UNIVERSITÀ DEGLI STUDI DI CAMERINO

FACOLTÀ DI SCIENZE E TECNOLOGIE

Corso di Laurea in Informatica

Dipartimento di Matematica e Informatica

SISTEMA DI MONITORAGGIO SNMP IN AMBIENTE WINDOWS

Tesi di Laurea compilativa
In Reti di Elaboratori

Laureando

Marcello Turchetta

Relatore

Dott. Fausto Marcantoni

ANNO ACCADEMICO 2006 / 2007

INDICE

1. - Introduzione

Il monitoraggio degli apparati elettronici è una attività assolutamente critica in ordine al mantenimento di una infrastruttura tecnologica o, in senso più generale, al fine di assicurare quella che nel gergo anglosassone viene indicata come “business continuity”.

Questa attività può apparire banale se si è seduti davanti all’apparato, diciamo davanti al proprio computer, ma può diventare meno semplice se si sta monitorando un apparato remoto; ancor più complesso se si sta monitorizzando un elevato numero di apparati in remoto.

Basti pensare ad un comune scenario, in cui gli amministratori dell’infrastruttura ricevono notizia del malfunzionamento di un servizio di rete da un utente (magari un manager di alto livello dell’organizzazione) e devono avviare ricerche per isolare la macchina o l’apparato causa del malfunzionamento. Si potrebbe trattare di un problema sul servizio di “Exchange” sui server di posta aziendale, oppure di un’avaria su un apparato di rete, o semplicemente di un hard disk che ha raggiunto la sua massima capacità e comincia a causare strani problemi al server su cui è montato.

Con la sempre crescente penuria di personale tecnico da impiegare nell’amministrazione dell’infrastruttura, soprattutto in grandi organizzazioni, è praticamente impossibile controllare manualmente regolarmente tutti gli apparati in uso. Con il crescere delle dimensioni e della complessità dell’infrastruttura sarebbe utile anche riuscire ad avere dei “warning” circa degli eventi o circa il superamento di soglie critiche che possano far prevedere malfunzionamenti, così che gli amministratori possano intervenire prima che l’utente sperimenti a sue spese il malfunzionamento e il telefono di “helpDesk” sia bombardato di telefonate.

Un software per il monitoraggio dei sistemi e della rete è quindi fondamentale per supportare gli amministratori dell’infrastruttura nell’individuazione (auspicabilmente precoce !) dei problemi. Lo scopo di questa classe di software è di informare gli amministratori in maniera

tempestiva di eventi sospetti (WARNING) o di condizioni di avaria conclamate (CRITICAL); i criteri o le soglie utili a discernere queste diverse condizioni sono definiti dagli amministratori stessi in fase di configurazione del software; tipicamente il software mette a disposizione delle videate riassuntive (attraverso interfaccie web personalizzabili) che evidenziano le condizioni di operatività di sistemi, apparati e servizi presenti all'interno dell'infrastruttura, utilizzando artifici grafici (colori verde, giallo o rosso) significativi (fig. 1), o permettendo l'invio automatico di e-mail e/o SMS a selezionate qualificate utenze.

Nelle figure farò riferimento al software di monitoraggio open source Nagios, uno dei più diffusi sul mercato

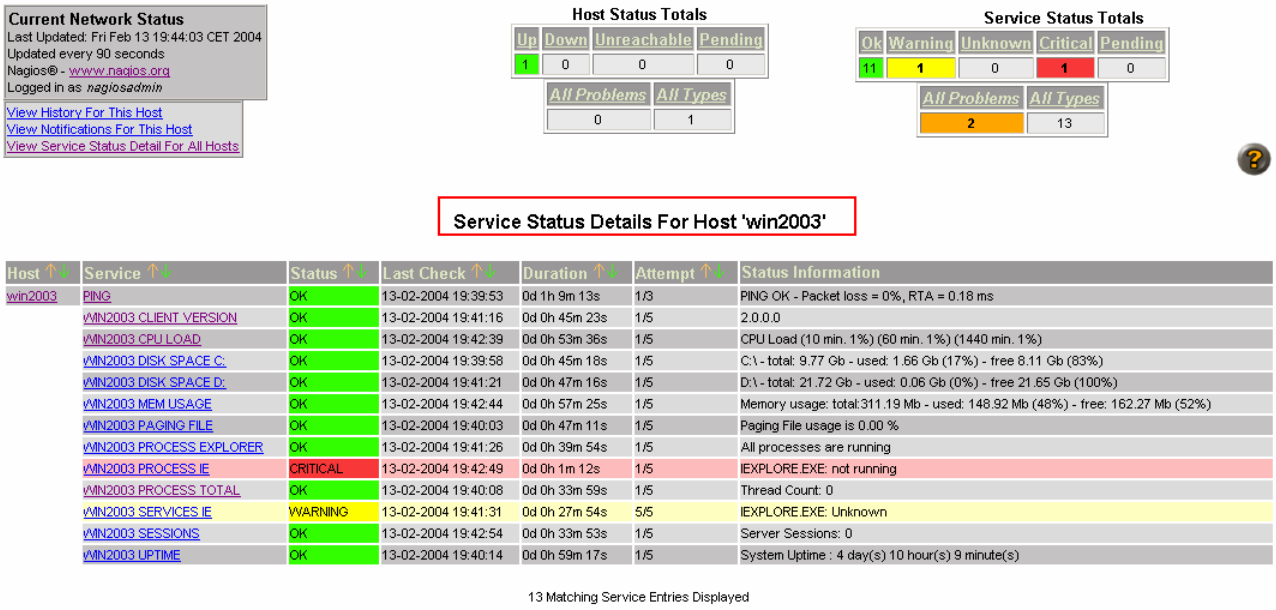

Fig. 1 – Una finestra di status disponibile con Nagios.

Naturalmente anche la conservazione e storicizzazione dei dati di monitoraggio può essere utili a fini statistici e di predizione di situazioni anomale (fig. 2).

Fig. 2 – Una finestra di statistiche disponibile con Nagios.

Per determinare l'operatività dell'infrastruttura, i sistemi di monitoraggio effettuano dei test diversificati per apparati e servizi. Per un apparato server un test comune è quello che verifica la raggiungibilità dello stesso sulla rete (un semplice "ping"), ma vanno verificati anche i servizi offerti dal server stesso o in generale disponibili sulla rete (HTTP, SMTP, DNS, ecc.), o i processi in esecuzione, il carico della CPU o gli "event viewer" del server stesso. Analogamente vengono effettuati test su servizi di rete, per esempio controllando se specifiche porte sono aperte e se i servizi associati sono in ascolto; ciò talvolta potrebbe essere non sufficiente ad avere un reale riscontro dell'operatività del servizio e quindi ulteriori test possono essere compiuti per verificare il corretto funzionamento e il tipo di risposte fornite a specifiche interrogazioni e richieste di servizio.

Fra le caratteristiche più rilevanti per questa tipologia di software c'è senz'altro la modularità, intesa come la capacità di disporre di un "core" applicativo e dalla possibilità di potere acquistare e/o sviluppare plugin che permettono di gestire in un ambiente comunque integrato specifiche esigenze di monitoraggio dell'ambiente; è una tendenza ormai consolidata e affermata in tutti i comparti tecnologici, soprattutto in un'ottica di contenimento dei costi di acquisizione ed esercizio dei sistemi. Peraltro spesso attraverso Internet è possibile condividere in comunità di utenti esigenze e soluzioni comuni, accedendo a "librerie" di plugin vastissime ed in grado di soddisfare qualsivoglia necessità. E' il caso, ad esempio, della comunità di utenti di Nagios, che è possibile conoscere sul sito http://www.nagiosexchange.org. In particolare un plugin di Nagios è un semplice programma, o anche un banale script, che restituisce un valore nell'ambito di un set predefinito (OK, WARNING, CRITICAL, UNKNOWN); questo significa che in principio Nagios può essere configurato per verificare qualunque situazione purchè esista uno strumento di misurazione elettronica: la temperatura e l'umidità nella sala server, la quantità d'acqua piovana in una certa area, la presenza di persone in una certa area quando nessuno dovrebbe accedere, e così via.

Per assicurare un corretto scambio di informazioni fra il sistema di monitoraggio ed i vari apparati dell'infrastruttura tecnologica, è necessario fissare un protocollo, cioè un criterio univoco e concordato che regoli il formato e l'ordine dei messaggi scambiati tra le entità comunicanti, così come le azioni che hanno luogo a seguito della trasmissione e/o ricezione di un messaggio o di altri eventi. Nel complesso mondo di router, switch e server, il protocollo che si è imposto è il Simple Network Management Protocol (SNMP); introdotto nel 1988, successivamente cresciuto con versioni via via più complete, è diventato lo standard per la gestione remota di quella tipologia di apparati e servizi.

Da quanto sopra esposto credo sia chiaro il carattere strategico delle attività di monitoraggio, sia per quanto riguarda una corretta ed economica gestione dell'infrastruttura, ma

anche ai fini di garantire un elevato grado di operatività dei sistemi, in altri termini assicurare la “business continuity”.

Nella trattazione che segue entreremo nel merito della filosofia alla base delle attività di monitoraggio dei sistemi e della rete, di come questa attività si integra nel contesto più ampio di “gestione” dell’infrastruttura tecnologica, dei processi che la caratterizzano; poi verrà descritto il protocollo SNMP, sopra citato, standard indiscusso in materia, delle sue caratteristiche e delle semplici operazioni che il protocollo mette a disposizione per la gestione remota di apparati e servizi. Infine verranno analizzati alcuni dei più diffusi software applicativi open source disponibili sul mercato, con un approfondimento particolare per Nagios e per la sua architettura di plugin, un esempio significativo di implementazione di software modulare.

2. - Monitoraggio dei sistemi e della rete

Il termine monitoraggio deriva dal latino monitor –oris, derivato di monere, con il significato di ammonire, avvisare, informare, consigliare. Il termine ha origine in ambiente industriale, per indicare la vigilanza continua di una macchina in funzione, mediante appositi strumenti che ne misurano le grandezze caratteristiche (velocità, consumo, produzione, ecc.). Il significato originale si è ampliato: dalla macchina all'intero processo, a tutta una struttura operativa, includendo in essa anche le risorse umane. Nel contempo il suo uso si è diffuso in tutte le discipline, sia tecniche che sociali, sempre con il significato generale di rilevazione di dati significativi sul contesto interessato.

