	Bibliografia	58
7	Sitografia	58
8	Ringraziamenti	59

1 INTRODUZIONE: OBIETTIVO TESI

Lo straordinario sviluppo delle reti di computer dell'ultimo decennio ha aperto nuovi scenari per quanto riguarda l'utilizzo dei personal computer, si pensi a come si è modificato l'utilizzo di Internet grazie alla banda larga, e soprattutto a come è cambiata la concezione del personal computer, trasformando un computer isolato in uno strumento in grado di comunicare con tutto il mondo.

Ovviamente lo sviluppo delle reti porta ad un inevitabile incremento delle loro dimensioni e se fino a poco tempo fa il concetto di rete era legato ad ambiti aziendali o comunque locali, adesso le reti possono coprire intere aree geografiche. Questa crescita in dimensioni e in numero di host collegati crea agli amministratori notevoli problemi di gestione e manutenzione della rete; infatti se una rete si estende in un'area molto vasta non è detto che l'amministratore sia in grado di raggiungere fisicamente e in qualsiasi momento tutti gli host e tutti i nodi della rete, quindi per rimediare a questi problemi si utilizzano programmi che siano in grado di monitorare la rete e di prevenire possibili guasti ai dispositivi collegati.

Nel 1989 nasce il protocollo SNMP che viene pensato come punto di partenza su cui sviluppare dei sistemi che siano in grado di risolvere e prevedere tutti i problemi che nascono dalla gestione di una rete. La versatilità di questo protocollo gli ha permesso di trovare da subito un riscontro positivo dal mondo degli sviluppatori e dalle aziende del settore, infatti dopo la prima versione ne sono state implementate altre due (anche se la prima continua ad essere la più utilizzata). Oggi lo standard SNMP è supportato da una grandissima quantità di dispositivi alcuni dei quali esulano dalla categoria dei componenti di rete come ad esempio alcune stampanti.

L'obiettivo di questa tesi è analizzare tutta la teoria che sta dietro a questo protocollo (nei primi capitoli) e analizzare un software open source in grado di gestire dei nodi che siano equipaggiati con agenti SNMP.

2 FUNZIONI ED UTILITÀ DEL MONITORAGGIO DELLE RETI

Per motivare il nostro studio della gestione delle reti, cominciamo con un semplice esempio in cui analizzeremo una piccola rete composta da tre router e da un certo numero di host e server.

Anche in una rete così semplice ci sono molti scenari in cui un responsabile della rete può trarre enorme profitto dall'avere gli appropriati strumenti per la sua gestione:

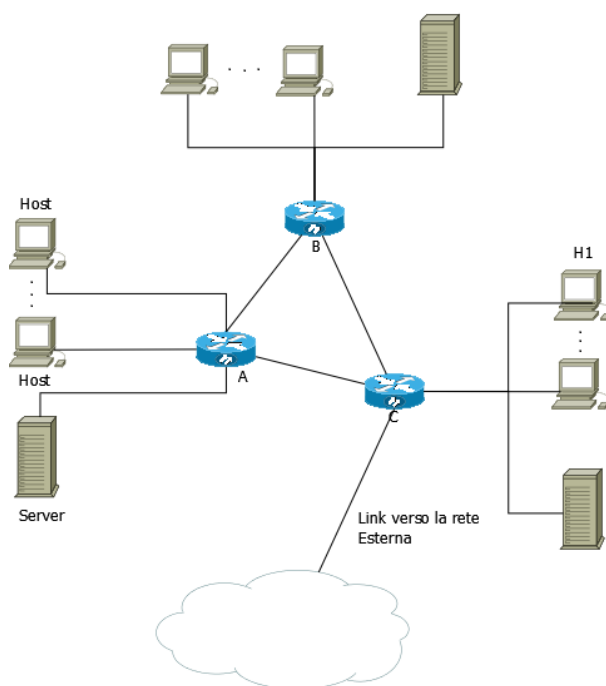


Figura 1 Schema di Rete

- *Rilevazione di un guasto di una scheda d'interfaccia a un host o a un router.* Con appropriati strumenti di gestione della rete, un'entità di rete (per esempio il router A) può segnalare al responsabile della rete che una delle sue interfacce è fuori servizio. Un responsabile che monitora e analizza attivamente il traffico sulla rete, in realtà può essere in grado di evitare le lamentele di un possibile utente rilevando in anticipo i problemi alle interfacce e sostituendo la scheda prima che si guasti. Questo può essere fatto, per esempio, se il responsabile ha notato un incremento degli errori di checksum nei frame che sono stati inviati attraverso l'interfaccia prossima al guasto.
- *Monitoraggio degli host.* Qui, il responsabile della rete può controllare periodicamente che tutti gli host della rete siano accesi e operativi. Ancora una volta, il responsabile della rete può essere in grado di anticipare la risposta a un problema (un host fuori uso) prima che il guasto sia riferito da un utente.
- *Monitoraggio del traffico per aiutare nell'utilizzo delle risorse.* Un responsabile della rete può monitorare gli schemi del traffico da sorgente a destinazione e rilevare, per esempio, che attraverso la commutazione dei server fra i segmenti LAN, la quantità di traffico che attraversa molte LAN può diminuire significativamente. Attraverso il monitoraggio dell'utilizzazione dei link, un

responsabile della rete può determinare che un segmento LAN, o il link verso il mondo esterno sia sovraccarico e che è necessario un link con una larghezza di banda superiore, che dovrà quindi essere approvvigionato (rappresentando un costo per l'azienda). Il responsabile della rete può anche desiderare la notifica automatica nel caso in cui i livelli di congestione su un link eccedano un dato valore soglia, per poter mettere a disposizione un link con larghezza di banda superiore prima che la congestione diventi seria.

- *Rilevazione rapida dei cambiamenti nelle tabelle di instradamento.* La fluttuazione dei percorsi (variazioni frequenti nelle tabelle di instradamento) possono indicare instabilità nell'instradamento o perdita di configurazione in un router. Certamente il responsabile della rete che ha impropriamente configurato un router preferirà scoprire da solo l'errore, prima che la rete si blocchi.

3 SNMP

3.1 INFORMAZIONI E CARATTERISTICHE GENERALI SUL PROTOCOLLO

Il protocollo SNMP (Simple Network Management Protocol) ufficialmente nasce nel 1989 e viene definito dalla Internet Engineering Task Force (IETF). Da quel momento SNMP diventa uno standard industriale per controllare gli apparati di rete tramite un'unica applicazione di controllo. SNMP rappresenta una serie di funzioni e protocolli per la gestione di rete che comunicano tra di loro attraverso l'Internet Protocol (IP), infatti la prima implementazione avviene su protocollo TCP/IP, ma in seguito verrà sviluppato anche su reti IPX e AppleTalk. Questo protocollo permette agli amministratori di rete di individuare ed inseguito isolare i componenti difettosi che si possono trovare su una rete, configurare i vari componenti in remoto e monitorare lo stato e le performance della rete. SNMP opera allo strato applicativo del livello OSI (livello 7) e utilizza un'architettura di comunicazione di tipo client-server con il protocollo UDP (User Datagram Protocol) sfruttando di default la porta 161.

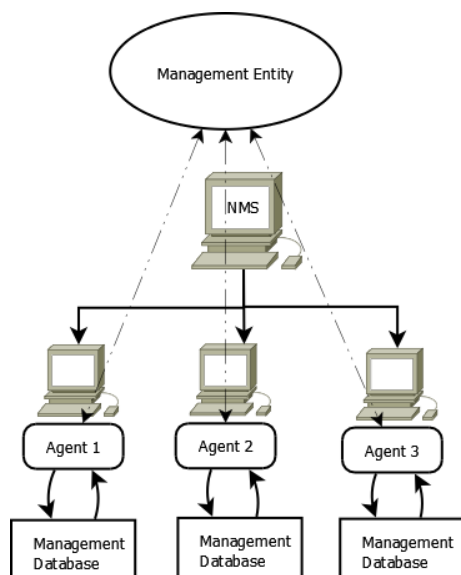


Figura II Implementazione Client Server

SNMP si costituisce di quattro parti fondamentali:

- **Sistema di Gestione da remoto(Manager):** è l'applicazione remota che prende le decisioni di gestione, per esempio sotto il controllo diretto dell'operatore umano(amministratore). Viene installato su un computer della rete per essere utilizzato come stazione di controllo mettendo in comunicazione diretta l'amministratore di rete e il sistema da gestire. Il manager dialoga con i sistemi gestiti essenzialmente in due modi: invia richieste SNMP e riceve notifiche SNMP.
- **Agente di Gestione (Agent):** I software che risiedono su device di rete (router, switch, workstation, stampanti, etc.) e segnalano svariate informazioni come gli indirizzi fisici, il carico di lavoro e altri dettagli tecnici utili all'amministratore.

Questi dati vengono poi salvati all'interno di un database, il Management Information Base (MIB).

- **Management Information Base:** Il MIB costituisce l'insieme delle informazioni effettivamente recuperabili dal Sistema di Gestione sul dispositivo da monitorare. Nello specifico il MIB è definito come il database nel quale vengono risolte tutte le richieste fatte dal manager. Tale database ha una struttura gerarchica ad albero.
- **Protocollo per la gestione:** Consente al Sistema di Gestione di recuperare i valori delle variabili MIB grazie al comando GET e all'Agent di segnalare la presenza di particolari eventi al manager grazie al comando TRAP.

Tale sistema permette di interrogare i diversi segmenti di rete creando delle connessioni virtuali tra il Network Manager e l'Agent SNMP, presente sul dispositivo remoto e comunicando informazioni o eventuali segnalazioni di errore.

Uno dei punti di forza del Protocollo in questione è la sua incredibile diffusione e la capacità di adattarsi a qualsiasi dispositivo che faccia parte di una rete di computer, infatti gli agenti SNMP si possono trovare su computer, bridge di rete, switch, router, modem e anche stampanti. Il motivo per cui SNMP è nato e per il quale tuttora è così diffuso è la sua interoperabilità. In più questo protocollo è flessibile ed estensibile in base alle necessità che si presentano. Siccome le funzioni degli agenti SNMP possono essere facilmente estese, per soddisfare le specifiche di ogni componente hardware, e soprattutto esiste un meccanismo abbastanza semplice per sviluppare le applicazioni software che poi dovranno interfacciarsi con certe tipologie di agenti, SNMP dispone un grande numero di specifiche per la gestione non strettamente legate alla gestione di apparati di rete, ma anche per esempio per la gestione di una stampante.

Dopo aver parlato dei motivi che hanno reso famoso questo protocollo, ora illustriamo anche i suoi punti deboli. Innanzitutto a discapito del nome Simple Network Management Protocol, SNMP è un protocollo molto complicato da implementare, per stessa ammissione dei progettisti, un nome più appropriato sarebbe stato Moderate Network Management Protocol, ma anche questa definizione potrebbe sembrare alquanto generosa se si pensa alla complessità delle regole che codificano questo protocollo. Un altro punto debole è l'efficienza del protocollo; infatti molta banda utilizzata viene in realtà sprecata con informazioni inutili come per esempio la versione del protocollo che viene trasmessa in tutti i pacchetti o altre informazioni sui data descriptors inserite in ogni pacchetto. Il modo con cui il protocollo identifica le variabili (come le stringhe di byte, dove ogni byte corrisponde a un particolare nodo in una database MIB) comporta un inutile spreco di buona parte del messaggio.

Sebbene anche questo protocollo sia oggetto di critiche e non privo di imperfezioni, si può comunque dire che per quanto riguarda la complessità delle sue regole, il problema è esclusivamente dei programmatori in quanto l'utente finale non sarà mai in grado di capire a fondo la complessità degli algoritmi con i suoi dati vengono trattati. Invece per quanto riguarda l'efficienza e l'occupazione di banda possiamo dire che lo sviluppo delle tecnologie di comunicazione può nascondere parzialmente il fatto che molte informazioni che viaggiano in pacchetti SNMP sono in sostanza inutili.

3.2 EVOLUZIONE DEL PROTOCOLLO

In questo paragrafo verrà fatta una panoramica abbastanza rapida sull'evoluzione del protocollo SNMP.

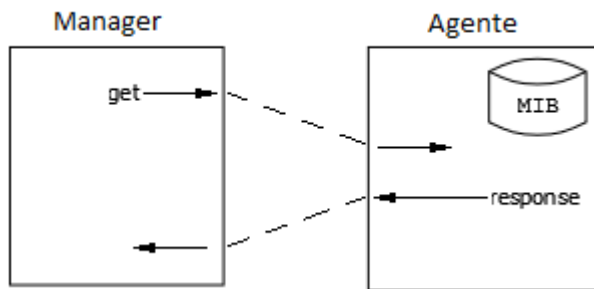
3.2.1 SNMP v1

Le caratteristiche principali del protocollo che nascono e si mantengono tali anche dopo la realizzazione delle versioni successive sono:

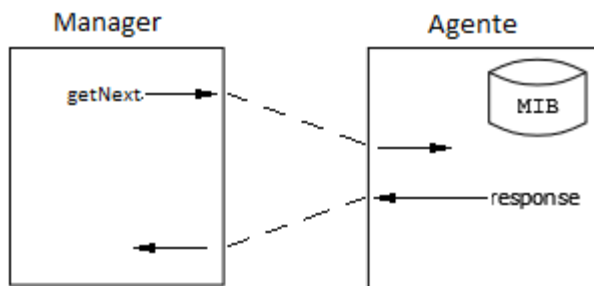
- I moduli Agent sono in ascolto sulla porta UDP 161;
- Le risposte sono inviate alla Stazione di Gestione (Manager) utilizzando un numero di porta casuale;
- La dimensione massima del pacchetto SNMP è limitata solamente dalla massima dimensione del payload UDP (65507 byte);
- I messaggi di errore e le eccezioni (Trap) sono spediti dall'Agent al Manager in maniera asincrona utilizzando la porta UDP 162.

Le principali operazioni del protocollo SNMPv1 sono:

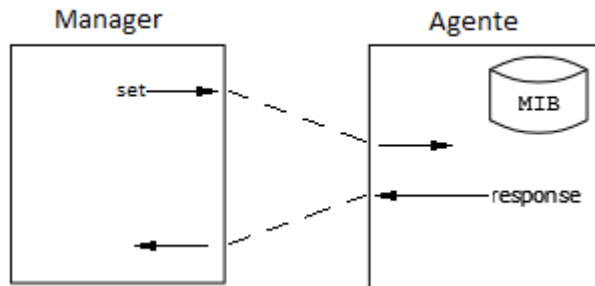
- **Get:** utilizzata dal Manager per reperire un valore dal MIB dell'Agent.



- **Get-Next:** Utilizzata dal Manager per accedere ricorsivamente sul MIB.



- **Set:** utilizzata dal Manager per impostare un valore sul MIB.



- **Trap:** Utilizzata dall'Agente per inviare messaggi di errore al Manager.