L'attività di monitoraggio va programmata, predisponendo i valori assoluti o i valori di soglia o gli indicatori, o i valori desiderati che, in continuo o ad intervalli regolari, vengono usati per confrontare l'andamento del contesto che viene monitorato. Il livello di accuratezza dell'attività di monitoraggio può variare in funzione del livello di criticità del sistema sotto osservazione: il sistema di controllo della strumentazione in uso nella sala di rianimazione o nell'unità di terapia intensiva di un ospedale, o il sistema di controllo delle centrali elettriche o di quelle nucleari, avrà requisiti di criticità decisamente più marcati rispetto ad altre realtà.

2.1 – Il monitoraggio come componente della gestione

In un corretto approccio di tipo organizzativo, il monitoraggio dei sistemi va inquadrato come parte della gestione complessiva degli stessi: è una attività caratterizzante di tale gestione, senza la quale risulterebbe impossibile poter utilizzare efficacemente ed economicamente sistemi e apparati di una certa complessità.

Ci sono due tipologie fondamentali di monitoraggio, in relazione allo scopo dell'attività:

- real-time: sostanzialmente monitoraggio di eventi e/o guasti; verifica ogni cambiamento imprevisto dello stato di sistema, che sia la caduta di un sottosistema o il superamento di

una soglia di valore prevista. Elementi caratterizzanti sono il controllo ininterrotto del sistema, la trattazione e disseminazione istantanea dell'informazione, quale ad esempio l'immediata notifica dell'evento ad un responsabile. Tale situazione permette solo azioni di "reazione" all'evento occorso, che normalmente significa che il comportamento imprevisto ha già avuto luogo;

- storico: sostanzialmente monitoraggio della performance del sistema, finalizzato alla generazione automatica di statistiche su elementi salienti dello stesso (disponibilità, utilizzo, prestazione). Si prevede una fase di collezione e memorizzazione dei dati caratteristici, seguita dalla generazione di grafici che relazionano fra loro i dati disponibili, l'andamento nel tempo delle grandezze rilevate, o la sequenza storica di eventi avvenuti nel passato. Tale attività costituisce la base per prevedere il comportamento dei sistemi in futuro, per identificare azioni preventive per prevenire interruzioni di servizio o avarie (ad esempio prevedere esigenze di risorse aggiuntive per taluni servizi/apparati).

La classificazione in categorie, sopra citata, non significa che le due attività siano incompatibili tra loro; al contrario i dati provenienti dal monitoraggio real-time saranno opportunamente utilizzati quale base per la trattazione storica degli eventi; vanno però tenuti in mente i diversi obiettivi delle due tipologie di attività e la loro semplice integrazione non fornisce riscontri sempre soddisfacenti; ad esempio, la semplice storicizzazione dei dati real-time senza alcun criterio di correlazione di per sé renderà disponibili solo enormi insiemi di valori, invece di utili e finalizzate statistiche.

Ai fini concettuali si può ora definire il confine tra l'azione di monitoraggio e quella di gestione: la prima si può definire come il processo di raccolta ed organizzazione dei dati riguardanti un sistema e/o un apparato, la seconda quale il processo decisionale riguardo gli stessi, nonché le azioni di controllo necessarie, basate sui dati resi disponibili; il ciclo di controllo fra monitoraggio, gestione ed oggetto target è rappresentato in Fig. 3. La distinzione è importante per definire quali azioni appartengono al monitoraggio e quali alla gestione.

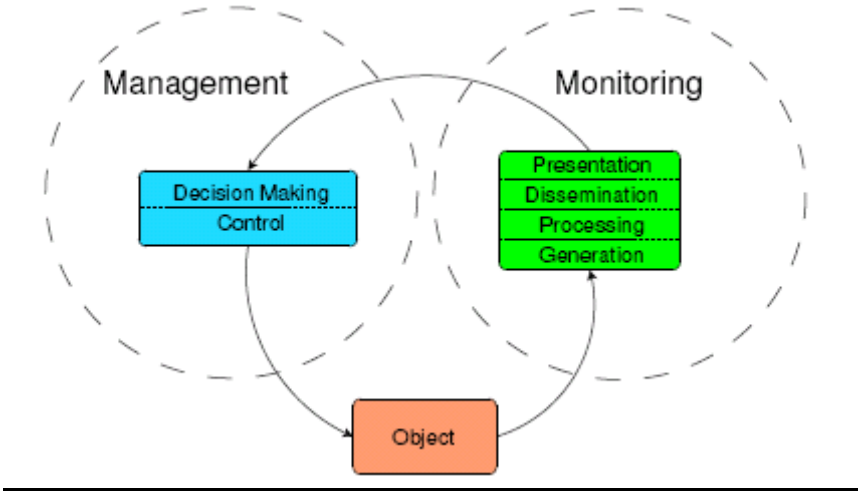

Fig. 3 – Correlazione tra Gestione e Monitoraggio.

2.2 – Un modello di monitoraggio

Il modello funzionale di un efficiente sistema di monitoraggio prevede quattro fasi distinte: Generazione, Elaborazione, Disseminazione e Presentazione dei dati.

- Generazione: individuazione di un evento e generazione della relativa reportistica ;
- Elaborazione: applicazione di semplici funzioni di trattazione dati, quali validazione, correlazione e filtro. L'obiettivo è di convertire i dati grezzi e di basso livello in dati strutturati e con un adeguato livello di dettaglio;
- Disseminazione: distribuzione della reportistica generata all'appropriato livello di persone o ai software che provvedono ad organizzare i dati;
- Presentazione: visualizzazione delle informazioni di monitoraggio raccolte e trattate agli opportuni end-user delle stesse.

Nel seguito entreremo nel merito delle fasi del modello di monitoraggio sopra introdotto.

2.3 – Il processo di generazione dei dati di monitoraggio

L'inizio del processo di monitoraggio consiste nella generazione dei dati; questo può essere fatto in diverse maniere. Semplicemente facendo di tanto in tanto alcune azioni manuali, per esempio un ping su una macchina oppure facendo login su una macchina ed eseguendo alcuni

comandi, o ancora controllando il funzionamento di un servizio collegandosi ad un sito web o scaricando file via FTP, e così via. Sebbene questo tipo di controlli sia molto comune fra gli amministratori preposti ad una infrastruttura tecnologica di dimensioni contenute, quando la complessità e la numerosità degli apparati cresce è inevitabile automatizzare la generazione dei dati di monitoraggio, attraverso l'utilizzo di moduli software specializzati o attraverso l'impiego di log di status o di evento di terze parti.

In base alla dislocazione da cui si effettua il monitoraggio, esso si differenzia in locale o remoto. Nel caso di monitoraggio locale è l'oggetto stesso di monitoraggio che effettua le azioni necessarie ad estrarre le proprie informazioni di status, prendendosi carico di tutte le relative attività; questo avviene attraverso script o programmi eseguiti periodicamente (per esempio con l'aiuto del servizio cron), attraverso uno specializzato daemon di monitoraggio, o attraverso un sottosistema hardware dedicato. Le informazioni raccolte possono essere trasmesse periodicamente (o in alternativa solo al verificarsi di certe condizioni) al sistema che raccoglie i dati e li memorizza per la successiva trattazione.

Quando si fa riferimento al monitoraggio remoto, invece, si intende che l'attività di supervisione è condotta obbligatoriamente dall'esterno, utilizzando un server dedicato e nel rispetto di policy specifiche. Questo può avvenire rimanendo completamente all'esterno della realtà monitorata, ad esempio controllando passivamente il traffico di rete, oppure interagendo con essa, ad esempio controllando attivamente lo status di un host. A questo fine può anche essere installato in via accessoria un agente locale all'apparato monitorato, ma con la differenza che esso sarà collegato al server di monitoraggio e da esso controllato. La comunicazione fra server ed agente può essere organizzata in modalità "pulling", quindi il server periodicamente richiede informazioni (probe) attraverso il suo agente all'oggetto monitorato, che le restituisce. In aggiunta, quando necessario, sarà l'agente stesso ad inviare un allarme (trap) al server.

Oltre a questa diversificazione basata sulla dislocazione da cui viene effettuato il monitoraggio, un altro aspetto rilevante è la modalità attraverso cui l'oggetto monitorato ed il server di monitoraggio comunicano. Fra i vari possibili scenari di comunicazione, quello più comune vede l'utilizzo dello standard SNMP (Simple Network Management Protocol)[1]; questo standard, nato per il monitoraggio di apparati di rete, con gli sviluppi definiti nelle varie versioni soddisfa anche più esigenze di gestione dell'infrastruttura tecnologica.

Per quanto specificamente relativo allo status della rete, si fa riferimento al Internet Control Message Protocol (ICMP) definito nella RFC 792. Per esempio la connettività e le prestazioni della rete possono essere verificati in modalità attiva attraverso dei ping, o in modalità passiva memorizzando i messaggi di errore ICMP come "destination unreachable".

Al di là dei metodi di comunicazione implementati, particolare attenzione va posta nel ridurre al minimo l'impatto dell'attività sulle prestazioni dell'oggetto monitorato e nel minimizzare il carico di traffico sulla rete.

A questo punto andiamo a vedere quali sono gli oggetti su cui viene effettuata l'attività di monitoraggio; la tabella in fig. 4 propone una panoramica, unitamente alla specifica di quale aspetto viene controllato.

Nella tabella vengono riportati solo i valori di quelle grandezze che potenzialmente possono variare; in effetti insieme ai dati di monitoraggio possono essere raccolti anche altri dati statici, relativi ad esempio a tipo ed identificativo di CPU, il costruttore dell'hard disk, la sua capacità, la quantità di RAM, l'indirizzo MAC, ecc, che pur esulando dagli obiettivi dell'attività di monitoraggio, sono estremamente utili per quanto riguarda la raccolta automatica dei dati utili al controllo di configurazione dell'ambiente. Naturalmente la raccolta di questi dati provoca un carico di traffico aggiuntivo sulla rete che deve essere valutato.

[1] Lo standard SNMP è definito da numerose Request For Comment (RFC), il dettaglio della versione 3 di SNMP attualmente in uso può essere trovato nelle RFC da 3410 a 3418 sul sito web del Internet Engineering Task Force (IETF) – http://www.ietf.org/rfc/.

Classificazione	Oggetto	Stato da monitorare
Specifici apparati hardware	Wireless LAN access point	Funziona ?
	Stampante	Carta e livello toner
	Sala server	Temperatura
	Rack	Temperatura, ventole, ecc.
	Switch di rete	Funziona ? prestazioni
Computer	Hard disk	Spazio libero, valori SMART[2] (temperatura, presenza di difetti, ecc.)
	CPU	Utilizzo, temperatura
	RAM	Utilizzo, presenza difetti
	Controller di rete, collegamenti	Funziona ? prestazione (velocità, throughput, latenza, larghezza di banda), indirizzo IP, risoluzione DNS
	Scheda grafica	Temperatura della GPU
	Ventole (scheda madre, case, alimentatore)	Velocità di rotazione
	Alimentatore	Tensione fornita
	UPS	Carico ?
	Scheda madre, case	Temperatura
Processi software	Sistema operativo	Uptime, carico, processi in esecuzione, I/O, file di log (eventi, errori), account utenti, licenze software
	Servizi (Http, FTP, DB, DNS, e-mail (POP, SMTP, IMAP), file server (NFS e/o DFS))	Disponibilità, corretta esecuzione, numero di connessioni, tempi di risposta, coda, ecc.

Fig. 4 – Oggetti da monitorare.

[2] Acronimo per Self-Monitoring, Analysis and Reporting Technology, uno standard open per lo sviluppo di sistemi automatici di verifica dello stato di salute di hard disk, e di potenziali criticità degli stessi, al fine di prevenire il crash degli stessi e la perdita dei dati.

2.4 – L’elaborazione dei dati di monitoraggio

Prima di tutto i dati di monitoraggio raccolti devono superare una fase di validazione, tesa a verificare la plausibilità e correttezza degli stessi; ciò può essere fatto con varie modalità, ad esempio andando a verificare che i dati di identificazione dell’apparato siano quelli attesi, oppure che il time-stamp sia corretto. I dati che non superano questi test vengono scartati.

Un’altra classe di valori non validi sono quelli fuori da specifici range, e che quindi sono ovviamente invalidi, come un carico di elaborazione sulla CPU superiore al 100% o una velocità di rotazione della ventola doppia del suo massimo. Le cause che portano a tali errori possono essere diverse e la loro trattazione dipende dalle strategie di analisi adottate; possono essere ignorati oppure memorizzati per successive indagini, qualora si ripetano con una certa frequenza, allo scopo di individuare le opportune azioni riparatrici.

I dati di monitoraggio vengono memorizzati in un database, così da poter facilmente utilizzarli per le successive fasi di elaborazione, differenziando l’archiviazione in funzione della finalità real-time o storica delle informazioni; in quest’ultimo caso può essere previsto l’utilizzo di apposite procedure che sintetizzino la grande mole di dati real-time per estrarre delle informazioni storiche in grado di riassumere l’andamento delle grandezze sotto osservazione.

Il passo successivo è quello della correlazione dei dati fra loro; i dati generati, validati e memorizzati sono fin qui poco significativi, fino a quando non vengono messi in relazione con dei valori di riferimento attesi, processo che consente di addivenire ad una valutazione dello status del sistema. Sostanzialmente si tratta di definire oggettivamente quali siano le condizioni di corretto funzionamento di un apparato e/o un sistema e quali quelle di “errore”.

Le informazioni raccolte danno luogo a degli eventi di classe:

- OK (nessun problema);
- unknown: (nessuna classificazione è possibile);
- evento sospetto (WARNING);

- evento di avaria conclamata (CRITICAL);

Per attribuire ad ogni evento l'appartenenza ad una di queste classi dovranno essere definiti i valori attesi o il comportamento previsto per le singole grandezze, così da poter definire ad esempio un evento WARNING o CRITICAL per la temperatura della CPU troppo alta.

Il criterio più semplice per la classificazione delle grandezze è quello di definire il range entro il quale i valori sono legittimi e quali discostamenti siano sospetti o critici; sempre rimanendo sull'esempio della temperatura della CPU, si definisce il valore X_1 sotto il quale l'evento è OK, sopra il quale e fino al valore X_2 si dà luogo ad un WARNING, per valori oltre la soglia X_2 si classifica l'evento come CRITICAL.

Scenari più complessi richiedono la comparazione del valore in esame con dati precedenti o con valori di altri oggetti, per consentire una corretta classificazione; per esempio se la temperatura della CPU è fra X_1 ed X_2 (WARNING) per y ore, si può innalzare la classificazione a CRITICAL. Anche la semplice assenza di valori sulla misurazione attuale o su quelle precedenti potrebbe dar luogo alla generazione di un evento WARNING, od ancora ad un evento CRITICAL se l'assenza persiste, sempre in funzione della strategia di monitoraggio definita dagli amministratori dell'infrastruttura.

Ulteriori e più complesse correlazioni possono essere previste, per riassumere insieme di valori che diano un'indicazione complessiva dello stato di salute di un sistema; rimanendo sempre sulla temperatura della CPU, un evento di WARNING, oltre a poter essere valutato in funzione del carico di lavoro corrente della stessa CPU, potrebbe essere confrontato con le temperature rilevate del case e/o del rack, per fissare regole che escludano falsi allarmi e consentano di non innalzare a CRITICAL delle situazioni dovute magari solo al malfunzionamento di un sensore. Questo approccio di "sintesi" dei dati risulta utile anche per evitare la concatenazione di eventi di allarme dovuti alla stessa causa: se uno switch di rete va in avaria, solo la segnalazione dell'evento CRITICAL a suo carico ha significato e non tutti quelli

che in cascata interessano tutto il segmento di rete divenuto irraggiungibile. Un'altra situazione può riguardare dei valori che oscillano velocemente sopra e sotto il valore di soglia, dando luogo ad eventi di WARNING ripetuti, non significativi.

Le strategie di interpretazione dei dati sopra descritte devono essere implementate dagli amministratori del sistema di monitoraggio, assunto che esso lo supporti. La classificazione e correlazione delle informazioni di monitoraggio è essenziale ai fini di un corretto controllo dello stato di salute dell'infrastruttura; l'innalzamento del livello di astrazione associato con le misure di sintesi sopra descritte, consente di non essere sommersi da enormi volumi di informazioni, di poter ragionare solo su misure di reazione a classi di eventi, aggregando i singoli allarmi, razionalizzando gli interventi.

Per ridurre la mole di informazioni elaborate, vengono anche fissati dei criteri di filtraggio dei dati, tanto più efficaci quanto più precoci nel ciclo di vita di tali informazioni; anche la validazione cui sono sottoposti i dati di monitoraggio può considerarsi una modalità di filtro, così come anche il controllo della disseminazione delle informazioni, limitando la spedizione della reportistica solo a chi ne è specificamente interessato.

Un altro aspetto importante dell'elaborazione delle informazioni di monitoraggio è l'analisi delle stesse; l'analisi può essere l'obiettivo primario di un sistema orientato al monitoraggio storico, che abitualmente prevede specifiche funzionalità per questo, ma può essere di ausilio anche per i sistemi di monitoraggio real-time, per prevenire avarie attraverso il controllo dello status di specifiche componenti identificate dall'analisi stessa.

2.5 – Disseminazione delle informazioni

I rapporti scaturiti come risultato dell'elaborazione dei dati di monitoraggio, vanno trasmessi ai diversi destinatari interessati, limitando il flusso ai soli utenti direttamente interessati alle specifiche informazioni contenute nella reportistica, evitando un inutile carico sul sistema e

limitando di fornire ai destinatari informazioni inutili. I destinatari possono essere persone fisiche o ulteriori sistemi di gestione.

Le seguenti azioni possono essere intraprese in presenza di specifici eventi, in funzione della classificazione di essi:

Inoltro dell'informazione ad altri sistemi

- Database del log degli eventi;
- Ulteriori processi di elaborazione;
- Sistema di gestione;
- Modulo di presentazione;

Notifica utente o amministratore – modalità push

- E-mail;
- Messaggio breve di testo (SMS);
- Instant message (IM) oppure segnalazioni sonore se la persona sta lavorando al computer;
- Chiamata telefonica;

La disseminazione delle informazioni ad altri sistemi è piuttosto comune, ma lo scopo primario dell'azione di disseminazione è quello di allertare le persone fisiche, nel caso di informazioni importanti; anche le modalità d'invio cambiano in funzione della classe dell'evento: per eventi di classe WARNING può essere sufficiente una e-mail all'amministratore, mentre un errore di tipo CRITICAL può essere notificato via SMS oppure IM. Le regole che vengono definite per la disseminazione delle informazioni devono quindi fissare il "Chi ?" ed il "Quando ?" informare circa il verificarsi dell'evento, anche in considerazione delle figure tecniche disponibili in turno o in posizione di reperibilità per ottenere l'effettivo intervento di personale qualificato. Possono esse previsti anche meccanismi di "risposta" alla segnalazione di evento dalla persona notificata, così da segnalare al sistema che la trattazione dell'evento è in corso; in assenza di tale riscontro, il sistema di monitoraggio può essere configurato per procedere ad

ulteriori notifiche ad altri livelli di utenti, “scalando” livelli organizzativi, per assicurare l’intervento a salvaguardia dell’infrastruttura.

Una soluzione di questo genere potrebbe prevedere, a fronte di una notifica, una e-mail di risposta, oppure un SMS, oppure marcando l’evento come “in trattazione” sul front-end del sistema di monitoraggio. Naturalmente la trattazione dell’evento può anche consistere nella marcatura dello stesso come “non risolvibile fino a…..” oppure “fuori uso, in attesa di ricambio”, così da rendere evidente che la mancata risoluzione è collegata ad altri vincoli.