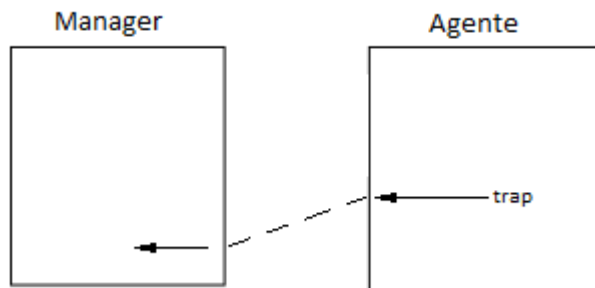


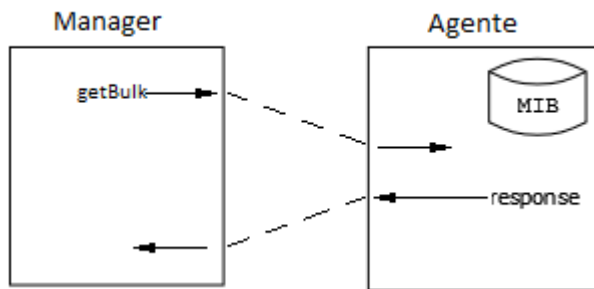
Figura III Scambio di Messaggi SNMPv1

Il protocollo assume che i canali di comunicazione siano connection-less, quindi utilizza come protocollo di livello Transport, il protocollo UDP. Di conseguenza, SNMP non garantisce l'affidabilità dei pacchetti SNMP.

3.2.2 SNMPv2

Nella versione 2 del protocollo non sono state apportate modifiche sostanziali, sebbene la versione precedente del protocollo avesse molte limitazioni: presenza di regole non documentate; codici di errori limitati; tipi di dato limitati; scarse prestazioni; dipendenza dal protocollo di trasporto; assenza di gerarchia nell'architettura Manager/Agent; scarsa sicurezza. Possiamo sintetizzare le modifiche principali mostrando le due funzioni che sono state aggiunte:

- **GetBulk:** Utilizzata dal Manager per recuperare grandi blocchi di data.



- **Inform:** In questo caso è l'agente che interroga il Manager per ottenere un'informazione.

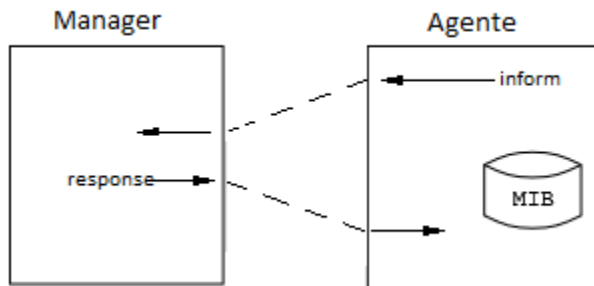


Figura IV Scambio di Messaggi SNMPv2

3.2.3 SNMPv3

A partire dalla seconda metà del 1999 è disponibile una ulteriore versione del protocollo SNMP, ovvero la terza versione. Poiché le differenze con le precedenti sono notevoli, si analizzeranno le caratteristiche maggiormente innovative. Nasce per sopperire alle mancanze dei suoi predecessori nell'ambito della sicurezza delle trasmissioni. Questo protocollo è stato pensato, inoltre, per essere scalabile, duraturo, per quanto riguarda l'architettura, portabile, compatibile con le precedenti versioni (usa gli stessi MIB).

Nonostante ciò, la terza versione non ha, almeno per ora, trovato grosso spazio sul mercato, dove continua ad essere utilizzata maggiormente la prima, forse anche perché, nonostante fosse fra gli obiettivi di questa nuova versione, la maggiorazione del numero delle caratteristiche è andato a discapito della semplicità del protocollo.

La classica architettura di tipo Manager/Agent, nella versione 3, è stata sostituita da una più complessa composta da Motore ed Applicazioni. Infatti, un'entità SNMPv3 è composta da:

- **SNMP-Engine (Motore):** Contiene un Dispatcher (smistatore di messaggi), un sottosistema per elaborare i messaggi, uno per la sicurezza e uno per il controllo dell'accesso.
- **SNMP-Applications (Applicazione):** Contiene un generatore di comandi, un ricettatore di notifiche, un risponditore ai comandi e altre funzioni.

Il formato dei messaggi di SNMPv3 è sostanzialmente diverso da quello delle precedenti versioni; infatti include anche alcuni parametri di sicurezza ed il controllo dell'accesso. Tramite appropriate politiche di sicurezza, SNMP versione 3 consente di accettare i messaggi solo nel caso in cui alcune domande ricevano una risposta affermativa o valida. Ad esempio:

- Messaggio autentico?
- Chi intende eseguire una certa operazione? (Usa l'autenticazione con chiavi di crittografia pubbliche e private)
- Quali sono gli oggetti coinvolti dall'operazione?
- Quali diritti di accesso ha il richiedente sull'oggetto al quale vuole accedere?

Queste politiche di sicurezza sono implementate tramite crittografia, funzioni di hash e altri strumenti che consentono l'autenticazione dei pacchetti (ad esempio contro un attacco di sniffing e ripetizione di pacchetto), delle password e, anche, delle PDU (le quali possono essere codificate). Tramite diversi livelli di sicurezza si può stabilire se consentire un accesso:

- Senza autenticazione
- Con autenticazione
- Con autenticazione e codifica dei dati.

3.3 SNMP NEL DETTAGLIO

Si andrà ora ad affrontare un'analisi dell'SNMP nello specifico. Si può vedere il Protocollo come diviso in tre standard distinti:

1. **Formato dei Messaggi:** Il protocollo è uno standard di comunicazione che definisce dei messaggi in formato UDP. Questa parte dello standard ha subito una notevole involuzione che non produce quasi nessuna conseguenza per l'utente, ma suscita grande interesse per il programmatore.
2. **Set di Oggetti:** Il set di Oggetti in questione non è altro che un insieme di valori (object), che possono essere richiesti ad un dispositivo. Ci sono, in questo set, dei valori utili al monitoraggio TCP, IP, UDP, etc. Ogni oggetto è identificato da un nome ufficiale e con un identificatore numerico che è espresso in denominazione puntata (es. 1.2.1.3.12).
3. **Metodo standard per la creazione di un oggetto:** E' possibile estendere il set degli oggetti definendone nuovi ad-hoc per il proprio hardware in modo da poter personalizzare i componenti prodotti.

3.3.1 Formato dei Messaggi

Possono essere definite cinque tipologie di messaggi SNMP che sono: la richiesta "get" che come valore di risposta riceve il nome dell'oggetto interrogato; la richiesta "get-next" richiede un altro nome o un valore di un oggetto che si trova su un altro dispositivo, che abbia un nome SNMP valido; il comando "response" viene generato dal dispositivo agente e serve ad inviare i dati che sono stati richiesti dagli altri comandi; il comando "trap" viene generato anch'esso dal dispositivo agente in maniera asincrona quando deve segnalare o notificare un evento speciale al network manager.

Tutti questi messaggi viaggiano sulla rete incapsulati in PDUs (Protocol Data Unit) e lo scambio di questi tra i dispositivi avviene tramite protocollo SNMP.

Si andranno ora ad analizzare nello specifico ciascun messaggio, partendo dalla prima versione di SNMP:

- **Get:** l'amministratore può richiedere valori specifici tramite il comando get per determinare le prestazioni e le condizioni di funzionamento del dispositivo. Molti di questi valori possono essere determinati esclusivamente analizzando i messaggi generati dal protocollo SNMP stabilendo appositamente una connessione TCP.
- **Get Next:** Questo comando viene utilizzato dall'amministratore per "navigare" sulla rete alla ricerca di tutti i dispositivi che supportano il protocollo SNMP. Questa operazione di ricerca parte dal manager di rete e viene reiterata da ogni nodo SNMP che incontra, sempre attraverso lo stesso comando, fino a quando non viene riscontrato qualche errore. In tal caso il manager è in grado di mappare tutti i nodi SNMP della rete.
- **Set:** Questo comando mette a disposizione del manager un metodo per effettuare delle operazioni associate al dispositivo di rete come ad esempio stabilire l'interfaccia, disconnettere gli utenti, pulire i registri, etc. In sostanza il "set" permette di configurare e controllare in modo remoto il dispositivo tramite SNMP.
- **Get Response:** Questo comando viene utilizzato dal device di rete per rispondere alle richieste che gli vengono inoltrate tramite get, get-next e set.
- **Trap:** Consiste in un meccanismo attraverso il quale i dispositivi di rete possono mandare delle comunicazioni sul loro stato ai network manager. Generalmente

questa funzione viene utilizzata per notificare degli errori. Per ricevere i pacchetti trap di solito è necessario configurare i vari dispositivi di rete in modo che questi restino in attesa della ricezione.

Queste tipologie di messaggi, come detto in precedenza, appartengono alla prima versione del protocollo SNMP. Di seguito si elencano i nuovi comandi introdotti con la versione 2:

- **Get Bulk:** Questo comando serve ad accumulare in un'unica transazione request/response molte informazioni relative ad un dispositivo. In pratica il get-bulk si comporta come una serie di get next, eccetto nel caso in cui sia sufficiente una singola interazione.
- **Trap v2:** Nella seconda versione è stato introdotto un tipo di messaggio trap che ha caratteristiche quasi identiche alla versione precedente, ma con qualche differenza.
- **Inform:** Questo comando non comunica valori nuovi, ma ha solo la funzione di confermare la notifica di certi eventi al Manager.

3.3.2 Oggetti SNMP

La lista dei valori che un oggetto può supportare è spesso chiamata SNMP MIB (Management Information Base). Il MIB è un'astrazione come Database a cui possono essere attribuiti molteplici significati.

La grande varietà di valori SNMP nello standard MIB sono definiti nel RFC 1231. Lo standard MIB include molti oggetti (o valori) per misurare e monitorare le attività IP, TCP, UDP, le connessioni TCP, le interfacce, il sistema in generale, etc. Tutti questi valori sono associati ad un nome ufficiale (come ad esempio sysUpTime, che misura da quanto tempo è acceso il device dall'ultimo avvio) e un valore numerico ufficiale espresso in notazione puntata (il numero identificativo del sysUpTime è 1.3.6.1.2.1.1.3).

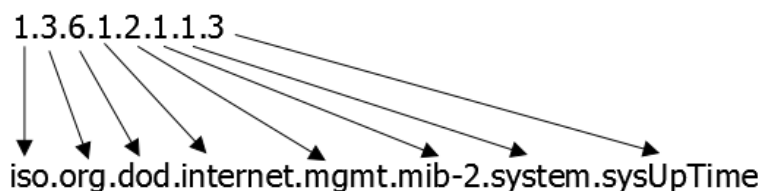


Figura V OID sysUpTime

3.4 MIB

Il database che viene gestito dall'agente SNMP è più comunemente conosciuto come MIB (Management Information Base) ed è una raccolta di valori statistici e di controllo riferiti al dispositivo. SNMP permette di estendere questi valori standard con valori specifici per particolari necessità di un agente o di un utente sempre attraverso l'utilizzo dei MIBs.

Per usare efficacemente SNMP, gli utenti devono conoscere SNMP MIB, che definisce tutti i valori che il protocollo è in grado di leggere o di settare. Per diventare esperto in SNMP, un manager network deve per forza approfondire la sua conoscenza del MIB.

SNMP MIB è organizzato con una struttura ad albero, molto simile a quella usata dai PC per organizzare i files in directory come si vede in Figura.

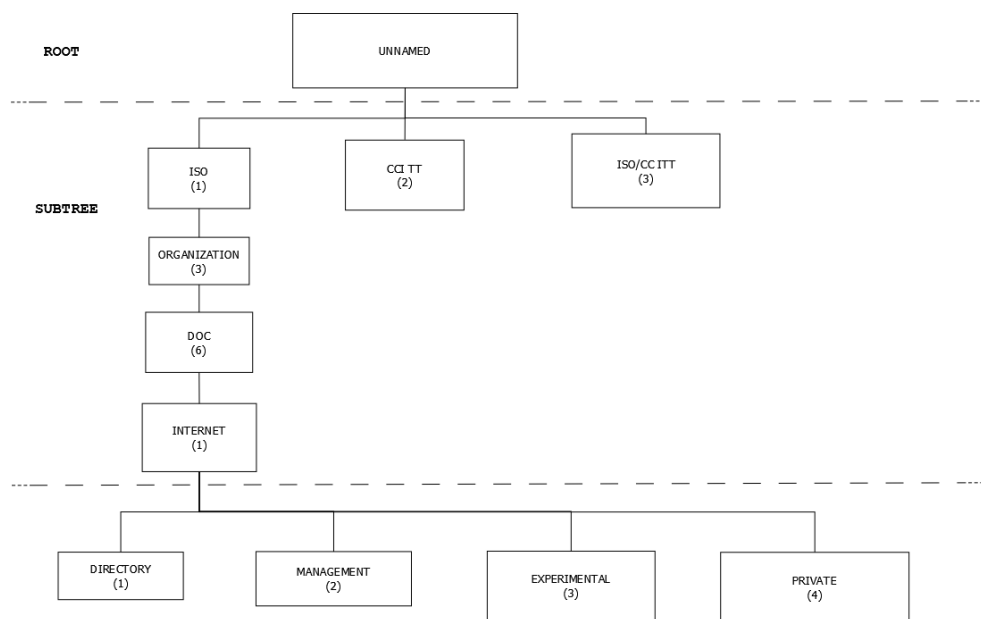


Figura VI Albero MIB - Root e SubTree

Come si evince dalla figura, la cartella radice (ROOT) non ha home, poi seguono le tre cartelle principali in cui sono contenuti tutti i sottoalberi che contengono le definizioni di tutti gli standard tra cui anche Internet che è contenuto in una sottocartella di ISO (International Organization for Standardization).

L'insieme di standardizzazioni che riguardano Internet possono essere suddivise in quattro rami principali (come si vede in Figura).

- I rami "Directory" e "Experimental" sono sostanzialmente privi di valori e oggetti che abbiano un significato rilevante.
- Il ramo "Management" contiene tutti gli standard degli oggetti supportati da tutti i dispositivi della rete.
- Il ramo "Private" contiene le definizioni degli standard per gli oggetti SNMP creati dai vari produttori e implementati sui propri dispositivi.

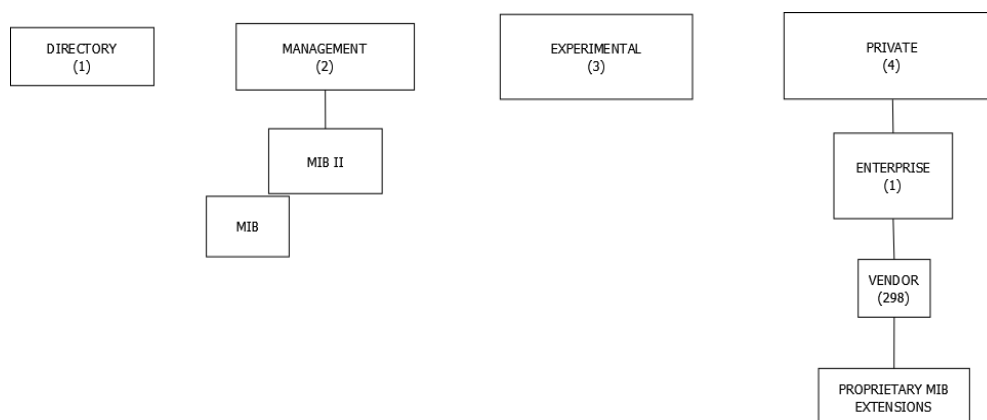


Figura VII Albero MIB – Leaf

Generalmente, gli oggetti SNMP, possono essere divisi in due categorie simili ma con piccole sostanziali differenze che riflettono l'organizzazione in una struttura ad albero:

1. **Discrete MIB Objects:** Gli oggetti "discreti" SNMP contengono solamente una parte precisa e ben definita dai Table Objects aggiungendo un'estensione ".0" ai loro nomi (se viene omessa da un nome di un oggetto SNMP, è sempre considerata implicita).
2. **Table MIB Objects:** Gli oggetti SNMP definiti "table", ovvero tabella, contengono parti multiple di dati o valori per la gestione. Questa categoria di oggetti si distingue in "discrete" aggiungendo "." ai loro nomi seguito da un numero che distingue univocamente il valore particolare a cui si fa riferimento.