Il meccanismo di “scalare” livelli organizzativi può avvenire in via verticale (notificando persone di livello gerarchico o professionale superiore) oppure in via orizzontale (notificando persone diverse del gruppo degli amministratori), prevedendo anche l’utilizzo di metodi di comunicazione diversificati per prevenire anche malfunzionamenti di qualche canale di comunicazione.

2.6 – Presentazione dei risultati

La fase finale dell’attività di monitoraggio è la presentazione delle informazioni, quale risultato delle precedenti fasi di lavorazione. Le modalità con cui ciò avviene dipendono dagli strumenti in uso agli amministratori (dal computer al telefono cellulare al segnalatore sonoro), ma anche dai requisiti da loro definiti circa le modalità di presentazione: può trattarsi di rappresentazioni testuali su una console di sistema o di rappresentazioni grafiche su display specializzati, dotati magari di sofisticate interfacce grafiche. In questo caso il modulo che gestisce la presentazione dovrà controllare la mole di informazioni di monitoraggio, il livello di astrazione delle stesse (e le modalità di salire e scendere di livello di astrazione) e la velocità con cui vengono presentate, assicurando in altri termini un’elevata “usabilità” del sistema. In questo caso infatti, l’amministratore del sistema di monitoraggio potrà focalizzare gli eventi rilevanti e reagire ad esse, senza essere sommerso da elementi irrilevanti.

Sono utilizzate tecniche che rendono le rappresentazioni sui monitor del sistema user-friendly, per esempio dando la possibilità di raggruppare le informazioni per una più efficace rappresentazione, prevedendo messaggi di notifica esplicativi specie quando inviati per e-mail o SMS, visualizzando anche dei tag di priorità nelle liste degli eventi, permettendo di intervenire su quelli più pregiudizievoli l'operatività dell'infrastruttura. Utile anche, soprattutto per il monitoraggio storico, l'utilizzo di diagrammi bidimensionali, con un asse che rappresenta il valore di una specifica grandezza osservata e l'altro asse il tempo, per monitorare i cambiamenti avvenuti.

3. – Il protocollo SNMP

Il Simple Network Management Protocol è il meccanismo più comune usato per il monitoraggio dell'infrastruttura di rete. In questo capitolo entriamo nel dettaglio del suo funzionamento, adottando una visuale orientata all'esperienza pratica.

3.1 – Visione generale

Il protocollo SNMP nasce nel 1990 nell'RFC 1157 e viene definito nella sua prima incarnazione per volontà della IETF (Internet Engineering Task Force) e sotto la spinta di numerose aziende quali IBM e HP presto SNMP diventa lo standard de facto per il controllo degli apparati di rete.

Il protocollo opera a livello applicativo utilizzando UDP per il trasporto ed utilizza un metodo di comunicazione client/server. SNMP consente la gestione e la supervisione di apparati collegati in una rete, rispetto a tutti quegli aspetti che richiedono azioni di tipo amministrativo.

Permette agli amministratori di rete di individuare ed in seguito isolare i componenti difettosi che si possono trovare su una rete, configurare i vari componenti da remoto e monitorare lo stato e le performance della rete. Il suo obiettivo è quello di fornire un modo standard e vendor-indipendent per la gestione degli apparati di rete.

In contraddizione con quanto porta tuttavia a pensare il nome, SNMP non è un protocollo particolarmente semplice: dispone di poche operazioni possibili ma richiede una sua approfondita comprensione per poter essere utilizzato. Inoltre la letteratura su SNMP spesso risulta inaspettatamente più complicata da capire che non la realtà stessa.

3.2 – Caratteristiche e principi di funzionamento

Ruoli

SNMP prevede una chiara e semplice distinzione di ruoli tra gli attori che utilizzano il protocollo.

Solitamente si ha a che fare con:

- Una serie di dispositivi di rete da gestire e monitorare su cui è presente un ”agent”, ovvero un demone che si occupa di rispondere alle richieste SNMP;
- Una NMS (Network Management Station) che è la postazione che si occupa di collezionare i dati provenienti dai vari dispositivi e che spesso coincide con una workstation amministrativa su cui lavorano gli amministratori di rete.

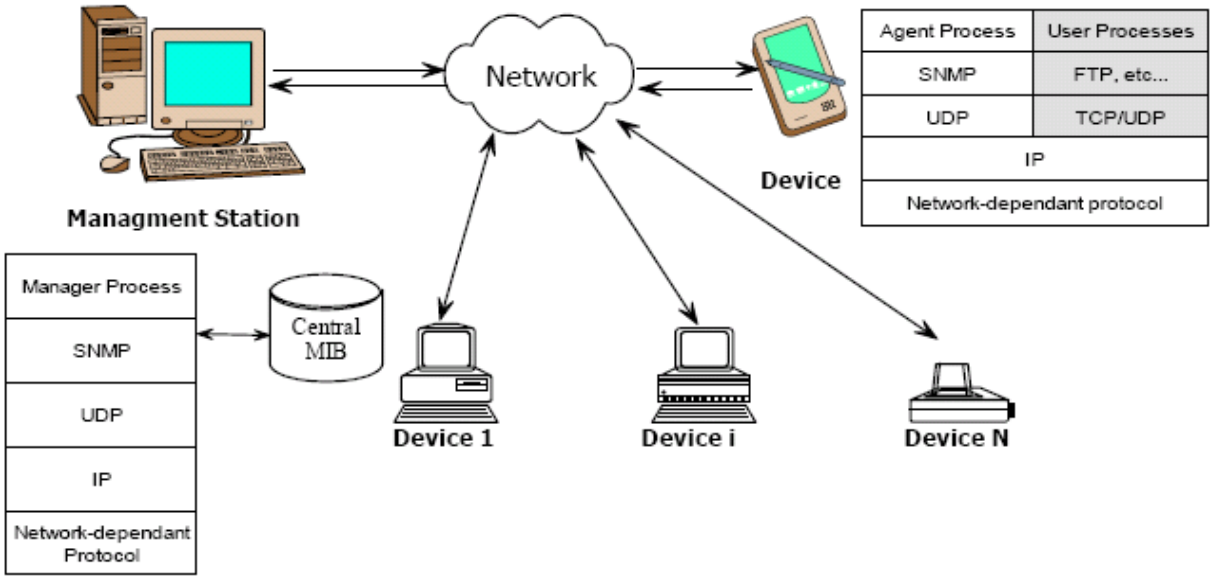

Fig. 5 – Immagine esemplificativa dei ruoli.

La Fig. 5 rappresenta una rete con una NMS che comunica con 3 dispositivi dotati di agent SNMP.

Oggetti ed informazioni

L’agent è in grado di gestire un’ampia gamma di informazioni individuate univocamente da un OID (Object Identifier), controllando gli accessi a queste informazioni e preoccupandosi di mantenerle sempre aggiornate.

Quando si vuole accedere ad un'informazione in realtà si richiedere di accedere a ciò che è individuato da un particolare OID: è pertanto necessario sapere in anticipo la posizione in cui è collocata l'informazione di nostro interesse.

L'accesso ad un OID può essere permesso, secondo il grado di privilegi, in scrittura o in lettura: tali permessi, nella versione v2 del protocollo, sono gestiti attraverso le "communities", delle semplici password in chiaro.

Comunicazione e porte

Come già accennato in precedenza SNMP usa UDP come mezzo di trasporto.

La porta 161 è la scelta di default su cui sono in ascolto gli agent che ricevono le richieste snmp.

La porta 162 è la scelta di default per la NMS destinata a ricevere i traps (vedi paragrafo seguente).

Utilizzi

SNMP può essere utilizzato principalmente in due modi: polling e traps.

Il polling consiste nell'utilizzare o scrivere un'applicazione che interroghi attivamente uno o più dispositivi attraverso operazioni di get per ottenere le informazioni desiderate. La non risposta ad un polling è certamente sintomo di un problema.

L'approccio duale sono le traps: in questo caso chi è incaricato di monitorare la rete svolge un ruolo passivo. Egli rimane in ascolto di notifiche (traps) provenienti dagli agent snmp opportunamente configurati e generate in risposta a precisi eventi.

Una volta ricevuto un trap il "trap host" procede solitamente a memorizzarla ed attiva una politica prestabilita in relazione all'evento che prevede opportune azioni ed eventuali ulteriori notifiche.

Versioni

SNMPv1: La prima versione del protocollo nasce con delle caratteristiche che si manterranno tali anche con lo svilupparsi del protocollo nelle versioni successive:

- Indipendenza dalle architetture di rete e dalle piattaforme utilizzate;
- La semplicità di utilizzo (la prima versione del protocollo prevedeva solo pochi essenziali comandi);
- La trasportabilità;
- L' estensibilità applicativa (in altre parole la possibilità di sviluppare nuove funzioni di management senza, per questo, dover modificare gli Agent);
- L' indipendenza dalle architetture delle reti o delle piattaforme utilizzate;
- La robustezza dovuta alla semplicità del protocollo di trasporto UDP e alla facilità;
- di testing che questo implica (non necessita di multithreading per gestire le connessioni);
- Indipendenza da altri servizi di rete;
- Applicabilità a basso costo a tutti i devices, nuovi ed esistenti (almeno per il lato Agent);

SNMPv2: è presente in numerose versioni. La più diffusa è SNMPv2c che tuttavia non è ancora formalmente standardizzata da IETF ed è ritenuta ancora sperimentale.

In SNMPv2c, tra le novità rispetto a SNMPv1, sono state aggiunte alcune operazioni che rendono più semplice l'accesso ai MIB.

Tali operazioni, che verranno approfondite nel proseguio, sono:

- GetBulk Request, che ha permesso di ridurre sensibilmente l'overhead del traffico SNMP
- Inform

Un'altra miglioria che stata introdotta con SNMPv2c l'ampliamento del numero dei possibili messaggi di errore. Spesso in SNMPv1, i messaggi di errore erano generici e poco chiarificatori; nel v2c, invece, se ne sono aggiunti di nuovi, così da permettere al Manager di avere un'idea più chiara dei problemi in cui è incorsa la PDU che ha inviato.

SNMPv3: è stato introdotto per migliorare la sicurezza del protocollo SNMP.

Esso introduce autenticazione e cifratura avanzate e proprio perciò richiede notevoli risorse applicative: per questo motivo risulta ancora scarsamente implementato e diffuso.

3.3 – MIB e OID

Gli OID, o Object Identifiers, sono un modo per individuare univocamente ed accedere al un valore di un parametro reso disponibile dall'agent SNMP.

I MIBs, o Management Information Bases, forniscono una corrispondenza uno-a-uno tra un OID ed una descrizione dell'OID in una forma comprensibile e più significativa per un essere umano.

Prima di fornire un esempio vale la pena spendere qualche parola in più sull'organizzazione dei dati in SNMP.

A differenza degli usuali database relazionali che si è soliti usare con SQL, un MIB è database gerarchico in cui la sua definizione coincide con la struttura.

Come ogni database gerarchico le informazioni sono organizzate ad albero: ogni ricerca parte dalla radice e percorre progressivamente ogni nodo intermedio fino a giungere alle foglie, luogo in cui è contenuta l'informazione.

E' pertanto necessario conoscere la posizione "fisica" dell'informazione per potervi accedere.

Concettualmente è più simile ad un file-system che ad un database.

L'OID si occupa quindi di descrivere la collocazione ed è costituito da una sequenza di numeri. Il punto iniziale individua la radice, i successivi numeri i nodi intermedi, l'ultimo numero l'informazione.

Ad esempio il percorso per conoscere l'informazione sull'uptime del sistema è:

OID:

.1.3.6.1.2.1.1.3.0

Il MIB per ogni nodo espresso in forma numerica definisce anche un'etichetta (o "label") che lo identifica.

E' quindi possibile accedere all'oggetto utilizzando una più comprensibile e significativa stringa in corrispondenza biunivoca con l'OID.

Pertanto l'informazione accessibile attraverso l'OID qui sopra risulta raggiungibile anche attraverso il percorso:

.iso.org.dod.internet.mgmt.mib-2.system.sysUpTime.0

Lo 0 finale individua la prima, ed in questo caso unica, istanza dell'oggetto.

Analogamente se volessimo ottenere un nome identificativo della macchina esaminata dovremmo consultare l'oggetto sysDescr accessibile attraverso:

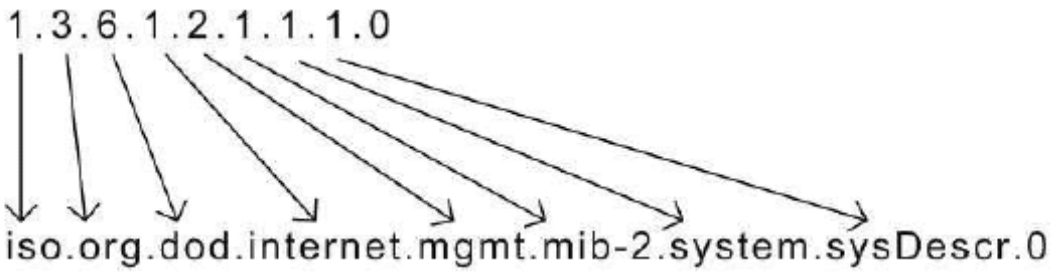

Fig. 6 – In evidenza la corrispondenza numerica-OID – text OID.

Organizzazione dei dati

La radice del MIB contiene le organizzazioni che promuovono standard: ccitt(0), iso(1), joint-iso-ccitt(2).

Al di sotto del nodo iso(1) c'è il nodo org(3) che individua le organizzazioni.

Al di sotto di questo il nodo dod(6) del Dipartimento della Difesa.

Sotto di questo vi è il nodo internet(1) in cui ci sono le informazioni che riguardano "internet".

Sottoalbero	Descrizione
directory(1)	OSI directory
mgmt(2)	RFC Standard MIBs
experimental(3)	Internet sperimentale
private(4)	Ogni produttore definisce i suoi MIB qui
security(5)	Sicurezza
snmpv2(6)	Impostazioni interne SNMP

Fig. 7 – Sottoalberi del nodo internet (1).

Il MIB di interesse per la gestione dei dispositivi di rete è il mgmt (2) al cui interno è presente un unico nodo il mib-2 (1)[3].

Il mib-2 è il MIB di riferimento, definisce tutti gli oggetti fin qui esposti dalla radice fino alla collocazione del mib-2 stesso (insomma definisce elementi dell'albero fino al mib).

Quando in generale si parla di MIB in realtà si dovrebbe parlare di moduli-MIB: infatti l'unico MIB è il MIB-II e tutti gli altri sono organizzati in riferimento a questo.

MIB-II è l'attuale MIB standard per Internet, è stato definito nel 1991 nell'RFC 1213 e contiene 171 oggetti. Tali oggetti sono stati raggruppati per protocollo (ad esempio TCP, IP, UDP, ecc) e per categoria (ad esempio System e Interfaces).

Nel MIB-II sono pertanto presenti numerosi sottonodi che contengono informazioni di nostro interesse:

system(1)
interfaces(2)
at(3)
ip(4)
icmp(5)

[3] Noto anche come MIB-II

tcp(6)

udp(7)

egp(8)

cmot(9)

transmission(10)

snmp(11)

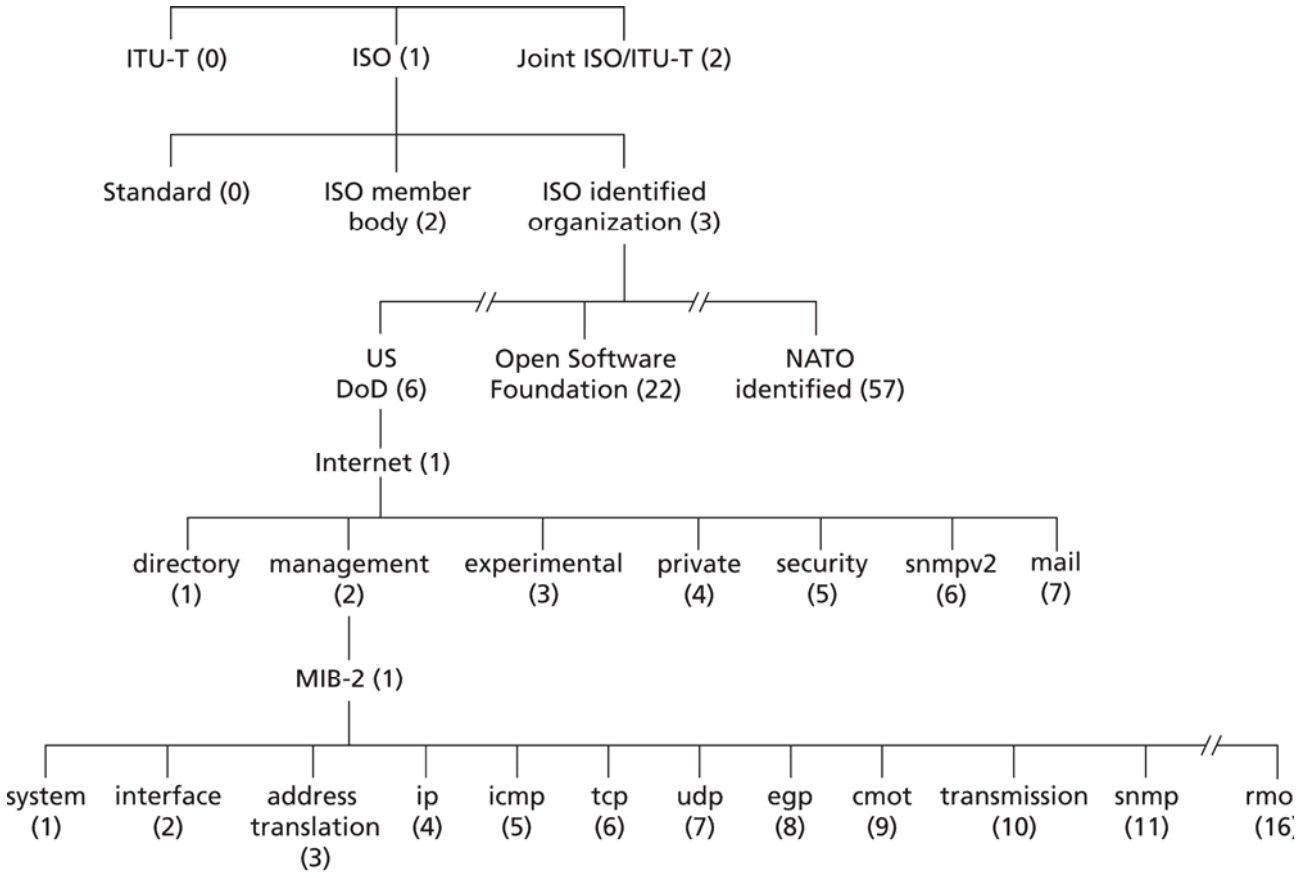

Fig. 8 – Schema della struttura del MIB.

Formato dei dati

Per consentire l'interoperabilità delle varie implementazioni di agent e manager, è necessario definire senza alcuna ambiguità:

- la struttura della MIB (in modo che sia univoco il mapping tra un object identifier e la semantica associata)
- la rappresentazione binaria dei valori scambiati

Tali obbiettivi sono realizzati grazie all'adozione di ASN.1[4] e BER[5].

SNMP utilizza alcuni tipi di dati definiti da ASN.1:

- Simple Type: un tipo semplice è definito specificando direttamente l'insieme dei suoi valori. Sono tipi atomici. Tutti gli altri tipi sono definiti in base ai tipi semplici.
- Structured Type: un tipo strutturato è costituito da componenti e permette di costruire tipi di dati complessi.