3.5 APPLICATIVI E TOOL GRAFICI

Esistono sul mercato molte implementazioni e applicativi che si basano su SNMP. Ognuna ha le sue peculiarità che la differenzia, in un modo o in un altro, dalle altre. Ci sono prodotti che si concentrano sul monitoraggio dell'infrastruttura di rete ed offrono un supporto elevato per il protocollo. Altri applicativi invece sono stati sviluppati con il solo compito di controllare particolari componenti, sia hardware che software, di una rete, o a raggiungere determinati obiettivi.

Esistono poi dei software per il monitoraggio di ambienti eterogenei, quindi che lavorano su diverse piattaforme, quali Linux, Windows, etc., e che si rendono utili per un controllo quanto più completo di tutta l'infrastruttura web. Sono, infatti, i programmi che più spesso vengono utilizzati dagli amministratori di rete.

Si andranno ora ad esaminare alcuni dei software open-source più diffusi, andando più nello specifico per quanto riguarda il Software Zabbix.

3.5.1 Net-SNMP

E' una suite di applicativi per utilizzare e sviluppare il protocollo SNMP. L'utilizzo è macchinoso, in quanto necessita del Terminale.

3.5.1.1 *SNMPwalk*:

Uno degli applicativi contenuti all'interno della suite è *SNMPwalk*. Quest'ultimo recupera una sottostruttura di valori di gestione utilizzando richieste SNMP GetNext per interrogare un'entità di rete per un albero di informazione.

Sintassi:

```
snmpwalk [opzioni applicazione] [opzioni comuni] [OID]
```

Si può fornire un OID (identificato di oggetto) tramite riga di comando. Esso specifica quale porzione di identificatore verrà ricercata utilizzando richieste GetNext. Tutte le variabili nella sottostruttura verranno interrogate e i loro valori presentati all'utente.

Se non si fornisce un OID, *snmpwalk* cercherà la sottostruttura con radice SNMPv2-SMI::mib-2 (compresi gli eventuali valori degli oggetti MIB da altri moduli MIB). Se l'entità riscontra un errore durante l'elaborazione del pacchetto richiesto, restituirà un pacchetto d'errore e mostrerà a schermo un messaggio di errore, che aiuterà l'amministratore ad individuare il motivo per cui la richiesta non era valida.

3.5.2 SNMP MIB Browser

Si tratta di un Tool grafico per il monitoraggio di device SNMP nella rete. Si possono effettuare richieste Get, GetNext e Set, osservare i moduli MIB, etc. Ha pieno supporto per tutte e tre le versioni di SNMP ed offre anche supporto agli "allarmi" all'amministratore. Si tratta un tool tanto semplice quanto poco completo. Uno dei vantaggi principali del programma è l'interfaccia utente, che rende lo studio delle reti più intuitivo, anche se non al livello di software più avanzati. Altro vantaggio del programma è la portabilità. Girerà, infatti, su tutti i sistemi operativi più diffusi.

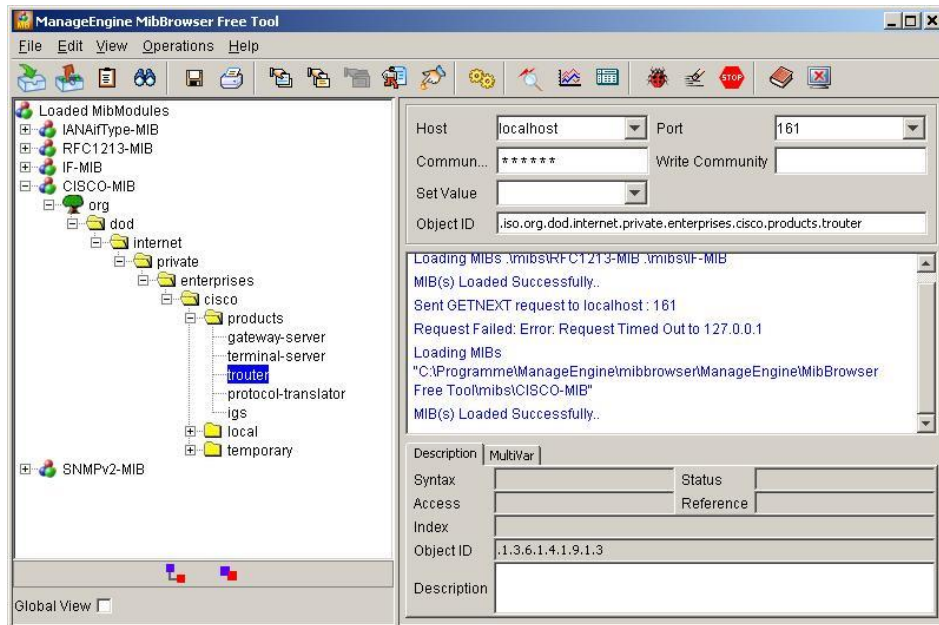


Figura VIII SNMP MIB Browser

3.5.3 OpenNMS

OpenNMS è un software open-source per il monitoraggio di reti e la loro gestione. E' sviluppato e supportato dalla community degli user e da OpenNMS Group. L'obiettivo del prodotto è essere quanto più completo possibile, pur rimanendo gratuito e open-source. E' scritto in java, perciò può girare su qualsiasi piattaforma che supporti una versione di SDK Java pari o superiore alla 1.6. In aggiunta a Java richiede PostgreSQL come database per la raccolta dei dati. Fra le sue caratteristiche, presenta attitudine per il monitoraggio remoto, in modalità passiva e supporta l'estensione delle proprie funzionalità tramite plugin. Si tratta comunque di un prodotto estremamente potente per il monitoraggio e la gestione delle componenti di rete, sotto standard SNMP.

3.5.4 Ganglia

Si tratta di un'applicazione che è stata sviluppata specificatamente per il monitoraggio in ambiente cluster. L'architettura software consiste in un agente che viene installato in ogni computer componente il cluster. Le informazioni raccolte dall'agente vengono inviate ad un processo daemon in esecuzione sul server deputato alla raccolta e memorizzazione dei dati. L'applicazione permette un efficiente scambio di informazioni fra agent e semplicità di installazione e configurazione. Tuttavia rimane un software limitato, sia nell'estensibilità che nelle sue capacità. Ciò compromette la raccolta di informazioni dettagliate dell'infrastruttura, rimanendo quindi un software utile solo per determinati ambienti.

3.5.5 Nagios

Nagios è un sistema di monitoraggio flessibile che permette alle aziende di identificare e risolvere problemi di infrastruttura IT prima che possano compromettere processi di business.

Consente di rilevare e prevenire e correggere i problemi prima che danneggino gli utenti finali e clienti.

N.A.G.I.O.S. è un acronimo ricorsivo di "Nagios Ain't Gonna Insist On Sainthood". È un riferimento al nome originale del software, NetSaint, che fu cambiato per via di problemi di marchi. Questo applicativo si caratterizza per la sua particolare architettura: si tratta sostanzialmente di un framework su cui si innestano moduli aggiuntivi (plugin ed extension) che consentono di realizzare soluzioni efficienti e molto flessibili, in grado di soddisfare le più varie esigenze. Per contro presenta un carico di rete superiore a quello di altri prodotti ed una modalità di gestione dei task che appesantisce il lavoro del server utilizzato per il monitoraggio. La possibilità di aggiungere plugin, grazie alla diffusione "open" delle API per la definizione dei plugin, unitamente alla ampia e valida documentazione disponibile sul prodotto, ha fatto nascere comunità Internet in cui vengono resi disponibili moduli aggiuntivi per virtualmente qualunque esigenza.

3.5.6 Tabella Riassuntiva Software di Monitoraggio

Name	IP Report	SLA	Logical Grouping	Trending	Trend Prediction	Auto Discovery	Agentless	SNMP	Plugins	Triggers/Alerts	WebApp	Distributed Monitoring	Platform	Data Storage Method	Licenza	Costo
Nagios	Via Plugin		Si	Si	No	Via Plugin	Supportato	Via Plugin	Si	Si	Full Control	Si	C, PHP	Flat File, SQL	GNU GPL	Gratis / A Pagamento
Zabbix	Si		Si	Si	No	Si	Supportato	Si	Si	Si	Full Control	Si	C, PHP	Oracle, MySQL, PostgreSQL, IBM DB2, SQLite	GNU GPL (Free Software)	Gratis
Ganglia	No		Si	Si	No	Via Gmond Check In	No	Via Plugin	Si	No	Viewing	Si	C, PHP	RRDtool	BSD	Gratis
OpenNMS	Si		Si	Si	No	Si	Supportato	Si	Si	Si	Full Control	Si	Java	Jrobin, PostgreSQL	GPLv3	Gratis

Legenda:

- **IP SLA Reports:** Supporto all' IP Service Level Agreement di Cisco.
- **Logical Grouping:** Supporto all'organizzazione degli host o dei device monitorati in gruppi definiti dall'utente.
- **Trending:** Fornisce l'andamento dei dati di rete nel tempo.
- **Trend Prediction:** Il software utilizza algoritmi per predire statistiche di rete future.
- **Auto Discovery:** Il software scopre automaticamente host o device di rete.
- **Agentless:** Non si basa su un software agent che deve essere eseguito sugli host che si vogliono monitorare, in modo che i dati possano essere poi rinviati al server centrale. "Supportato" significa che un agent può essere utilizzato, ma non è obbligatorio per il funzionamento del software.
- **SNMP:** Il software è capace di recuperare e realizzare le statistiche SNMP.
- **Plugins:** Possibilità di estendere le funzionalità del software con plugins.
- **Triggers/Alerts:** Capace di individuare violazioni Threshold nei dati della rete e allertare l'amministratore in diversi modi.
- **WebApp:** Gira su un'applicazione Web:
 - Viewing: I dati di rete possono essere visualizzati in un'interfaccia grafica Web.
 - Full Control: Tutti gli aspetti del programma possono essere controllati attraverso un frontend Web, inclusa la manutenzione a basso livello, come la configurazione del software e gli upgrade.
- **Distributed Monitoring:** Può sfruttare più di un server per distribuire il carico di monitoraggio della rete.
- **Data Storage Method:** Il Database utilizzato per fare lo store dei dati monitorati della rete.
- **Licenza:** Sotto quale licenza si trova il software.
- **Costo:** Indica se il prodotto è fornito gratuitamente o a pagamento:
 - Gratis/A Pagamento: Il software ha due versioni. Una a pagamento, una gratuita.
 - Gratis: Il software viene fornito gratuitamente ed accessibile a tutti

4 ZABBIX: MONITORING TOOL

4.1 COSA È ZABBIX

Zabbix è una soluzione open source per il monitoraggio delle reti. E' un software che monitora numerosi parametri di una rete e lo stato e l'integrità dei server. Zabbix utilizza un meccanismo flessibile di notifiche che permettono all'utente di configurare allarmi basate sulle e-mail virtualmente per qualsiasi evento. Questo permette di reagire prontamente alle problematiche dei server. Zabbix offre una funzionalità di visualizzazione dei dati in base ai dati già memorizzati.

Tutte le statistiche e i rapporti di Zabbix, così come i parametri di configurazione, sono accessibili tramite un front-end web-based. Quest'ultimo assicura che lo stato della rete e la sua salute dei server possano essere valutati e controllati da qualsiasi luogo.

4.2 FUNZIONALITÀ GENERALI

Zabbix è, quindi, una soluzione altamente integrata per il monitoraggio dei network, che offre molteplici features in un singolo pacchetto. Di seguito si elencano alcune delle funzionalità presenti nel software:

Recupero Dati

- Check per testare le performance e la disponibilità.
- Supporto al protocollo SNMP (sia via trapping che polling), IPMI e JMX.
- Check personalizzati.
- Recupero dei dati desiderati ad intervalli personalizzati.
- Recupero dati via server/proxy e via agent.

Definizioni flessibili dei Threshold

- Si possono definire soglie molto flessibili, chiamati Triggers, che facciano riferimento ai valori del database.

Alerting Configurabile

- L'invio di notifiche può essere personalizzato tramite pianificazioni, destinatari e tipo di supporto richiesto.
- Le notifiche possono essere rese molto significative e di aiuto grazie all'uso di variabili Macro.
- Possibilità di configurare azioni automatiche, includendo l'esecuzione di comandi da remoto.

Grafici Real-Time

- Gli oggetti analizzati possono essere utilizzati per realizzare grafici real-time dei dati ricevuti.

Capacità di Monitoring del Web

- Zabbix può seguire un percorso simulato di “click” del mouse su di un sito Web per verificare il funzionamento e le performance dello stesso.

Opzioni di Visualizzazione

- Possibilità di creare grafici personalizzati che combinino molteplici dati in una singola schermata.
- Visualizzazione della Mappa della Rete.
- Screen personalizzabili per la Dashboard e per l’Overview degli elementi principali.
- Vista ad alto livello delle risorse monitorate.

Storage Cronologico dei Dati

- Dati salvati in un Database.
- Cronologia configurabile.
- Procedura di pulizia dati integrata.

Uso dei Template

- Ereditarietà dei Template.

Network Discovery

- Scoperta automatica dei device nella rete.
- Auto registrazione degli Agent.
- Possibilità di scoperta di File Systems, interfacce di rete e OID SNMP.

Interfaccia Frontend Web

- Frontend basato su Web in PHP.
- Accessibile ovunque.
- Registro di Controllo.

Zabbix API

- Le API di Zabbix permettono di programmare interfacce per il software altamente manipolabili, permettono di integrare software di terze parti ed altre funzionalità.

Sistema dei Permessi d’Accesso

- Autenticazione sicura degli Utenti.
- Possibilità di limitare per alcuni Utenti l’uso di determinati ambienti del Frontend.

Agent Configurabili

- Possono essere configurati ed installati su diverse macchine e diversi SO (Linux e Windows).

4.3 ARCHITETTURA

Zabbix si compone di diverse componenti maggiori, le cui responsabilità sono esplicitate di seguito:

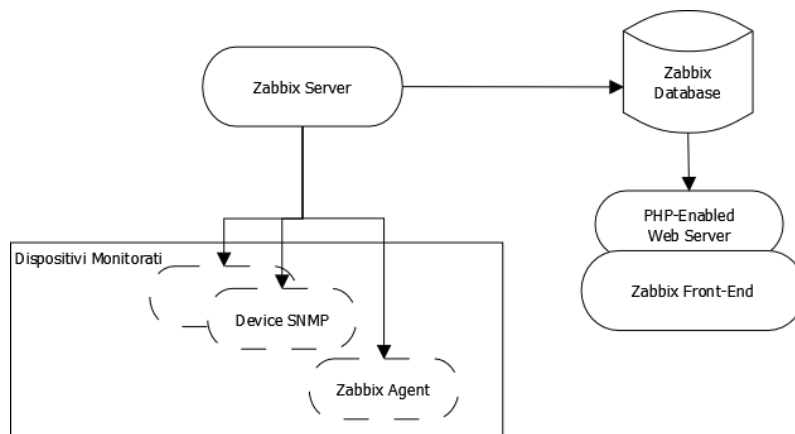


Figura IX Architettura Zabbix

Server

Lo "Zabbix Server" è la componente principale alla quale, gli agent presenti nelle macchine monitorate, riportano la loro integrità, le loro informazioni e statistiche. Il server è la repository principale nella quale vengono salvate tutte le configurazioni e tutti i dati statistici e operativi.