I dati in snmp risultano pertanto essere tutti e soltanto:

- valori scalari, atomici
- oppure tabelle: una struttura bidimensionale di semplici dati scalari definita come sequenza di valori

Per ogni dato scalare risultano definiti:

- SYNTAX: un tipo ASN.1 (e.g. Integer, DisplayString, Counter, IpAddress, PhysAddress, etc)
- ACCESS: tipo di accesso consentito (readwrite, read-only, write-only, notaccessible)
- STATUS: supporto richiesto alle implementazioni (mandatory, optional, deprecated, obsolete)
- DESCRIPTION: serve a fini di documentazione

Esempio relativo a sysDescr di definizione di un oggetto SNMP secondo ASN.1:

mib-2 OBJECT IDENTIFIER ::= { mgmt 1 } -- definizione della MIB II
system OBJECT IDENTIFIER ::= { mib-2 1 }
sysDescr OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
ACCESS read-only
STATUS mandatory
DESCRIPTION
"A textual description of the entity. This value
should include the full name and version
identification of the system's hardware type,
software operating-system, and networking
software. It is mandatory that this only contain

[4] Abstract Syntax Notation One.
[5] Basi Encoding Rules.

printable ASCII characters."
::= { system 1 }

3.4 – Operazioni

Le operazioni di cui SNMP dispone sono:

Get Request

Utilizzato per richiedere valori specifici all'agent utilizzando il comando get.

GetNext Request

Viene utilizzato per percorrere un sotto albero del MIB in ordine lessicografico ed ottenere tutti i valori in esso contenuti.

GetBulk Request

Si effettua un'unica richiesta che interessa una grande quantità di dati per prelevare più OID con il minimo numero di pacchetti.

Set Request

Permette di configurare un valore o dare una comando al dispositivo.
Es: si può impostare il valore ip per una data interfaccia ma anche provocare il reboot della macchina modificando un'opportuna variabile.

Get Response

Questo comando è utilizzato dal device per rispondere a richieste di tipo Get/Set ricevute dal manager.

Trap Message

Notifica asincrona da parte di un agent al manager in seguito al verificarsi di un evento.

Inform

Introdotto da SNMPv2 è un trap con attesa di conferma della ricezione da parte dell'agent.

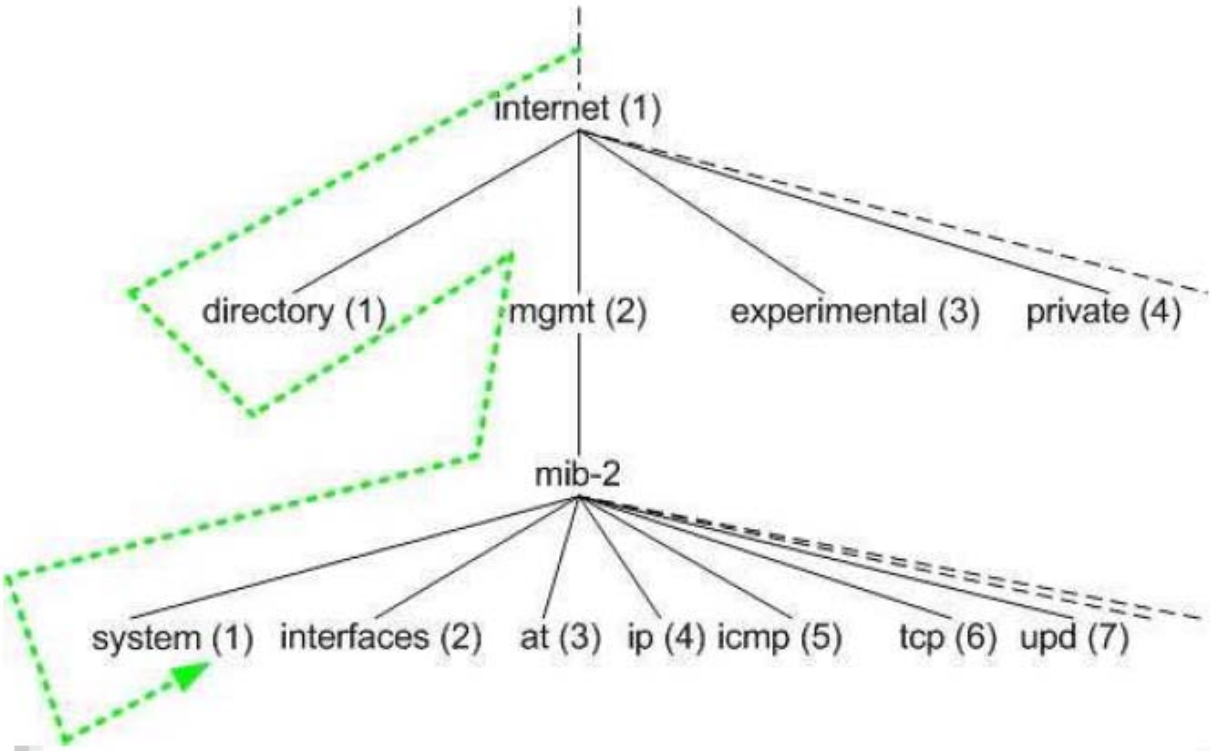

Fig. 9 – SNMP ”walk” sul mib-2 usando get-next.

3.5 – Evoluzione

SNMP ha sempre più adepti nel mondo dei produttori di apparati e software per la rete, quindi si rendono disponibili sul mercato sempre nuovi prodotti SNMP compatibili.

Con l'ampia copertura degli aspetti relativi alla sicurezza, specialmente con la disponibilità della versione 3, molti amministratori di rete usano prodotti basati su SNMP per il monitoraggio della loro infrastruttura.

In considerazione della facilità con cui è possibile aggiungere nuovi elementi nella MIB, assistiamo alla registrazione di nuovi oggetti non solo hardware, ma anche software, basati su SNMP. Questo è un aspetto molto importante ai fini del monitoraggio dei moduli di sistema operativo ed altri software, quali databases o daemons.

4. – Stato dell'arte delle applicazioni per il monitoraggio

Oggi esistono sul mercato molte implementazioni e suite applicative basate su SNMP; le caratteristiche che le differenziano sono in relazione al tipo di monitoraggio verso cui sono orientate. Ci sono prodotti che mettono il focus sulla rete, che significa una particolare facilità a monitorare l'infrastruttura di rete (switches, routers, ecc.), e tipicamente offrono un elevato supporto allo standard SNMP. Altri tipi di applicativi sono orientati a specifici task, per esempio al controllo di cluster di computer, oppure a specifiche componenti hardware o software.

Altri prodotti sono invece a più largo spettro, per coprire esigenze diverse in ambienti eterogenei (Unix, Linux, Windows, Mac OS, ecc.), o per il supporto di hardware / software proprietari (sistemi di storage, sistemi di controllo accessi, impianti di condizionamento, apparati embedded, controllers, ecc.).

Nella trattazione che segue, esamineremo alcuni fra i più diffusi software open-source, senza soffermarci sulle grandi suite applicative proprietarie, quali HP OpenView[6] (NNM[7]) o IBM Tivoli[8] (NetView[9]). Queste suite proprietarie hanno caratteristiche tali e ricchezza di funzionalità da soddisfare la totalità delle esigenze, ma per contro hanno costi di acquisto elevati e per la loro complessità richiedono competenze molto specialistiche, con relativi costi di formazione del personale dedicato al loro utilizzo.

Dopo una valutazione delle caratteristiche di alcuni fra i più diffusi software open-source sul mercato, ci soffermeremo su quello che per filosofia di progettazione e per popolarità nella comunità Internet si è distinto: Nagios.

[6] Open View Monitoring: http://openview.hp.com/solutions/nsm/
[7] Network Node Manager: http://openview.hp.com/products/nnm/
[8] Tivoli Monitoring: http://www.ibm.com/software/tivoli/products/monitor/
[9] NetView: http://www.ibm.com/software/tivoli/products/netview/

4.1 – Selezione di applicazioni per il monitoraggio

Big Brother

Questo prodotto, pur essendo ancora diffusamente utilizzato soprattutto in ambienti scientifici, risulta piuttosto sorpassato; alcuni aspetti di architettura del prodotto unitamente alla pesante modalità di "scripting" adottata, comportano un elevato impatto in termini di personale da dedicare alla gestione ed alla manutenzione del sistema.

Nell'esperienza degli utilizzatori si riscontrano:

- uno scarso supporto per il monitoraggio storico, con funzionalità limitate e riscontri poco significati per l'osservazione dei trend;
- una modalità di notifica degli eventi poco intuitiva, soprattutto per quanto riguarda il testo delle e-mail inviate, breve e criptico, che richiede personale addestrato per leggerlo. Questo aspetto può diventare critico, quando l'amministratore deve circoscrivere la causa del problema ed identificare le azioni correttive;
- la scarsa flessibilità nel pianificare i task di monitoraggio. Ad esempio il controllo sulla connettività (attraverso dei ping) veniva eseguito esattamente ogni 5 minuti, con brevi periodi di sovraccarico della rete, invece di una opportuna distribuzione costante ma bassa del carico;
- un elevato numero di file di log creati, che soprattutto nelle fasi iniziali di utilizzo rendono confusa e pesante la gestione del prodotto;

Le più recenti versioni stanno andando a sanare alcuni dei problemi evidenziati.

Big Sister

Si tratta di un alternativa open-source molto simile alla precedente, su cui ci sono attività di sviluppo ed è compatibile con Big Brother; ciò significa anche, però, che riporta le stesse debolezze del precedente, in particolare con la scarsa efficienza dei meccanismi di parallelizzazione dei controlli. Inoltre la scarsa disponibilità di plugins, e la loro limitata

diffusione all'interno della ristretta comunità di users, non consente di considerarlo un prodotto di elevata utilità.

OpenNMS

Si tratta di un applicativo focalizzato principalmente sul monitoraggio della rete, grazie all'elevato supporto dello standard SNMP. E' riconosciuto unanimamente come un buon prodotto, ma è caratterizzato da un'elevata complessità, forse la maggiore fra i prodotti presi in considerazione; l'architettura è basata su Java ed XML e comporta difficoltà nella fase di configurazione iniziale del prodotto. Fra le sue caratteristiche, presenta attitudine per il monitoraggio remoto, in modalità passiva, mentre l'architettura dei plugin si appoggia a quella resa disponibile dall'infrastruttura di Nagios. Si tratta comunque di un prodotto estremamente potente per il monitoraggio e la gestione delle componenti di rete, sotto standard SNMP; per quanto riguarda altre funzionalità va a richiamare i plugin di Nagios.