Database

Tutte le informazioni di configurazione, così come i dati acquisiti da Zabbix, vengono salvati in un Database.

Interfaccia Web

Per un utilizzo semplificato di Zabbix da ogni luogo e piattaforma, viene fornita un'interfaccia web. Essa è parte del Zabbix Server e di solito (ma non obbligatoriamente) viene eseguita dalla stessa macchina fisica sulla quale è installata il comparto server del software.

Proxy

Lo Zabbix Proxy può raccogliere performance e disponibilità dei dati per conto del server di Zabbix. Un proxy è una componente opzionale; tuttavia, può essere utile per distribuire il carico delle operazioni da svolgere.

Agent

Gli Zabbix Agent vengono implementati nelle macchine da monitorare per controllare risorse locali e applicativi e riportare i dati raccolti al Server Zabbix.

4.4 DATA FLOW

Al fine di creare un Item che raccolga dati, è necessario creare prima un Host da monitorare. Occorre avere un Item per poter creare un Trigger ed occorre avere un Trigger per poter creare una Action. Pertanto, se si desidera ricevere un avviso che il carico della CPU è troppo alto su di un Server X è innanzitutto necessario creare una voce Host per il Server X seguita da un Item per il controllo della CPU, quindi un Trigger che si attivi se l'uso della CPU è troppo alto, seguito da un'azione che invii una mail. Nonostante ci siano diversi passi da compiere per raggiungere lo scopo finale, grazie a questa strategia si può avere una configurazione molto flessibile. Inoltre si può semplificare di molto la procedura con l'utilizzo e la creazione di Template.

4.5 CONCETTI BASE NECESSARI

In questa sezione andremo ad analizzare i termini comunemente usati in Zabbix.

Definizioni

Host

Un device della rete che si vuole analizzare, tramite IP/DNS.

Host Group

Un raggruppamento logico di Host; Può contenere Host e Template. Host e Template all'interno di un gruppo non sono in alcun modo legati fra loro. I gruppi vengono usati per l'assegnazione dei diritti di accesso per diversi gruppi di utenti.

Item

Una precisa parte di dato che si vuole ricavare da un Host.

Trigger

Un'espressione logica che definisce la soglia di un problema e usata per "valutare" i dati ricevuti tramite gli Item.

Event

Una singola occorrenza di qualche avvenimento che merita attenzione, come un Trigger che cambia il suo stato o la scoperta/auto-registrazione di un Agent.

Action

Rappresenta un reagire predefinito ad un evento che occorre. Consiste di operazioni (es. Inviare una notifica) e condizioni (quando l'operazione viene effettuata).

Escalation

Uno scenario personalizzato per l'esecuzione di operazioni all'interno di una Action; una sequenza di invio di notifiche/esecuzione di comandi da remoto.

Media

Un mezzo per trasmettere notifiche; un canale di recapito.

Notification

Un messaggio riguardante un evento inviato ad un user attraverso il media prescelto.

Remote Command

Un comando predefinito che viene eseguito automaticamente su di un host monitorato in determinate condizioni.

Template

Un set di Entità (Item, Trigger, Grafici, etc.) pronti per essere eseguiti su uno o più host. Il compito di un Template è quello di velocizzare l'implementazione dei task di monitoraggio su di un host. Inoltre sono anche utili per applicare cambiamenti di massa ai task di monitoraggio. I Template sono collegati direttamente ad host individuali.

Application

Un raggruppamento di Item in un gruppo logico.

FrontEnd

L'interfaccia web provvista da Zabbix.

Zabbix API

Le API di Zabbix permettono di utilizzare il protocollo JSON RPC per creare, aggiornare e recuperare gli "oggetti Zabbix" (come host, item, grafici, etc.) o di eseguire un qualsiasi altro compito.

Zabbix Server

La parte centrale del software Zabbix, ovvero quella che si occupa del monitoraggio, quella che interagisce con Proxy e Agent, che calcola i Trigger ed invia le notifiche. La repository centrale dei dati.

Zabbix Agent

Un processo implementato in un host per monitorare attivamente le risorse e le applicazioni locali.

Zabbix Proxy

Un processo che ha la possibilità di raccogliere dati a nome del Server Zabbix, togliendogli del carico da elaborare.

Node

Un Server Zabbix pienamente configurato come un elemento all'interno di una gerarchia di monitoraggio distribuito; è responsabile del monitoraggio della propria posizione.

4.6 INSTALLAZIONE

Si andrà ora ad analizzare i procedimenti necessari per l'installazione del Server Zabbix e degli Agent nelle macchine da monitorare. Si utilizzeranno le distribuzioni Linux Ubuntu 13.10 e CentOS. Tra gli Agent vi saranno anche macchine Windows.

4.6.1 Prerequisiti

Al fine di installare Zabbix, occorre analizzare i prerequisiti:

- Una distribuzione Linux sulla quale installare il Server Zabbix.
- Una istanza funzionante LAMP
 - Apache 2
 - DataBase MySQL
 - PHP
 - Pacchetto Net-SNMP
- Sorgenti di Zabbix per l'installazione

4.6.2 Installazione e Configurazione LAMP

LAMP è una combinazione di software per lo sviluppo di applicazioni web il cui nome è l'acronimo dei componenti di base (Linux, Apache, MySQL o MariaDB, Php, Perl e/o Python). Si andrà ora ad elencare i passaggi per la corretta installazione dei pacchetti necessari. Si supponga di avere una macchina con all'interno la distro Ubuntu 13.10 di Linux. Gli step dell'installazione richiedono privilegi di root.

4.6.2.1 Apache

Apache è un web-server multi piattaforma open-source che fornisce una gamma completa di funzioni e caratteristiche, incluso il supporto per CGI e SSL. Per installare Apache, da terminale si digiti:

```
sudo apt-get update
sudo apt-get install apache2
```

Per verificare l'effettiva installazione sia avvenuta in modo corretto, si apra da browser l'indirizzo <http://indirizzo-ip-computer> o altrimenti <http://localhost>. Se tutto si è svolto correttamente vedremo un messaggio a schermo dicente "It Works!"

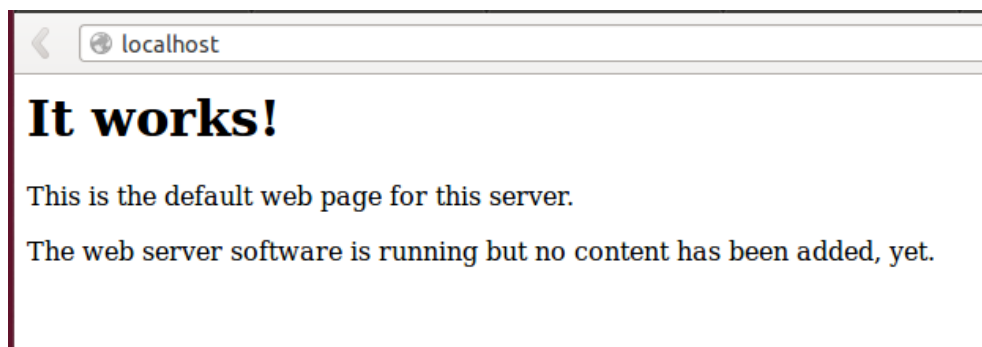


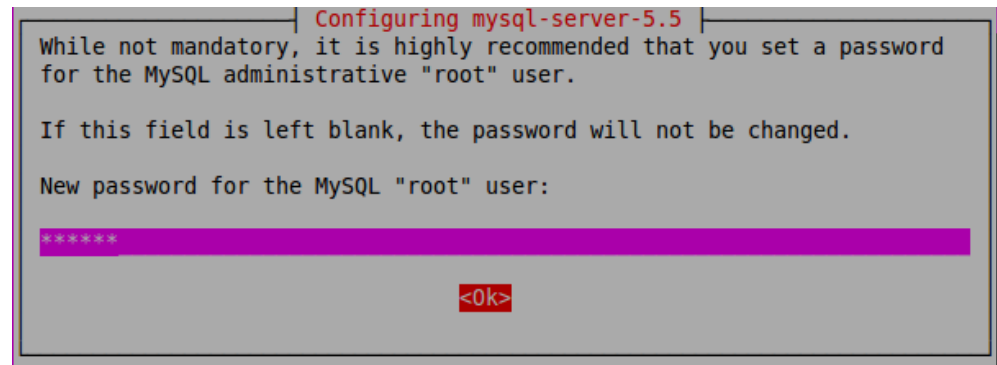
Figura X Apache Configurato

4.6.2.2 MySQL

MySQL è un sistema per la gestione di Database relazionali. Per installarlo, da terminale si digiti:

```
sudo apt-get install mysql-server
```

Durante l'installazione chiederà di impostare la password dell'utente root di MySQL: si inserisca la password e si dia Ok, dando di seguito la conferma inserendola di nuovo.



A questo punto il server MySQL è installato. Per verificare il funzionamento corretto, si digiti da terminale:

```
sudo service mysql status
```

Si otterrà una risposta simile a:

```
mysql start/running, process 1271
```

4.6.2.3 PHP

PHP (acronimo ricorsivo di "PHO: Hypertext Preprocessor", preprocessore di ipertesti) è un linguaggio di programmazione utilizzato principalmente per sviluppare applicazioni web lato server. Per installare PHP si digiti da terminale:

```
sudo apt-get install php5 libapache2-mod-php5
```

Per testare il corretto funzionamento di PHP si può agire in questo modo. Si crei un file "test.php" nella cartella root di Apache digitando da terminale:

```
sudo gedit /var/www/info.php
```

Nel file vuoto si scriva:

```
<?php  
phpinfo();  
?>
```

Si salvi e si chiuda il file, poi si riavvi Apache da terminale:

```
sudo service apache2 restart
```

Ora, tramite browser, andando all'indirizzo <http://localhost/test.php>, se l'installazione è avvenuta correttamente si potrà vedere una schermata riassuntiva.

4.6.2.4 SNMP

SNMP è un protocollo di rete. Consente la configurazione, la gestione e la supervisione di device in una rete. Occorre installare dei pacchetti necessari per il funzionamento, dando da terminale:

```
sudo apt-get install snmp libsnmp-dev snmpd
```

Si avrà ora una configurazione funzionante della LAMP. Si procederà installando il software di Zabbix, lato server.

4.6.3 Installazione e Configurazione Zabbix Server (compilato da sorgenti)

In questo capitolo si andrà ad affrontare l'installazione del lato Server di Zabbix, all'interno di una macchina Linux

1. Creazione User Zabbix

Si dovrà creare un nuovo Utente e assicurargli i permessi da admin. Da terminale:

```
sudo adduser zabbix
enter in new password
confirm
```

Ora si aggiunga l'utente creato al gruppo admin. Da terminale:

```
sudo adduser zabbix admin
```

2. Configurazione Zabbix

Si vuole ora passare alla compilazione da sorgenti di Zabbix. Occorrerà scaricare dal sito la versione più recente (qui si prenderà in considerazione la 2.0.11), e scompattarla all'interno di /home/nome-utente/

Posizionandoci da terminale all'interno della cartella appena estratta, si diano i seguenti comandi:

```
sudo ./configure --enable-server --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl
```

A seguire, si dovrà dare il comando:

```
sudo make install
```

Il prossimo passo sarà quello della configurazione di un Database, nel quale verranno immagazzinati i dati raccolti dal Server Zabbix. Sono supportati diversi tipi di Database. Qui si andrà ad utilizzare MySQL.

Per una corretta configurazione, dare i seguenti comandi:

```
sudo mysql -u root -p
create user 'zabbix'@'localhost' identified by 'Password-
Prescelta';
create database zabbix;
grant all privileges on zabbix.* to 'zabbix'@'localhost';
flush privileges;
exit;
```

Sostanzialmente, prima si andrà a creare un user, dando una password, che si dovrà annotare poiché necessaria per le configurazioni future. Successivamente si creerà il database, dando poi i permessi di controllare il database al nuovo utente appena creato. L'ultimo comando è necessario per confermare i nuovi permessi. In fine si uscirà dai comandi MySQL.

Ora si passerà alla configurazione del front-end del software. Occorrerà posizionarsi all'interno della cartella Zabbix, precedentemente estratta. Si vada al percorso:

```
cd zabbix-x.x.x/frontend/php
```

e copiarne il contenuto nella propria webroot (per Ubuntu 13.10, sarà nel percorso `/var/www/zabbix`).

Si diano i permessi al web user per accedere alla directory di zabbix. Prima di procedere con la configurazione finale, si dovranno modificare dei valori in alcuni file di sistema. Si modifichi il file `php.ini`, reperibile dal percorso `/etc/php5/apache2/php.ini` con i seguenti valori (se necessario, aggiungere i valori mancanti).

```
Post_max_size = 16M
max_execution_time = 300
max_input_time = 300
date.timezone = Italy
```

Salvare, infine, il file e riavviare apache, dando il comando:

```
sudo service apache2 restart
```

Si navighi all'indirizzo <http://proprio-indirizzo-ip/zabbix> (oppure <http://localhost/zabbix>) e si seguano le indicazioni a schermo.

Una volta terminata l'installazione, non si potrà entrare direttamente all'interno dell'interfaccia web. Occorrerà importare dei file fondamentali di MySQL. Ci si posizioni all'interno della cartella `zabbix-x.x.x/database/mysql` e si diano i seguenti comandi:

```
mysql -u zabbix -p zabbix < schema.sql
mysql -u zabbix -p zabbix < images.sql
mysql -u zabbix -p zabbix < data.sql
```

La configurazione è ora terminata. Si potrà accedere al front-end di Zabbix usando come Username: Admin e come Password: zabbix.

4.6.4 Installazione e Configurazione Zabbix Agent su varie macchine (da pacchetti)

Si passi ora alla configurazione degli Agent nelle macchine che si intende monitorare. In questo capitolo si prenderanno in considerazione, come installazione su dei Sistemi Operativi:

- Macchine Linux (Debian/apt)
- Macchine Windows (Windows Server)

4.6.4.1 Linux / Debian

Per una corretta installazione di un Agent Zabbix su Debian, basterà installare da apt i pacchetti necessari e modificare dei file php.

Come detto in precedenza, si necessita una installazione di LAMP all'interno del sistema. Si procederà dando i seguenti comandi:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install zabbix-agent
```

Successivamente sarà necessario modificare il file di configurazione. Si dia il comando:

```
sudo nano /etc/zabbix/zabbix_agentd.conf
```

All'interno del file si trovi la proprietà "Server". Essa riflette l'indirizzo IP del Server, appunto, Zabbix. Per cui si sostituirà il valore con l'indirizzo del Server. (Se l'agent che si sta installando è all'interno del Server stesso, si potrà scrivere l'indirizzo di loopback 127.0.0.1)

```
Server=Indirizzo.IP.Server.Zabbix
```

Si dovrà andare a modificare anche la proprietà "Hostname" in modo che rifletta il nome host della macchina che si vuole monitorare. Sarà necessario per la configurazione durante l'uso del front-end php, per cui si consiglia di annotare il nome.