Ganglia

Si tratta di un'applicazione che è stata sviluppata specificamente per il monitoraggio in ambiente cluster. L'architettura software consiste in un agente che viene installato su 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 ha delle caratteristiche molto interessanti, ad esempio una modalità di scambio delle informazione fra gli agenti molto efficiente, così come una semplicità di installazione e configurazione nell'ambiente. Carente invece risulta la documentazione, che non copre adeguatamente molti aspetti rilevanti nell'utilizzo del prodotto; anche la capacità di modellizzazione dell'ambiente è molto limitata, cosa che comporta per esempio limitazioni della fase di discovery automatica per quei computer non disponibili sulla rete al momento della

configurazione. Ciò in conseguenza di quella carenza di informazioni di descrizione dell'infrastuttura legate alla definizione del modello di lavoro.

In sintesi l'applicazione, così come concepita, è fortemente orientata ad essere utilizzata in infrastrutture di tipo cluster.

Nagios

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. Ciò nonostante si è imposto come la soluzione più apprezzata fra i prodotti open-source, in virtù delle sue caratteristiche e delle elevate prestazioni in grado di assicurare in svariate condizioni d'utilizzo.

Prima di tutto, l'architettura flessibile sopra descritta consente di configurare Nagios, con le sue componenti, in maniera da effettuare il monitoraggio di praticamente qualunque realtà; per esempio il modulo Nagios Remote Plugins Executor (NRPE) consente l'esecuzione di software su una macchina posizionata in remoto, ovunque sulla rete; inoltre il modulo Nagios Service Check Acceptor (NSCA) adotta avanzati criteri di controllo passivo basati sulle "trap". L'elevata flessibilità che caratterizza l'architettura di Nagios, consente di monitorare efficacemente apparati di rete (sia in standard SNMP che non), così come computer client, hardware e software dedicati.

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.

Inoltre è stata particolarmente curata sia la funzionalità di notifica ed allarme, sia quella relativa alla parte di presentazione grafica e visualizzazione dei risultati del monitoraggio, sia per la parte

real-time che per i trend storici, con la possibilità di utilizzare tool aggiuntivi come il Multi Router Traffic Grapher (MRTG)

4.2 – Nagios e l'architettura dei suoi plugin

Le applicazioni per il monitoraggio sopra descritte, pur rappresentando solo un campione della pletora di applicativi presenti sul mercato open-source, sono rappresentative delle caratteristiche più interessanti che auspichiamo avere disponibili nella attività di supervisione dell'infrastruttura tecnologica. In questo senso il prodotto che spicca tra gli altri, sia per la sua architettura di progetto, sia per la sua ampia diffusione in comunità di utenti, è senz'altro Nagios. Per approfondire la particolare flessibilità di tale prodotto, andiamo ora a vedere l'architettura dei plugin di Nagios.

La modalità più semplice per monitorare una rete è quella di far sì che il server preposto a tale azione controlli gli apparati sotto monitoraggio direttamente attraverso la connessione di rete, spedendo delle richieste agli apparati stessi ed elaborando le risposte che riceve, ed avviando le opportune azioni in base ai riscontri ottenuti, oppure in base all'assenza di riscontri, se configurato a farlo. Con questo tipo di approccio, il monitoraggio consiste in un azione di verifica dello stato di funzionamento dell'apparato (check_ping) o se un certo servizio è operativo (check_ssh, check_mail, ecc.). Il vantaggio di questa semplice azione è che la configurazione dell'apparato target non viene intaccata e le modalità di configurazione / interazione sono estremamente semplici.

Se i requisiti di monitoraggio di un apparato richiedono informazioni più complesse, per esempio la temperatura della CPU o il carico del sistema, le modalità di rilevazione devono essere ampliate. A questo scopo sull'apparato target va configurato uno specifico servizio in grado di raccogliere le informazioni necessarie; si tratta del Nagios Remote Plugins Executor (NRPE), che si prende carico sia della trasmissione al server di monitoraggio delle informazioni

sopra citate, sia dell'esecuzione degli ulteriori specifici plugin necessari per raccogliere le informazioni aggiuntive (check_Temperature, check_Load, ecc.). Quindi il modulo NRPE opera come un agente che il server di monitoraggio richiama per la raccolta di tutte le informazioni richieste per il controllo dell'apparato; tale modulo, essendo in grado di eseguire qualunque plugin conforme alla sua API, così come di avviare un'altra istanza NRPE, può anche essere utilizzato non solo in modalità locale all'apparato, ma anche per fungere da gateway remoto verso ulteriori instradamenti. Ciò significa che qualora un apparato da sottoporre a controllo non sia direttamente raggiungibile dal server di monitoraggio (ad esempio perché un client è ubicato all'interno di un altro segmento di rete), il modulo NRPE può essere utilizzato come gateway se installato su un computer che abbia accesso ad entrambi le reti; così il server può controllare lo stato di funzionamento del client attraverso l'effettuazione di un comando check_ping via NRPE. Il server di monitoraggio, quindi, invierà una richiesta per l'esecuzione di un controllo, circa lo stato di funzionamento di un client, al computer su cui è in esecuzione il modulo NRPE, che a sua volta eseguirà il check_ping per conto del server di monitoraggio, restituendogli indietro il risultato ottenuto.

La terza principale possibilità che un server di monitoraggio ha per raccogliere le informazioni, consiste nello svolgere controlli "passivi". Ciò significa che il server viene configurato per rimanere in attesa di uno status report o di uno specifico valore da parte di un apparato, senza intraprendere iniziative attive. Se l'informazione relativa ad un controllo passivo, comunemente chiamato "trap", raggiunge il server, essa viene trattata analogamente a quelle provenienti dai controlli attivi. In funzione della configurazione del server, l'assenza di comunicazione da parte dell'apparato entro un predeterminato periodo di tempo, può essere interpretata come indicatore di corretto funzionamento oppure come segnale di criticità. Il modulo Nagios Service Chek Acceptor (NSCA) che è eseguito come servizio sul server, consente

a Nagios di trattare i valori provenienti da attività passiva e passarli al suo modulo applicativo centrale (core) che procede alla elaborazione successiva.

Per evitare che possano essere trasmesse al server informazioni false o fuorvianti, il collegamento tra il daemon NSCA sul server ed il client può essere criptato; va valutato il livello di cifratura, più o meno forte, in considerazione del carico aggiuntivo sulla rete generato dalla cifratura del traffico ed in funzione dei requisiti di sicurezza definiti in ogni specifica situazione.

Nel contesto di Nagios, la trasmissione di informazioni di stato verso il server di monitoraggio significa che il modulo agente send_nsca sul computer sotto controllo raccoglie le informazioni richieste e le trasmette al daemon NSCA attivo sul server di monitoraggio.

Un altro punto di forza di Nagios è la semplicità di configurazione, che utilizza dei file a base testuale facilmente comprensibili, con procedure di definizione dell'ambiente (host e servizi) molto immediata; la fase di definizione dei comandi di check richiede invece maggiore attenzione e può consentire di eseguire i comandi direttamente oppure utilizzare argomenti da passare al modulo NRPE. Anche l'utilizzo di funzioni di monitoraggio passivo attraverso il modulo NSCA richiede il rispetto di alcune accortezze, come la corrispondenza fra i nomi delle grandezze monitorate e quelli attesi dal server, ad evitare comportamenti erronei difficili da individuare.

4.3 – Il monitoraggio dei server Windows con Nagios

L'ambiente operativo in cui si opera non sempre è composto da apparati omogenei: fintanto il monitoraggio riguarda i soli servizi di rete, la presenza di sistemi operativi differenti non assume rilevanza; quando invece si vogliano raccogliere informazioni relative a specifici server vanno fatte alcune distinzioni. Per i sistemi basati su Unix si possono utilizzare i tool resi disponibili da Nagios (plugin locali, NRPE, NSCA); per i sistemi Windows non è sufficiente utilizzare i plugin nativi, pur eventualmente compilati in ambiente Windows, in quanto ci sono

delle peculiarità dei diversi sistemi operativi che non possono essere comparate fra loro, rendendo non attuabile un approccio di questo genere.

La soluzione a questa situazione è l'installazione sul server Windows un servizio che può poi essere interrogato sulla rete per raccogliere le informazioni richieste; questi servizi sono NSClient e NC_Net. Ciò consente di impiegare gli script nativi di Windows per controllare le risorse locali al server e, opportunamente estesi, restituire un valore della misurazione ed una linea di output testuale, esattamente come fanno i plugin nativi di Nagios.

NSClient è il modulo più vecchio, largamente testato e utilizzato, ma non più oggetto di sviluppo; la sua ultima release risale ad ottobre 2003 ed è utilizzabile su macchine Windows NT, Windows 2000, Windows XP e Windows 2003. Il suo successore è NC_Net e può essere utilizzato per rimpiazzare NSClient senza alterare in nulla la configurazione di Nagios; si tratta di un moderno package sviluppato in ambiente .NET, quindi utilizzabile su sistemi dotati almeno di Windows 2000.

Per installare NSClient è sufficiente prendere il file nsclient_201.zip dal sito di Nagios Exchange[10] ed estrarne il contenuto, verranno così create 2 cartelle la cui denominazione fa riferimento all'architettura dei sistemi: Win_NT4_bin per Windows NT e Win_2k_XP_Bin per Windows 2000 e versioni successive. Copiando il contenuto della cartella relativa al sistema su cui si sta operando su C:\Programs\NSClient, si può quindi installare il servizio:

c:\Programs\NSClient> pNSClient.exe /install

c:\Programs\NSClient> net start nsclient

L'esecuzione di questi comandi permette di installare il servizio (l'esecuzione con lo switch /uninstall lo rimuove), e di farlo partire. NSClient ha 2 parametri, porta (default 1248) e password (default none); questi valori possono essere cambiati direttamente nel registro nella chiave HKEY_LOCAL_MACHINE\SOFTWARE\NSClient\Parms.