```
Hostname=Hostname.macchina.corrente
```

Si salvi e si chiuda il file. Infine si dovrà riavviare il software dell'agent in modo che possa accogliere i nuovi dati inseriti:

```
sudo service zabbix-agent restart
```

4.6.4.2 *Macchina Windows*

Per procedere all'installazione dell'Agent Zabbix su di una macchina occorrerà fare il download dei sorgenti dal sito ufficiale. Si estragga il contenuto della cartella appena scaricata nel percorso `C:\zabbix_agent.win\`

Il passaggio successivo sarà quello di creare un file denominato `zabbix_agentd.conf` e di posizionarlo all'interno di `C:\`

All'interno di questo file si andrà a scrivere:

```
#Server = [zabbix server ip]
#Hostname=[Hostname macchina corrente]

Server=Indirizzo.IP.Server.Zabbix
Hostname=Hostname.Macchina.Corrente
```

Ora sarà necessario andare ad installare l'agent Zabbix all'interno dei servizi di Windows. Ci si posizioni all'interno della cartella estratta precedentemente, all'interno del percorso che segue:

```
C:\zabbix_agent.win\bin\win64>
```

Qui si dia il comando per l'installazione:

```
C:\zabbix_agent.win\bin\win64> zabbix_agentd.exe -install

zabbix_agentd.exe [3292]: service [Zabbix Agent] installed
successfully
zabbix_agentd.exe [3292]: event source [Zabbix Agent]
installed successfully
```

Il Prompt confermerà l'avvenuta installazione dell'Agent. A questo punto si dovrà far partire lo stesso, dando il comando:

```
C:\zabbix_agent.win\bin\win64> zabbix_agentd.exe -start

zabbix_agentd.exe [5048]: service [Zabbix Agent] started
successfully
```

L'Host è ora pronto per il monitoraggio.

4.7 ANALISI INTERFACCIA UTENTE

Si andrà ora a studiare l'interfaccia utente di Zabbix. Come descritto in precedenza, il front-end è in PHP.

4.7.1 DashBoard

Nella schermata iniziale, troveremo la Dashboard. Essa fornisce una visione generale della situazione della rete monitorata. Eventuali problematiche saranno immediatamente visibili all'amministratore, che potrà procedere allo studio e all'acknowledgment degli eventi.

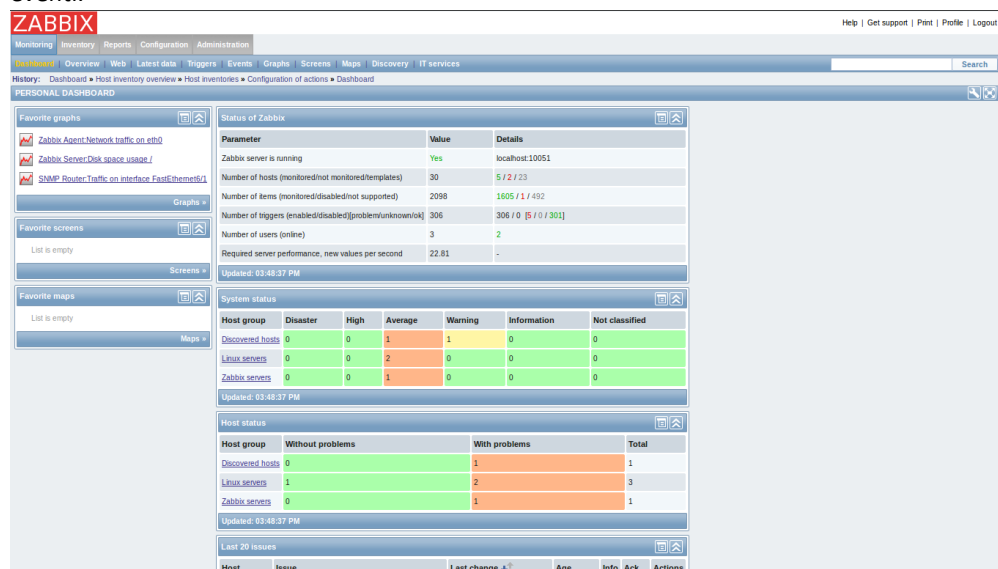


Figura XI Dashboard Zabbix

La prima finestra che ci verrà proposta sarà quella dello Stato attuale di Zabbix. I parametri principali sono riportati nella tabella sottostante. In questo modo sarà possibile stabilire, ad esempio, il corretto funzionamento del Server.

A seguire troveremo System Status, che indica la salute degli Host monitorati. La tabella mostra gli Host Group presenti ed indica eventuali Trigger scattati, con l'ordine di importanza.

A sinistra si avrà accesso rapido a Grafici, schermate e mappe preferite, previa selezione da parte dell'Amministratore.

In altro si avranno i Tab, necessari alla navigazione all'interno del front-end. Il più importante è Configuration, all'interno del quale, come si vedrà nei prossimi capitoli, si potrà accedere alle impostazioni degli Host.

4.7.2 Overview

L'Overview è una particolare sezione dell'interfaccia. In essa si accederà ad una visione generale dello stato degli Host, ma tuttavia più approfondita di quella fornita nella DashBoard.

Infatti si potranno osservare i risultati degli Item ed effettuare, in maniera diretta, l'acknowledgment di eventuali Trigger scattati. Come si osserva nella figura soprastante, tramite Item sarà possibile anche visualizzare lo stato del disco rigido di un Host, la memoria, utilizzata e non, dello stesso ed altre importanti informazioni. In via definitiva, l'overview fornisce una schermata fondamentale del software, tramite la quale sarà possibile interagire con tutti i nodi configurati.

Device name	-	-	uncam_south_uncam...	-	-
Device uptime	-	-	135 days, 21:06:47	-	-
Download speed for scenario "Disponibilità Google".	-	-	-	-	225.78 KBits
Download speed for scenario "Zabbix Frontend".	-	-	-	-	726.62 KBits
Download speed for step "Home" of scenario "Disponibilità Google".	-	-	-	-	336.81 KBits
Download speed for step "About" of scenario "Disponibilità Google".	-	-	-	-	114.76 KBits
Download speed for step "Panna Pagina" of scenario "Zabbix Frontend".	-	-	-	-	235.57 KBits
Download speed for step "Login" of scenario "Zabbix Frontend".	-	-	-	-	1.71 MBits
Download speed for step "Check del Login" of scenario "Zabbix Frontend".	-	-	-	-	624.13 KBits
Download speed for step "Logout" of scenario "Zabbix Frontend".	-	-	-	-	241.34 KBits
Failed step of scenario "Disponibilità Google".	-	-	-	-	0
Failed step of scenario "Zabbix Frontend".	-	-	-	-	0
File read bytes per second	-	-	-	0 Bytes	-
File write bytes per second	-	-	-	2.56 KBits	-
Free disk space on /	-	60.52 B	-	-	6.99 GB
Free disk space on /boot	-	62.15 MB	-	-	-
Free disk space on / (percentage)	-	-	-	-	62.95 %
Free disk space on /boot (percentage)	-	66.38 %	-	-	-
Free disk space on C. (percentage)	-	-	-	17.66 % ✓	-
Free disk space on C.	-	-	-	6.39 GB	-
Free inodes on / (percentage)	-	96.72 %	-	-	75.66 %
Free inodes on /boot (percentage)	-	99.66 %	-	-	-
Free memory	-	-	-	654.94 MB	-
Free swap space	1.27 GB	1023.93 MB	-	1.88 GB	1023 MB
Free swap space in %	99.6 %	99.99 %	-	-	100 %
Host boot time	04/15/2014 12:01:29 PM	04/03/2014 12:39:37 PM	-	-	05/19/2014 03:19:19 PM
Host local time	05/19/2014 03:54:03 PM	05/19/2014 03:54:10 PM	-	-	05/19/2014 03:53:28 PM
Host name	emby	dbs.cs.uncam.it	-	-	micos-zabbix
Host name of zabbix_agentd running	-	Agent2	-	Windows1	Zabbix Server
Hostname	-	-	-	cdm11111	-

Figura XII Overview Zabbix

4.8 CONFIGURAZIONE GENERALE

4.8.1 Configurazione Host con Agent

Tipicamente un Host Zabbix rappresenta un device che si intende monitorare (server, switches, workstation, etc.) Creare un Host è uno dei primi passi per il monitoraggio in Zabbix. Se si vuole monitorare dei parametri in un server "x", prima sarà necessario creare un Host, del tipo, "Server X" nel front-end e si potrà poi settarlo con degli Item per il monitoraggio. Gli host sono organizzati in gruppi.

Per configurare un Host nel front-end, occorre:

- Andare su *Configuration* → *Hosts*
- Premere su *Create* sulla destra
- Immettere i parametri corretti desiderati all'interno della form

Si possono anche usare le funzioni *Clone* e *Full Clone* su degli Host già creati per creare un nuovo host. Usando *Clone* rimarranno tutti i parametri dell'host selezionato e i link ai template. *Full Clone* in aggiunta conserverà tutte le altre entità (Item, trigger, grafici, etc.).

Nota Bene: Quando un host viene clonato, le entità del template modificate non vengono ereditate ma tornano quelle standard iniziali.

Configurazione Host Agent

Il Tab Host contiene gli attributi generali.

IP address	DNS name	Connected to	Port	Default
192.168.3.2		IP DNS	10050	Monitor

Parametri	Descrizione
Host Name	Si inserisca un hostname univoco. Nota: Se si intende monitorare un Host con un Agent installato, il parametro Host Name deve corrispondere al nome inserito nel file di configurazione dell'Agent. E' necessario per l'utilizzo degli active checks.
Visible Name	Se si setta questo parametro, si visualizzerà il nome scelto nelle liste, mappe, etc. Questo attributo supporta UTF-8.
Groups	Seleziona un Host Group nel quale l'Host che verrà creato sarà inserito. Un Host deve appartenere almeno ad un gruppo.
New Host Group	Shortcut per creare un nuovo gruppo e linkarlo all'host. Viene ignorato se vuoto.
Interfaces	Per gli Host sono supportate diversi tipi di interfacce: <i>Agent, SNMP, JMX e IPMI</i> .
IP Address	Indirizzo IP dell'Host (Opzionale).
DNS Name	Nome DNS dell'Host (Opzionale).
Connect To	Selezionando i rispettivi bottoni, si dirà al Server Zabbix come recuperare i dati dagli Agent.
Port	Il numero della porta TCP. Il valore di default dell'Agent Zabbix è 10050.
Default	Permette di selezionare l'interfaccia di default.
Monitored by Proxy	L'Host può essere monitorato dal Server Zabbix o da un Proxy Zabbix: (no proxy) – L'Host verrà monitorato dal Server Zabbix Proxy Name – L'Host verrà monitorato dal Proxy Zabbix identificato dal nome selezionato.
Status	Stato dell'Host: Monitored – Host Attivo, pronto per essere monitorato. Not Monitored – Host non Attivo, per cui non monitorato.

Il Tab **Templates** permette di linkare degli, appunto, Template agli Host che si vuole creare. Tutte le entità verranno collegate al nuovo nodo da monitorare.

Per eliminare il link ad un template, si possono usare due funzioni:

- *Unlink* – Scollega il template, ma conserva gli elementi ormai aggiunti (item, trigger, e grafici)
- *Unlink and clear* – Scollega il template e rimuove tutti gli elementi che erano stati aggiunti

Il Tab **IPMI** contiene gli attributi di gestione dell'IPMI

Parametro	Descrizione
Authentication Algorithm	Selezione dell'algoritmo di autenticazione
Privilege Level	Selezione del livello di privilegio
Username	Username per l'autenticazione
Password	Password per l'autenticazione

Il Tab **Macros** permette di definire delle Macro al livello Host.

Il Tab **Host Inventory** permette di riempire manualmente le informazioni dell'host. Sarà possibile selezionare *Automatic* per la popolazione automatica dei dati, o disattivarla per l'Host corrente.

Configurazione Host Group

Per configurare un gruppo dal front-end Zabbix, occorre procedere nel seguente modo:

- Andare su *Configuration* → *Host Groups*.
- Premere su *Create Group* nell'angolo in alto a destra.
- Inserire i parametri del gruppo nella form.

Parametro	Descrizione
Group Name	Inserire un nome univoco per il gruppo.
Hosts	Selezionare gli Host, membri del gruppo. Un gruppo può avere zero, uno o molteplici Host.

4.8.2 Configurazione Host con SNMP

Si andranno ora ad analizzare i vari step per la configurazione di un Host con SNMP.

4.8.2.1 Step 1: Creazione Host

Si procede creando un Host per un device con interfaccia SNMP.

Come già illustrato precedentemente, si inserisca l'indirizzo IP del device e, momentaneamente, si setti lo stato a "Not Monitored". Per facilitare la configurazione degli Item, si può utilizzare un Template, già integrato nel software, chiamato *Template SNMP Device*.

4.8.2.2 Step 2: Trovare l'OID da monitorare

Si vuole, ora, trovare la stringa SNMP (o l'OID) dell'object che si vuole monitorare. Si può usare il comando `SNMPwalk` (parte della suite `net-snmp`) per ottenere una lista di stringhe SNMP.

```
shell> snmpwalk -v 2c -c public <host IP>
```

E' quindi possibile scorrere l'elenco fino a trovare la stringa che si desidera monitorare come, ad esempio, per monitorare i byte che arrivano al proprio switch nella terza porta, si può usare:

```
IF-MIB::ifInOctets.3 = Counter32: 3409739121
```

Ora si potrà utilizzare il comando `snmpget` per trovare il valore numerico OID per 'IF-MIB::ifInOctets.3'

```
shell> snmpget -v 2c -c public -On 10.62.1.22 IF-MIB::ifInOctets.3
```

Il risultato sarà qualcosa di simile a questo:

```
.1.3.6.1.2.1.2.2.1.10.3 = Counter32: 3472126941
```

In questo contesto si è utilizzata la suite Net-SNMP, descritta nei capitoli precedenti. E' tuttavia possibile utilizzare un qualsiasi altro programma in grado di rilevare l'OID desiderato.

I dati appena raccolti saranno necessari durante la configurazione di un Item per il monitoraggio tramite SNMP. Di seguito, si andrà ad analizzare l'implementazione e la corretta configurazione di quest'ultimi.

4.8.3 Configurazione Item

Gli Item sono i responsabili del recupero dati dagli Host.

Una volta configurato un Host, occorre aggiungere degli Item affinché si possa cominciare a tutti gli effetti ad ottenere dati di monitoraggio.

Uno dei metodi per aggiungere molteplici Item velocemente è quello di linkare all'Host un Template predefinito. Per ottimizzare le performance di recupero dati, tuttavia, potrebbe essere necessario ridefinire gli Item e conservare solo quelli necessari e con la frequenza di recupero necessaria.

In un Item individuale si può specificare quale tipo di dato recuperare dall'Host.

A tal proposito si utilizza una Item Key. Ve ne sono di diversi tipi ed utilizzi; ad esempio l'Item Key **system.cpu.load** raccoglierà di dati relativi al carico sul processore, mentre un Key **net.if.in** recupererà le informazioni del traffico in entrata. A tali parole chiave si possono passare di parametri per specificare nel dettaglio i dati da raccogliere (es. **net.if.in[eth0]** mostrerà il traffico in entrata sull'interfaccia eth0).

4.8.3.1 Creazione di un Item

Overview

Per creare un Item dal frontend Zabbix, si proceda come segue:

- Andare su *Configuration*→*Hosts*
- Premere sulla colonna *Item* dell'Host desiderato
- Premere *Create Item* in alto a destra
- Inserire i parametri dell'Item nella form

Configurazione Item

The screenshot shows the configuration form for a Zabbix item named 'Available memory'. The form is titled 'Item: Available memory'. Key fields include:

- Host:** Zabbix server (with a 'Select' button)
- Name:** Available memory
- Type:** Zabbix agent
- Key:** vm.memory.size[available] (with a 'Select' button)
- Host interface:** 192.168.3.41:10050
- Type of information:** Numeric (unsigned)
- Data type:** Decimal
- Units:** B
- Use custom multiplier:** (value: 1)
- Update interval (in sec):** 60
- Flexible intervals:** A table with columns 'Interval', 'Period', and 'Action'. It currently shows 'No flexible intervals defined.'
- New flexible interval:** A row with 'Interval (in sec)' set to 50 and 'Period' set to 1-7,00:00-24:00, with an 'Add' button.
- Keep history (in days):** 7
- Keep trends (in days):** 365
- Store value:** As is
- Show value:** As is (with a 'show value mappings' link)
- New application:** (empty field)
- Applications:** A dropdown menu with options: -None-, CPU, Filesystems, General, Memory, Network interfaces.
- Populates host inventory field:** -None-
- Description:** Available memory is defined as free+cached+buffers memory.
- Status:** Enabled

 At the bottom, there are 'Save' and 'Cancel' buttons.

Attributi degli Item:

Parametri	Descrizione
Host	Seleziona l'Host o il Template.
Name	Il nome che avrà l'Item una volta creato.
Type	Il type dell'Item.
Key	La chiave dell'Item. Rappresenta una sorta di "istruzione" per identificare un dato da recuperare. La chiave deve essere univoca in un singolo Host. Esempio di chiave: vfs.fs.size[/,free] ← recupererà lo spazio libero nel disco /
Host Interface	Selezione l'interfaccia dell'Host. Questo campo sarà disponibile quando si andrà a modificare un Item a livello di singolo Host.
Type of Information	Il tipo di dato raccolto nel database. Può essere di tipo: <ul style="list-style-type: none"> • Numeric (unsigned) – Intero unsigned a 64bit. • Numeric (float) – Numero in virgola mobile. Possono essere raccolti valori negativi. • Character – Carattere(o stringa) limitato a 255 bytes. • Log – File di Log. • Text – Testo.
Data Type	Viene usato per gli Item Integer in modo da specificare precisi Data Type previsti:

	<ul style="list-style-type: none"> • Boolean – Rappresentazione testuale tradotta in 0 od 1. Il primo rappresenta il valore “falso”, mentre il secondo rappresenta “vero”. Qualsiasi valore numerico non zero viene considerato “vero”. • Octal – Dato in forma ottale. • Decimal – Dato in forma Decimale. • Hexadecimal – Dato in forma esadecimale.
Units	<p>Se viene settato un simbolo per le unità, Zabbix li aggiungerà dopo aver processato i dati e li mostrerà insieme all’unità prescelta.</p> <p>Di default , se il valore sorpassa i 1000, viene diviso per 1000 e mostrato di conseguenza. Per esempio, se si setta bps e si riceve un valore come 881764, esso verrà mostrato come 881.76 Kbps.</p> <p>Un processo speciale viene usato per le unità B (Byte), Bps (Bytes per second), i quali vengono divisi per 1024. Per cui, se si setta come unità B o Bps Zabbix mostrerà: 1 come 1B/1Bps 1024 come 1KB/1KBps 1536 come 1.5KB/15KBps</p> <p>Un processo speciale viene usato per le seguenti unità relative al tempo:</p> <p>unixtime – tradotto in “yyyymm.dd hh:mm:ss” ovvero la data e l’ora con precisione a secondi. Per ottenere un dato corretto occorre selezionare come Type of Information <i>Numeric(unsigned)</i></p> <p>uptime – tradotto in “hh:mm:ss” o come “N° giorni, hh:mm:ss”. Per esempio se si riceve come valore 881764(secondi), verrà mostrato come dato “10 giorni, 04:56:04”</p> <p>s – tradotto in “yyy mmm ddd hhh mmm sss ms”; il parametro viene trattato come il numero di secondi. Per esempio, se si riceve il valore 881764 (secondi), verrà mostrato come dato “10d 4h 56m”</p> <p>Se non sono presenti dei giorni, verranno mostrati i secondi e i millisecondi.</p>
Use Custom Multiplier	Se si abilita questa opzione, tutti i dati ricevuti verranno moltiplicati per l’intero o il numero in virgola mobile settato nel campo. Si utilizzi questa opzione per convertire valori ricevuti in KB, MBps, etc. in B, Bps.
Update Interval (in sec)	Ricarica l’Item ogni N secondi.
Flexible Intervals	Si possono creare eccezioni nell’intervallo di aggiornamento. Per esempio: Intervallo: 10 . Periodo: 1-5.09:00-18:00 – ricaricherà i dati ogni 10 secondi durante l’orario stabilito, durante la settimana. Intervallo: 0 . Periodo: 1-7 0:00-07:00 – non caricherà i dati durante la notte.
Keep History (in days)	Il numero di giorni in cui verranno conservati i dati raccolti dall’Item nel database. Quelli più vecchi verranno eliminati. Si raccomanda di mantenere i dati solo per il tempo strettamente necessario a studiare i risultati.
Keep trends (in days)	Mantiene aggregati i valori massimi e medi, così come il numero totale di valori in quel momento per N giorni.
Store Value	<p>As is – Conserva il dato così come è, senza processarlo.</p> <p>Delta (speed per second) – Valuta il risultato come $(value-prev_value)/(time-prev_time)$, dove: value – rappresenta il valore corrente. Value_prev – il valore ricevuto precedentemente Time – l’ora corrente Prev_time – l’ora del valore ricevuto precedentemente.</p> <p>Questo settaggio è estremamente utile per ottenere la velocità al secondo per un valore in crescita costante.</p>

	Delta (simple change) – Valuta il risultato come <i>(value-prev_value)</i> .
Log Time Format	Disponibile solo per Item di tipo Log. Formati supportati: <i>*y</i> : Anno (0001 – 9999) <i>*M</i> : Mese (01-12) <i>*d</i> : Giorno (01-31) <i>*h</i> : Ora (0-23) <i>*m</i> : Minuto (00-59) <i>*s</i> : Secondo (00-59) Se lasciato in bianco, l’orario non verrà processato. Per esempio, considerando la seguente linea di log da un Agent Zabbix: <i>“23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).”</i> Comincia con sei caratteri per il PID(numero identificante il processo Agent Zabbix), seguiti da data, ora e il resto della linea di log. Il formato dell’ora per questo risultato sarebbe: <i>“pppppp:yyyyMMdd:hhmmss”</i> .
New Application	Si inserisca il nome per una nuova applicazione per l’Item.
Applications	Linka l’Item ad una o più applicazioni esistenti.
Populates Host Inventory Field	Si può selezionare un campo dell’inventario che verrà riempito con il risultato ottenuto dall’Item. Funzionerà solo se già impostato il valore Automatic nell’apposito Tab dell’Host.
Description	Si inserisca una descrizione per l’Item
Status	Enabled – L’item verrà processato Disabled – L’item non verrà processato Not Supported – L’item non è supportato. L’item non verrà processato, tuttavia Zabbix potrebbe periodicamente settare lo stato su Enabled se riconosciuto valido dopo un determinato periodo di tempo.

4.8.3.2 Item Types

Sono i vari controlli offerti da Zabbix. I principali sono: Zabbix Agent, Simple Checks, SNMP, etc.

Alcuni controlli vengono effettuati dal Server Zabbix autonomamente (in quanto non necessitano di Agent) mentre altri richiedono uno Zabbix Agent attivo sulla macchina sulla quale si intende effettuare dei check. Si andranno ora ad analizzare uno dei Tipi principali del software.

Zabbix Agent

Questi controlli si basano sulla comunicazione con l'Agent per il recupero dei dati. Segue una tabella rappresentate alcuni degli Item Key principali, utilizzabili con questo Item Type:

Key				
Name	Description	Return Value	Parameters	Comments
agent.hostname	Ritorna il nome dell'Host su cui è stato installato l'Agent.	String	-	Ritorna il nome dell'Host, prendendolo dai file di configurazione dell'Agent.
Agent.ping	Controlla la disponibilità dell'Agent.	Ritorna '1' se l'Agent è disponibile, niente altrimenti.	-	
net.if.in[if, <mode>]	Statistiche sul traffico dati in entrata sull'interfaccia di rete scelta	Integer	If – Nome dell'Interfaccia di Rete Mode – Possibili valori: <i>bytes</i> – numero dei bytes(default) <i>packets</i> – numero dei pacchetti <i>errors</i> – numero di errori <i>dropped</i> – numero di pacchetti persi	Esempio di chiavi: <i>net.if.in[eth0,errors]</i> <i>net.if.in[eth0]</i>
system.cpu.load[<cpu>,<mode>]	CPU load	Processor load. Float.	Cpu – possibili valori: <i>all</i> (default), <i>percpu</i> (il totale del carico diviso con il count CPU) mode – possibili valori: <i>avg1</i> (un minuto di media, default), <i>avg5</i> (5 minuti di media), <i>avg15</i> (una media di 15 minuti)	Esempio di chiave: system.cpu.load[,avg 5]

4.8.4 Configurazione dei Trigger

Overview

I Trigger sono espressioni logiche che “valutano” i dati recuperati dagli Item e rappresentano lo stato corrente del sistema.

Mentre gli Item vengono usati per recuperare i dati dei sistemi, è molto poco pratico seguire i dati individualmente attendendo che si verificano problemi degni di attenzione. Il lavoro di “valutazione” viene lasciato ai Trigger.

Le espressioni dei Trigger permettono di definire una soglia entro il quale il dato ricevuto è “accettabile”. Dunque, quando uno dei dati ricevuti non è “accettabile”, viene lanciato un trigger o viene cambiato lo stato dello stesso in PROBLEM.

Un Trigger può avere i seguenti stati:

Valore	Descrizione
OK	Lo stato normale del Trigger.
PROBLEM	Di norma indica che è accaduto qualcosa. Per esempio, il carico sul processore troppo elevato.

Lo stato del Trigger (l'espressione) viene ricalcolato ogni volta che il Server Zabbix riceve un nuovo valore facente parte di una espressione.

Se si è fatto uso di funzioni basate sul tempo all'interno del Trigger, questo verrà ricalcolato ogni 30 secondi da un Timer integrato nel software.

4.8.4.1 Trigger Expression

Le espressioni usabili sono molto flessibili. Si possono creare dei test logici complessi riguardanti le statistiche monitorate.

Una espressione utile, quanto semplice, potrebbe essere la seguente:

```
{<server>:<key>.<function>( <parameter> )}<operator><constant>
```

Funzioni

Le funzioni dei Trigger consentono di fare riferimento ai valori acquisiti, il tempo corrente ed altri fattori.

Parametri delle Funzioni

La maggior parte delle funzioni numeriche accettano il numero di secondi come parametro. Si può utilizzare il simbolo “#” per specificare che un parametro ha un diverso significato:

Function Call	Significato
sum(600)	Somma dei valori racchiusi nello spazio di tempo di 600 secondi
sum(#5)	Somma degli ultimi 5 valori

La funzione **last** ha un diverso significato per i valori quando si usa come prefisso il simbolo “#” Permette di scegliere l’n-esimo valore; per esempio, dati i valori 3, 7, 2, 6, 5 (dal più recente al meno recente), **last(#2)** ritornerà 7 e **last(#5)** ritornerà 5.

Operatori

Le seguenti operazioni sono supportate nei Trigger (Con priorità discendente di esecuzione):

Priorità	Operatore	Definizione
1	/	Divisione
2	*	Moltiplicazione
3	-	Meno Aritmetico
4	+	Più Aritmetico
5	<	Meno di
6	>	Più di
7	#	Non Uguale
8	=	Uguale
9	&	AND Logico
10		OR Logico

Esempi di Trigger

Esempio 1:

Si supponga che il sito ‘www.zabbix.com’ sia sovraccarico:

```
{www.zabbix.com:system.cpu.load[all,avg1].last(0)}>5|{www.zabbix.com:system.cpu.load[all,avg1].min(10m)}>2
```

L’espressione è vera quando o il carico corrente sulla CPU supera il valore 5 o quando il carico sul processore risulta maggiore di 2 durante gli ultimi 10 minuti. ‘www.zabbix.com:system.cpu.load[all,avg1]’ fornisce un breve nome del parametro monitorato. Specifica che il server è ‘www.zabbix.com’ e che la chiave da monitorare è ‘system.cpu.load[all,avg1]’. Usando ‘last()’ ci si riferisce al valore più recente. Infine, ‘>5’ significa che il Trigger risulterà nello stato PROBLEM quando il valore più recente rappresentante il carico sul processore sarà maggiore di 5.

Esempio 2:

Si supponga che qualcuno stia scaricando un grosso file da Internet:

```
{www.zabbix.com:net.if.in[eth0,bytes].min(5m)}>10000K
```

L’espressione risulterà vera quando si riceve un numero di byte, dall’interfaccia eth0, maggiore di 10000KB negli ultimi 5 minuti.

4.8.5 Configurazione Ricezione Notifiche a Problemi

Overview

Si supponga che siano stati configurati Item e Trigger e che stiano accadendo degli eventi come risultato del cambiamento di stato di alcuni dei Trigger suddetti.

Per cominciare, invece di aspettare che i Trigger cambino di stato per agire, è necessario impostare la ricezione di notifiche nel caso in cui capiti qualcosa di significativo (come un PROBLEM). Inoltre, in questo caso, si procederà ad informare i soggetti interessati a tale informazione.

Per poter inviare e ricevere notifiche da Zabbix occorre aver:

- Definito dei Media
- Aver configurato un'Action che invii un messaggio ad un Media predefinito.

Le Action si compongono di condizioni e operazioni. In sostanza, quando vengono riscontrate delle condizioni, si effettuano delle operazioni. Le due operazioni principali che si usano più spesso sono inviare un messaggio (notifica) e eseguire comandi da remoto.

Media Type

I Media sono i canali di distribuzione utilizzati per l'invio di notifiche e avvisi in Zabbix. Sono supportati diversi tipi:

- E-mail
- SMS
- Jabber

Prendiamo come esempio, la configurazione del Media Type e-mail:

Per configurare le e-mail come canale di comunicazione, occorre configurare le impostazioni necessarie e assegnare lo specifico indirizzo dell'utente.

Configurazione Media Type

- Andare su *Administration* → *Media Type*
- Premere su *Create Media Type* (oppure su *e-mail* nella lista dei media predefiniti)



The screenshot shows the 'Media' configuration form in Zabbix. The form has a blue header with the word 'Media'. Below the header, there are several input fields and a checkbox. The 'Description' field contains the text 'Email'. The 'Type' field is a dropdown menu with 'Email' selected. The 'SMTP server' field contains 'mail.company.com'. The 'SMTP helo' field contains 'company.com'. The 'SMTP email' field contains 'zabbix@company.com'. At the bottom, there is a checkbox labeled 'Enabled' which is checked.

Di seguito si illustrano gli attributi dei Media Type:

Parametro	Descrizione
Description	Nome del Media Type.
Type	Selezionare <i>Email</i> come tipo.
SMTP Server	Serve a settare un server SMTP per gestire i messaggi in uscita.
SMTP Helo	Serve a settare un valore helo SMTP, normalmente il nome del dominio.
SMTP Email	L'indirizzo inserito sarà usato come " From " per il messaggio inviato

Configurazione User Media

Per assegnare uno specifico indirizzo ad un User:

- Andare su *Administration* → *Users*
- Aprire le proprietà dell' user
- Nel Tab Media, premere su *Add*

Attributi degli User Media:

Parametro	Descrizione
Type	Si selezioni <i>Email</i> come tipo.
Send to	Si specifichi l'indirizzo e-mail a cui inviare i messaggi.
When Active	Si può settare il limite di tempo nel quale inviare le notifiche. Per esempio, per i giorni lavorativi: (1-5, 09:00-18:00).
Use if severity	Si spuntino i tipi di allarmi che si vogliono ricevere, in ordine di importanza.
Status	Lo stato dell'User Media: Enabled – E' in uso. Disabled – Non è in uso.

Actions

Occorre configurare correttamente delle Action se si vuole, ad esempio, che vengano effettuate delle operazioni, una volta riscontrati dei problemi.

Le Azioni possono essere definiti per tutti i tipi di eventi che occorrono:

- Trigger – quando un Trigger cambia stato.
- Discovery – quando si scopre un device
- Auto Registration – quando un agente si auto registra

Configurazione Action

Per configurare una azione:

- Andare su *Configuration* → *Actions*
- Dal menu a tendina *Event Source* selezionare la risorsa desiderata
- Premere su *Create Action*
- Stabilire gli attributi generali
- Scegliere l'operazione da intraprendere nell'apposito tab *Operations*
- Scegliere le condizioni nelle quali effettuare l'azione, all'interno dell'apposito tab *Conditions*

Attributi Generali:

Action	Conditions	Operations
Name	Report problems to Zabbix administrators	
Default operation step duration	300 (minimum 60 seconds)	
Default subject	{TRIGGER.STATUS}; {TRIGGER.NAME}	
Default message	Trigger: {TRIGGER.NAME} Trigger status: {TRIGGER.STATUS} Trigger severity: {TRIGGER.SEVERITY} Trigger expression: {TRIGGER.EXPRESSION} 1. Item value on {HOST.NAME1}: {ITEM.VALUE1} ({ITEM.NAME1}) 2. Item value on {HOST.NAME2}: {ITEM.VALUE2} ({ITEM.NAME2})	
Recovery message	<input checked="" type="checkbox"/>	
Recovery subject	{TRIGGER.STATUS}; {TRIGGER.NAME}	
Recovery message	Trigger: {TRIGGER.NAME} Trigger status: {TRIGGER.STATUS} {EVENT.ID} {EVENT.AGE} {EVENT.DATE} {EVENT.TIME} {EVENT.ACK.STATUS} {EVENT.ACK.HISTORY} {EVENT.RECOVERY.ID}	
Enabled	<input checked="" type="checkbox"/>	

Parametri	Descrizione
Name	Nome univoco
Default Operation step duration	Durata di uno step operativo di default (minimo 60 secondi). Per esempio, uno step di un'ora significa che se un'operazione viene intrapresa, passerà un'ora prima di procedere allo step successivo.
Default Subject	Il messaggio subject di default
Default Message	Il messaggio di default
Recovery Message	Un messaggio di recovery speciale può essere inviato subito dopo la risoluzione di un problema.
Recovery Subject	Il messaggio subject di recovery
Enabled	Spuntare la checkbox per abilitare l'azione. Verrà ritenuta disattivata altrimenti.

4.8.6 Configurazione Template

Overview

Un Template è un set di entità che possono venire applicate a molteplici host, per semplificare il processo di configurazione di quest'ultimi.

Le entità possono essere:

- Item
- Trigger
- Grafici
- Application
- Etc.

Poiché molti Host sono identici o, quanto meno, molto simili fra loro, appare naturale che una serie di entità si possa adattare per uno o più Host. Con i Template si risparmia tempo e complessità di configurazione del Software.

Quando ad un Host viene "linkato" un Template, gli vengono aggiunte tutte le entità di quest'ultimo. I Template vengono assegnati direttamente individualmente ad ogni Host.

Vengono spesso usati per raggruppare Host con determinati servizi o applicazioni (come Apache, MySQL, etc...)

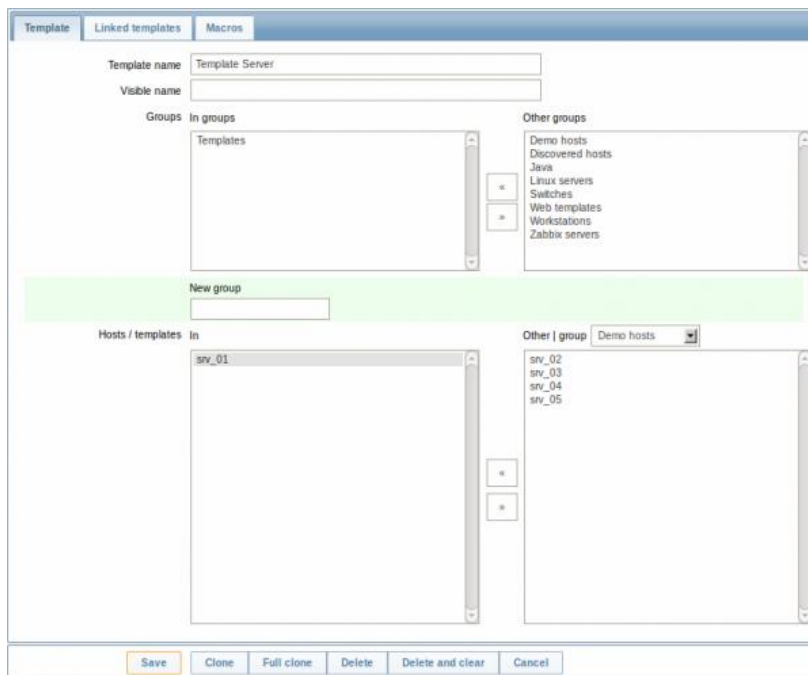
Un altro beneficio nell'usare i Template è che quando occorre modificare una delle entità per tutti gli Host, invece di procedere singolarmente, si può procedere con una alterazione del Template.

Configurazione

Configurare un Template richiede prima la creazione dello stesso, definendo i parametri generali, proseguendo con l'aggiunta delle entità

Per creare un Template:

- Andare su *Configuration* → *Templates*
- Premere su *Create Template*
- Modificare gli attributi del Template



Attributi Template:

Parametro	Descrizione
Template Name	Nome univoco del Template.
Visible Name	Se settato, sarà il nome visibile nelle altre configurazioni.
Groups	Host e/o Template ai quali apparterrà.
New Group	Si può creare un nuovo gruppo che conterrà il Template in creazione. Ignorato se vuoto.
Host/Templates	Lista degli Host e/o Template a cui verrà applicato.

Il Tab **Linked Templates** permette di linkare uno o più Template annidati a quello in creazione. Tutte le entità verranno aggiunte a quest'ultimo.

Aggiungere Item, Trigger, Grafici

Per aggiungere elementi al Template procedere come segue:

- Andare su *Configuration* → *Hosts* (o *Template*)
- Premere su *Items* nella riga dell'Host desiderato
- Spuntare i valori degli Item che si intende aggiungere al Template
- Selezionare *Copy selected to* sotto la lista e premere *Go*
- Selezionare il Template (o il gruppo) nel quale gli Item andranno copiati

Per aggiungere Trigger e Grafici si segua la stessa linea.

4.9 ANALISI DEI RISULTATI

4.9.1 Implementazione e studio dei Grafici

Essendo presenti parecchi dati che fluiscono in Zabbix, risulterebbe molto più facile, per gli utenti, osservare una rappresentazione visiva di ciò che sta accadendo, piuttosto che meri numeri.

Qui è dove entrano in gioco i Grafici. Essi permettono di comprendere il flusso di dati a colpo d'occhio, in modo da comprendere eventuali problemi, scoprire in quale momento se ne è creato uno o, anche, prevedere la possibile formazione di complicazioni.

Zabbix offre agli utenti semplici grafici incorporati, nonché la possibilità di crearne di personalizzati, più complessi.

Semplici grafici sono disponibili per la visualizzazione dei dati raccolti dagli Item. Non è necessaria alcuna configurazione da parte dell'utente per visualizzarli. Sono messi a disposizione dal software stesso.

4.9.1.1 Grafici Semplici

Selettore Tempo:



Figura XIII Grafico Zabbix: Carico sul Processore

In Figura si può osservare un esempio di Grafico, rappresentate il carico sul Processore all'interno del Server Zabbix. Il selettore sopra il grafico permette di selezionare, appunto, il periodo di tempo desiderato. Il cursore al suo interno può essere trascinato avanti e indietro, modificando il periodo di tempo visualizzato. Si potrà anche evidenziare una zona del grafico con il tasto sinistro del mouse. Il grafico effettuerà uno zoom nella parte scelta dall'utente.

Dati Recenti e Periodi Lunghi:

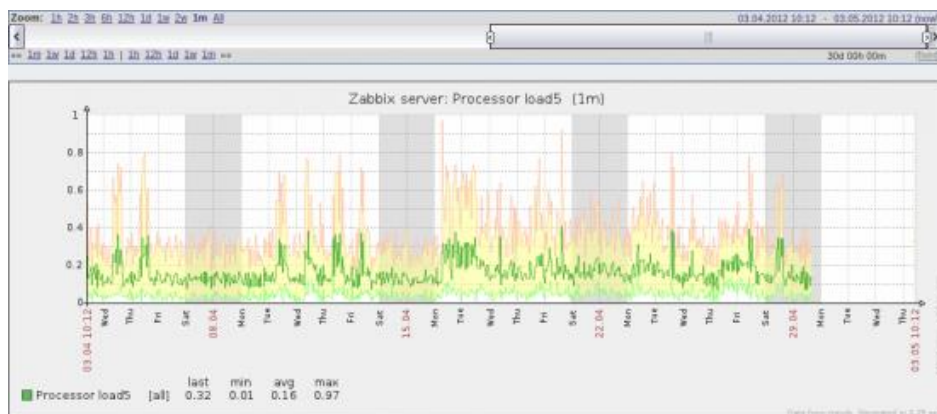


Figura XIV Grafico:Carico sul Processore 2

Per i dati molto recenti il grafico mostrerà una singola linea. Per i dati che mostrano un periodo di tempo più lungo, vengono disegnate diverse linee: una verde scuro che mostra la media dei valori ricevuti; una rosa chiaro e una verde chiaro che invece mostrano i valori massimi e minimi nello spazio temporale evidenziato (vedi Figura).

4.9.1.2 Grafici Custom

Mentre i grafici semplici sono buoni per visualizzare i dati di un singolo Item, non offrono possibilità di configurazione. Perciò, se si vuole cambiare lo stile del grafico o comparare Item di diverso tipo, come ad esempio il traffico in entrata e quello in uscita, occorre creare un Grafico Custom, ovvero personalizzato.

Si possono creare per un singolo Host o per un Template.

Configurazione

Per creare un Grafico Custom, si proceda come segue:

- Andare su *Configuration* → *Hosts*.
- Premere su *Graphs* nell'Host desiderato.
- Nella schermata che viene proposta, premere *Create Graph*.
- Impostare il grafico.

Name	Function	Draw style	Y axis side	Colour	Action
1: Zabbix server: Outgoing network traffic on eth0	avg	Filled region	Right	00C800	Remove
2: Zabbix server: Incoming network traffic on eth0	avg	Bold line	Right	C80000	Remove

Figura XV Configurazione Grafico Personalizzato

Attributi del Grafico:

Parametro	Descrizione
Name	Nome univoco del Grafico.
Width	La larghezza del Grafico in pixel.
Height	L'altezza del Grafico in pixel.
Graph Type	Tipo di Grafico: <ul style="list-style-type: none"> • Normal – Grafico normale. I valori vengono mostrati con delle linee. • Stacked – Grafico a pila. • Pie – Grafico a torta (utile per mostrare lo spazio sul disco rimasto e/o usato). • Exploded – Grafico a torta, dove le componenti risultano separate per una migliore comprensione delle parti.
Show Legend	Questa opzione mostrerà, se selezionata, la legenda del grafico.
Show Working Time	Questa opzione, se selezionata, mostrerà le ore in cui non sono stati raccolti dati in grigio. Non disponibile nei Grafici a Torta.
Show Triggers	Mostrerà i Trigger dell'Item, se configurati, all'interno del grafico, tramite una linea rossa. Non disponibile nei Grafici a Torta.
Y axis MIN value	Il minimo valore dell'asse delle ordinate: Calculated – Il valore minimo verrà calcolato automaticamente. Fixed – Il valore minimo viene fissato dall'utente. Item – L'ultimo valore ricevuto dall'Item verrà settato come valore minimo.
Y axis MAX value	Il massimo valore dell'asse delle ordinate: Calculated – Il valore massimo verrà calcolato automaticamente. Fixed – Il valore massimo viene fissato dall'utente. Item – L'ultimo valore ricevuto dall'Item verrà settato come valore massimo.
3D view	Abilita la vista 3d. Disponibile solo per i Grafici a Torta.
Items	Mostra gli Item del Grafico.

4.9.1.3 Esempi di Grafici

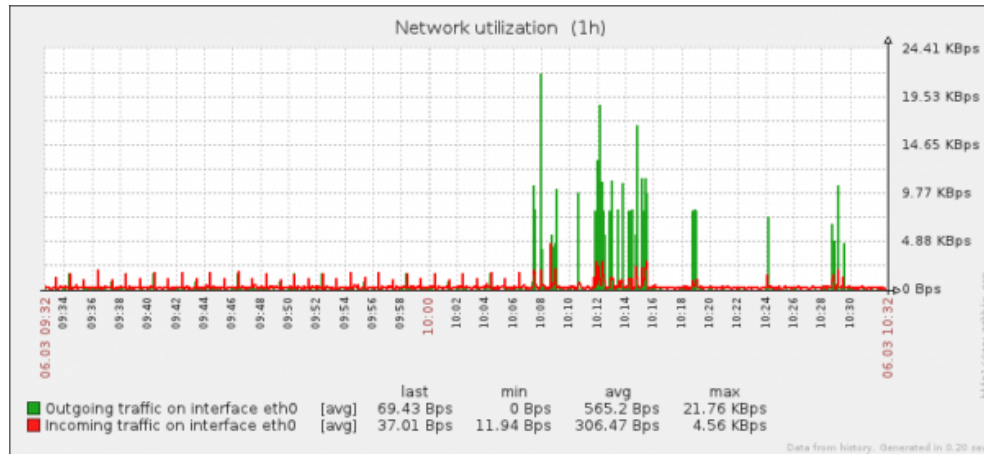


Figura XVII Grafico: Utilizzo della Rete

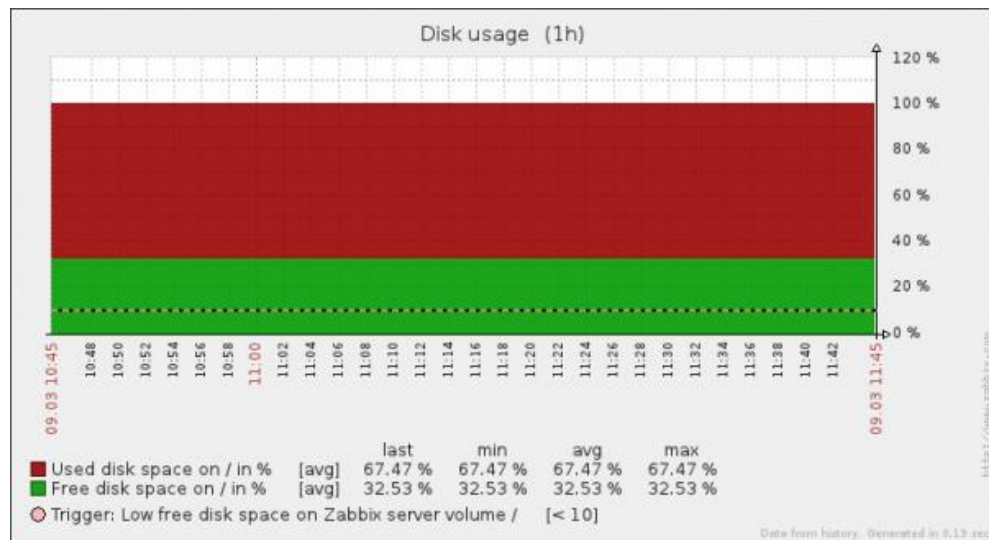


Figura XVI Grafico: Spazio sul Disco

Si noti come sull'ultimo grafico (Figura XV) si possa osservare il Trigger, ovvero la linea tratteggiata. Se lo spazio libero dovesse finire al di sotto di esso, quest'ultimo scatterebbe.

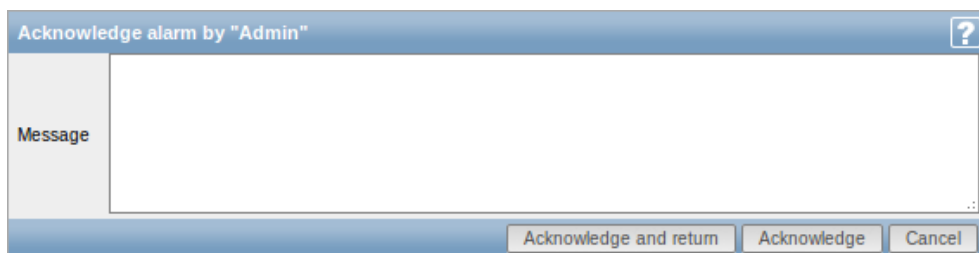
4.10 ACKNOWLEDGMENT DEGLI EVENTI

Gli eventi possono essere “riconosciuti” dagli Amministratori. Tale pratica viene definita in gergo “acknowledgment”. Consiste nel notificare l’avvenuta presa di coscienza dell’esistenza di un problema o evento.

Se un Utente Zabbix viene notificato di un problema, può recarsi nel frontend, navigare fra gli eventi nella schermata acknowledgment e “riconoscere” il problema (fare l’acknowledge). Si potrà, all’occorrenza, lasciare anche un commento, in modo da lasciare eventuali chiarimenti ad altri utenti e coordinare i lavori. Funzione utile se si sta utilizzando un Proxy Zabbix.

Lo stato dell’acknowledgment dei problemi viene mostrato in *Monitoring* → *Events*.

La colonna *Ack* contiene due possibili valori, ‘Yes’ oppure ‘No’, che indicano rispettivamente se un problema è stato riconosciuto o meno. Durante l’acknowledgment sarà possibile, come descritto precedentemente, lasciare un commento, tramite la seguente schermata:



Per riconoscere un problema, si inserisca il proprio commento e premere su “*Acknowledgment and return*” o “*Acknowledgment*”

4.11 DISCOVERY

4.11.1 Network Discovery o Automatic Discovery

Zabbix offre funzionalità di identificazione automatica della rete. Con il rilevamento di rete (Discovery Network) configurato correttamente è possibile:

- Accelerare l’implementazione di Zabbix.
- Semplificare l’amministrazione della rete.
- Utilizzare Zabbix in ambienti in rapida evoluzione senza difficoltà amministrative.

Il Network Discovery di Zabbix si basa su le seguenti informazioni:

- Intervalli IP.
- Disponibilità di servizi (FTP, SSH, WEB, POP3, IMAP, TCP, etc.).
- Informazioni ricevute da Agenti Zabbix.
- Informazioni ricevute da Agenti SNMP.

Il rilevamento di rete si costituisce di due parti: Scoperta e azioni.

Discovery

Zabbix scansiona periodicamente il range di indirizzi IP definito dall'amministratore. La frequenza di controllo può essere configurata a piacimento.

Ogni controllo di un servizio effettuato dal modulo Network Discovery genera un evento.

Evento	Risultato
Service Discovered	Appare quando viene scoperto per la prima volta un Host(IP).
Service Up	Il servizio è 'up'
Service Lost	Il servizio è in stato 'down' dopo essere stato in 'up'
Service Down	Il servizio è 'down'
Host Discovered	Almeno un servizio di un Host è 'up' dopo che tutti i servizi di quello stesso Host erano 'down'
Host Up	Almeno un servizio in un Host è 'up'
Host Lost	Tutti i servizi di un Host sono 'down' dopo essere stati in 'up'
Host Down	Tutti i servizi di un Host sono 'down'

Actions

Eventi di scoperta possono essere la base per Action rilevanti, come:

- Invio di notifica.
- Aggiunta/Rimozione di un Host.
- Attivazione/Disattivazione di un Host.
- Aggiunta di un Host ad un gruppo.
- Rimozioni di un Host da un gruppo.
- Link/Unlink di un Host ad un Template.
- Esecuzione remota di script.

Host Creation

Se si sceglie l'operazione Host Creation, a seguito di un rilevamento di rete, verrà aggiunto un Host. Quando ciò accade, quest'ultimo otterrà delle interfacce create in base alle seguenti regole:

- Se riesce un controllo SNMP, verrà creata un'interfaccia SNMP.
- Se un Host ha risposto sia per l'Agent Zabbix che a richieste SNMP, verranno create entrambe le interfacce.
- Se un Host ha risposto solo a controlli da Agent Zabbix, verrà creata solo l'interfaccia apposita.
- Se un Host risponderà in futuro a controlli diversi da quelli registrati alla creazione, verranno introdotte delle nuove interfacce.

4.11.2 Active Agent Auto-Registration

E' possibile consentire l'attivazione automatica degli Agent Zabbix. Con questa impostazione, nuovi Host verranno aggiunti automaticamente alla routine di monitoraggio senza bisogno di configurazione manuale.

L'auto registrazione può avvenire quando un Agent attivo, precedentemente sconosciuto, richiede dei controlli.

La funzione può essere molto utile per il monitoraggio automatico dei nuovi nodi. Non appena si trova un nuovo nodi nella rete, si avvierà automaticamente la raccolta di dati sulle prestazioni e disponibilità dell'Host.

Il Server Zabbix, quando procede con l'aggiunta dell'host scoperto, utilizza l'indirizzo IP e la porta ricevuti per configurarlo. Se nessun valore IP viene ricevuto, viene utilizzato quello per la connessioni in ingresso. Se nessun valore di porta viene ricevuto, viene utilizzata la porta standard 10050.

4.12 API E PLUGINS

Le API di Zabbix forniscono un'interfaccia programmabile per estendere le funzionalità del software con lo scopo di manipolazioni di massa, Integrazione di programmi di terze parti e per altri scopi.

Attualmente tutti gli oggetti contrassegnati come 'draft' sono sperimentali e devono essere usati con grande attenzione. Non è garantita la compatibilità con le versioni future.

Gli oggetti non contrassegnati sono invece stabili ed è possibile usarli anche in ambito aziendale, senza particolari problemi.

Questa sezione fornisce una panoramica delle funzioni previste dalle API:

Monitoraggio:

Consente di accedere alla cronologia dei dati raccolti durante il monitoraggio.

Cronologia:

Permette di recuperare i valori storici raccolti da processi di monitoraggio per la presentazione o l'ulteriore elaborazione.

Eventi:

Permette di recuperare gli eventi generati dai Trigger, l'individuazione della rete e altri sistemi per una gestione della situazione flessibile o l'integrazione di strumenti di terze parti.

Monitoraggio del Servizio:

Recupera informazioni dettagliate sulla disponibilità di qualsiasi servizio IT.

Configurazione:

Queste API consentono di gestire la configurazione del sistema di monitoraggio.

Host e gruppi di Host:

Permette di gestire Host, singoli e gruppi, e tutto ciò che li riguarda. Tra cui interfacce Host, periodi di manutenzione, etc.

Articoli e Applicazioni:

Permette di definire gli elementi da monitorare. Creare o rimuovere applicazioni e assegnare gli elementi dove si desidera.

Trigger:

Permette di configurare Trigger per la notifica di problemi nel sistema.

Grafici:

Permette di modificare i Grafici per una migliore presentazione dei dati raccolti.

Template:

Permette di gestire i Template e il loro collegamento a Host o ad altri Template.

Azioni e avvisi:

Permette di definire azioni e le operazioni di informazione degli utenti a determinati eventi o di eseguire automaticamente comandi da remoto.

Discovery:

Permette di gestire le regole di individuazione a livello di rete per trovare e monitorare i nuovi admin automaticamente. Permette inoltre l'accesso completo alle informazioni sui servizi e Host scoperti.

5 CONCLUSIONI

Con questa tesi è stato illustrato il protocollo SNMP. Durante il percorso di tesi ha svelato le sue potenzialità ed i motivi per cui è il protocollo più diffuso per la gestione della rete. Il fatto che si utilizzi ancora la prima versione (sebbene abbia alcune lacune) dopo sedici anni dalla sua nascita è sinonimo di buona progettazione. Quando si parla di estensibilità di SNMP si fa riferimento più che altro ai programmi Manager, che possono essere facilmente migliorati con nuove funzioni, mentre gli agenti forniti sui dispositivi, generalmente, non permettono modifiche al software interno.

Il software analizzato, Zabbix, funziona su tutti i principali Sistemi Operativi e si è rivelato uno strumento molto potente di monitoraggio di rete; infatti oltre ad avere molte funzioni mirate per il protocollo SNMP, dispone di ottime funzionalità di monitoraggio. La rielaborazione di dati è molto accurata ed è facile ottenere informazioni e grafici in merito al lavoro svolto dalla rete. Inoltre è anche possibile effettuare delle modifiche di configurazione ai dispositivi da una postazione remota, in modo da poter assecondare i cambiamenti continui della morfologia della rete.

6 BIBLIOGRAFIA

- 1) James F. Kurose, Keith W. Ross, *Internet e Reti di Calcolatori 2nd Edition*, McGraw-Hill, 2003
- 2) Douglas Mauro, Kevin Schmidt, *Essential SNMP 2nd Edition*, O'Reilly, 2005
- 3) Robert L. Townsend, *SNMP Application Developer's Guide*, VNR Communications Library, 1995

7 SITOGRAFIA

- 1) www.snmp.com
- 2) www.rfc-base.org
- 3) www.oid-info.com
- 4) www.dart.com
- 5) www.zabbix.com/documentation
- 6) www.zabbix.com/forum
- 7) www.net-snmp.org
- 8) <http://www.manageengine.com/products/mibbrowser-free-tool/documents.html>
- 9) <http://www.opennms.org/>
- 10) <http://ganglia.sourceforge.net/>
- 11) <http://www.nagios.org/>
- 12) RFC 1155
- 13) RFC 1156
- 14) RFC 1157
- 15) RFC 1450
- 16) RFC 1451
- 17) RFC 1452
- 18) RFC 2571
- 19) RFC 2572
- 20) RFC 2573
- 21) RFC 2574

8 RINGRAZIAMENTI

Eccoci qua. Finalmente siamo arrivati al fatidico evento. La mia Laurea. Colgo quest'occasione per fare un po' di ringraziamenti. Voglio anzitutto ringraziare il mio Relatore, Il Professor Marcantoni, che mi ha aiutato nella stesura di questa tesi e durante lo Stage.

Voglio poi ringraziare con tutto il cuore la mia Famiglia: mia Madre e mio Padre, che mi hanno sostenuto durante questi miei anni Universitari. Ci siete sempre stati, con i vostri consigli e con i vostri aiuti. Non avrei potuto chiedere dei genitori migliori. E' grazie a voi che sono divenuto l'uomo che sono ora (oserei dire, un gran bel pezzo d'uomo). Un grazie speciale lo voglio fare anche ad Andrea. Sei il miglior fratello del mondo e sei anche il mio migliore amico. Grazie per tutte le volte che mi hai dato una mano e per tutte le risate che mi hai fatto fare. Auguro a te e a Giulia tanta felicità. La meritate. Un ringraziamento voglio farlo anche ad una persona che oggi, purtroppo, non è più qui, ma non per questo meno importante. Grazie Nonna.

E adesso veniamo ai miei amici. Ringraziarvi singolarmente sarebbe un'impresa titanica. Voglio solo dire che sono fortunato ad avervi conosciuto tutti. Perché un uomo senza amicizie si limita ad esistere. Un uomo e i suoi amici, vivono.

Voglio ringraziare prima di tutto Michele, che oggi si Laurea con me. Sei stato la prima persona che ho incontrato all'Università e devo ammettere che senza di te sarebbe stato tutto molto più difficile. Un grazie anche a Fabio, anche lui insieme a Michele, fra i primi che ho conosciuto. Un grazie anche ad Andrea, il mitico Sampà. Spero di essere stato un coinquilino decente ragazzi. Inutile dire che sono contento di avervi come amici e spero di vivere un altro bel po' di esperienze insieme. Siete dei grandi.

Non posso non ringraziare Erica. Abbiamo passato una marea di avventure insieme, dalle passeggiate nei vicoli bui ai "simpaticissimi" pesci d'aprile che quei gran furboni dei nostri amici ci hanno fatto (ammettiamolo, siamo vittime facili). Sono contento di ogni singolo momento passato insieme. Sei una persona solare, allegra e ansiosa (beh, l'ultima cosa non è positiva per la tua salute, ma vabè...). Sei una persona speciale, non solo per me, ma di fatto. Sii felice e non cambiare. Ringrazio anche tutta la famiglia Comini, che mi ha sempre ospitato e alla quale mi sono ormai affezionato.

Come non ringraziare tutti gli altri amici di Camerino? Massi, Mazza, Pesc, Orad, Bellagamba, Bedde, Mosci, Ola, Luca voi siete i più stretti. Il gruppo di "Quelli che massera". I momenti che ho passato con voi hanno reso la mia esperienza Universitaria unica. Grazie di tutto ragazzi. Un grazie anche a tutte le persone che ho incontrato durante questi anni. Mircoli, Pierpaolo, Gianluca, Elisa, Maria Paola, Serena, Antonella, Alessio, Andrea, sul serio elencarvi tutti mi rimane quasi impossibile. Senza di voi non sarebbe stato lo stesso.

Passiamo ora alla zona di Ascoli e dintorni. Voglio fare un Grazie speciale ad Ester. L'amicizia che ho con te è irrinunciabile. Mi hai sostenuto nei momenti difficili e lo stesso ho cercato di fare io per te. Sei una ragazza speciale, anche se un po' mattacchiona, sempre in grado di farmi ridere. Ringrazio anche Valeria, Elisa, Laura, Sonia, Luana, Alessia e tutte le altre.

Veniamo agli amici di Macerata. Partiamo dai miei amici dal Liceo. Grazie a Ferri, Simò e Marta, che sopportano me e le mie idiozie. Un grazie speciale voglio farlo a Francesco, o meglio, Gabrio. Grazie per tutte le serate che abbiamo passato. Un grazie poi agli amici

dell'atletica: Fede, Lucio, Monia, Valentina e tutti gli altri. Grazie anche ai pistacoppesi Matteo e Cippo.

Vorrei ringraziarmi uno per uno, ma cavolo, questa tesi è già abbastanza lunga di suo! Se non vi ho messo nei ringraziamenti non è perché vi odio (o forse sì?) ma perché non ne ho la possibilità. Avrei voluto scrivere cose più cretine qui dentro ma si sa che la gente non gradisce quando le frasi non terminano come banana. Infine un grazie a te. Sì proprio TE, che stai leggendo questi ringraziamenti. You're awesome.

Non stropicciate la tesi per l'emozione, grazie.