[10] http://www.nagiosexchange.org/Windows.49.0.html

Per quanto riguarda l'installazione di NC_Net, è necessario prima di tutto assicurarsi che tutte le versioni precedenti (compreso eventualmente NSClient) siano rimosse; quindi si può reperire il modulo necessario (NC_Net_setup.msi) sul sito del suo creatore Tony Montibello[11]; una volta installato il package msi, è necessario assicurarsi che il servizio sia stato creato e che sia di tipo automatic. NC_Net ha gli stessi parametri di NSClient, porta e password, ma possono essere anche reimpostati dalla console di gestione servizi, nelle proprietà dei parametri di avvio, con una sintassi del tipo:

port 4711 password pippo

Il comportamento di NSClient è sostanzialmente analogo a quello dei plugin standard di Nagios, mentre per NC_net è necessario scaricare dal solito sito web un modulo aggiuntivo del plugin check_nt e compilarlo.

A questo punto è possibile utilizzare in maniera trasparente il plugin standard check_nt che utilizza i servizi in esecuzione sui server sotto controllo, attraverso il Nagios Remote Plugins Executor (NRPE). Riporto di seguito la sintassi dei principali parametri di check_nt:

–H *address*
Indirizzo IP oppure host name della macchina su cui NSClient/NC_Net è installato;

–v *command*
Il comando da eseguire;

–p *port*
Definizione di porta alternativa per NSClient/NC_Net. Il default è porta 1248;

–w *integer*
Definisce un limite di warning. Tale opzione non è disponibile per tutti i comandi;

–c *integer*
Definisce un limite di critical. Anche tale opzione non è disponibile per tutti i comandi;

–l *parameter*
E' usato per passare ulteriori parametri, come il drive nel caso del check di hard disk oppure il nome del processo nel caso di monitoraggio degli stessi;;

[11] http://www.shatterit.com/NC_Net

–s *password*
E' una password di autenticazione richiesta nel caso che NSClient/NC_Net lanciano i rispettivi servizi utilizzando il parametro password;

L'effetto reale dell'esecuzione del plugin check_nt sopra descritto dipende dal comando specificato al parametro -v *command* , e la cui esecuzione non differisce se abbiamo installato il servizio di NSClient o quello di NC_Net.

Riporto a titolo di esempio alcune delle prove fatte:

Verifica della versione installata: – eseguendo il comando

check_nt –H address –v CLIENTVERSION

si ottiene:

```
nagios@linux:nagios/libexec$ ./check_nt -H winsrv -v CLIENTVERSION
NC_Net 2.21 03/13/05
```

Verifica del carico sulla CPU – eseguendo il comando

check_nt –H address –v CPULOAD –l interval,warning_limit,critical_limit

ed inserendo 3 valori rispettivamente per intervallo di tempo da controllare in minuti, soglia percentuale per segnalazione di warning e per quella critical, si ottiene:

```
nagios@linux:nagios/libexec$ ./check_nt -H winsrv -v CPULOAD -l 5,50,90
CPU Load 10% (5 min average) |   '5 min avg Load'=10%;50;90;0;100
```

Verifica dello stato di un servizio – eseguendo il comando

check_nt –H address –v SERVICESTATE –d SHOWALL –l service1,service2, …..

si può controllare lo stato di un servizio Windows; il parametro opzionale –d SHOWALL assicura che la lista in output contenga tutti i servizi; omettendolo si ottiene soltanto la lista di quelli che non sono nello stato di "running". In alternativa si può indicare il nome dei servizi da controllare, facendo attenzione a riferirsi a quello reale presente nella chiave di registro e non a quello di display visibile nella console di management. In presenza di NC_Net questa limitazione non esiste e si possono utilizzare indifferentemente il display name o quello riportato nella voce di registro. Ecco cosa si ottiene come riscontro a 2 diverse interrogazioni:

```
nagios@linux:nagios/libexec$ ./check_nt -H winsrv -v SERVICESTATE \
   -l "Routing and RAS"
Routing and RAS: Stopped

nagios@linux:nagios/libexec$ ./check_nt -H winsrv -v SERVICESTATE \
   -l "VNC Server"
All services are running
```

Il servizio “Routing and RAS” risulta non attivo ed il plugin check_nt restituirà un valore 2 (CRITICAL). Il fatto che “VNC Server” stia funzionando correttamente comporterà, oltre alla linea di testo, un valore 0 (OK) di ritorno. Più servizi possono essere controllati contemporaneamente da un unico comando, separando i nomi dei servizi con virgole; il valore di ritorno segnalerà il peggior risultato.

Esistono molti altri comandi da poter utilizzare, la cui complessità può crescere notevolmente in funzione del numero delle grandezze da controllare e del livello di correlazione fra esse.

Solo per completezza del quadro degli strumenti disponibili con Nagios per il monitoraggio di sistemi Windows, bisogna citare l’esistenza di un versione del Nagios Remote Plugin Executor (NRPE) “portata” in ambiente Windows e nota come NRPE_NT; il suo utilizzo è necessario quando devono essere eseguiti dei controlli localmente sui sistemi target che non siano supportati dai protocolli di rete disponibili in remoto. Come al solito i plugin devono essere installati in locale sul sistema target, così come pure il daemon NRPE_NT; i test da effettuare devono essere riportati in un file di configurazione sempre locale al sistema. Una serie di plugin dedicati a questa tipologia di esigenze sono stati sviluppati e resi disponibili attraverso le “user community” di Internet.

5. – Conclusioni

Le reti ed i sistemi di elaborazione distribuiti stanno diventando sempre più importanti ed, allo stesso tempo, sempre più critici per il mondo dell'Information Technology.

Si sta infatti assistendo, ad ogni livello nella scala delle organizzazioni, alla nascita di reti sempre più complesse, che supportano un numero sempre più maggiore di utenti, di applicazioni e servizi offerti.

Come conseguenza di ciò si hanno due possibili effetti che non è possibile tralasciare:

- La rete dell'organizzazione diventa indispensabile per il corretto funzionamento di tutti i processi che delineano la sua attività.
- Di pari passo alla crescita delle reti, si ha inoltre l'aumento degli aspetti critici di cui tener conto; come è ovvio, infatti, in una struttura complessa come può essere un sistema di rete, l'interazione tra i vari componenti fa si che un piccolo guasto possa comportare un blocco totale della rete o di parte di essa, oppure degradare la performance a livelli inaccettabili.

Da ciò si deduce che un'attività di amministrazione dei componenti della rete diviene sempre più indispensabile. Il concetto di amministrazione della rete è alquanto articolato. Esso implica attività quali l'individuazione e la gestione dei vari elementi della rete (host,gateway,router,proxy…), il monitoraggio delle sue prestazioni, l'inventario dell'hardware e del software presente in una rete e molto altro ancora.

Gli argomenti affrontati in questa tesi riguardano la specifica tematica del monitoraggio della rete, nei termini di infrastruttura tecnologica, nel più ampio quadro della gestione dell'infrastruttura. La criticità del tema è in relazione alla diretta corrispondenza fra questa attività e la cosiddetta "business continuity" dell'organizzazione.

Partendo dall'ottica del monitoraggio quale attività componente la gestione, e viste le correlazioni fra loro nel ciclo di controllo di un oggetto da monitorare (vedi Fig. 3 a pag. 9), la

tesi definisce il processo di monitoraggio come una sequenza di attività (generazione, elaborazione, disseminazione e presentazione delle informazioni).

La crescita esponenziale delle reti, sia dal punto di vista quantitativo (numero di host, switch, router e linee di interconnessione) sia dal punto di vista qualitativo (tipologia e topologia della rete, variabilità di tecnologie sottostanti) determinarono la necessità di un protocollo standard per la misurazione ed il controllo dei dispositivi collegati alla rete; si trattava cioè di concordare un linguaggio comune durante l'attività di monitoraggio di rete.

Le risposte della comunità tecnologica a queste esigenze sono state svariate, con differenti livelli di complessità e di obiettivi. Il protocollo che si è imposto, soprattutto per le sue caratteristiche di interoperabilità ed espandibilità, è stato il Simple Network Management Protocol (SNMP), divenuto standard "de facto" sul mercato. La descrizione fatta di SNMP è orientata ad una visione pratica, presentando le caratteristiche ed i principi di funzionamento, in relazione alle operazioni previste dal protocollo.

Relativamente alle applicazioni che sono disponibili sul mercato, si è optato per un giro di orizzonte fra quelle open-source, tralasciando quelle proprietarie, caratterizzate da elevati costi di acquisizione ed esercizio. In particolare ci si è soffermati sul software Nagios, che si è conquistato una leadership nel settore grazie all'architettura modulare e flessibile dei suoi "plugins"; proprio questa filosofia di lavoro ha creato un numero crescente di "user community" che attraverso Internet scambiano ed ampliano moduli del prodotto. In ambiente Nagios sono stati anche sviluppati moduli destinati al monitoraggio di sistemi Windows, consentendo di trattare in maniera integrata reti composte da sistemi eterogenei, centralizzando opportunamente l'attività di supervisione.

Bibliografia

[LH02] Thomas A. Limoncelli and Christine Hogan: *The Practice of System and Network Administration*. Addison-Wesley – Pearson Education, 2002. [p. 7, 16, 21]

[MSS94] Masoud Mansouri-Samani and Morris Sloman: *Monitoring Distributed Systems. In Morris Sloman, editor, Network and distributed systems management*, Addison-Wesley,Wokingham, UK. 1994., pp. 303–347. [p. 8, 9, 19]

[Hal00] Eric A. Hall: *Internet Core Protocols*. O'Reilly, Sebastopel, CA, 2000. [p. 11]

[MAS+03] James McGovern, Scott W. Ambler, Michael E. Stevens, James Linn, Vikas Sharan and Elias K. Jo: *A Practical Guide to Enterprise Architecture*. Prentice Hall, Upper Saddle River, NJ, 2003. [p. 21]

[MH06] Yusef Hassan Montero and Victor Herrero-Solana: *Improving Tag-Clouds as Visual Information Retrieval Interfaces.* Mérida, Spain, 2006.
URL http://www.nosolousabilidad.com/hassan/ improving_tagclouds.pdf [p. 21]

[Bal05] Tarus Balog: *Enterprise-Wide Network Management with OpenNMS*. O'Reilly SysAdmin, 2005.
URL http://www.oreillynet.com/pub/a/sysadmin/2005/09/08/opennms.html [p. 33]

[SKMC03] Federico D. Sacerdoti, Mason J. Katz, Matthew L. Massie and David E. Culler: *Wide Area Cluster Monitoring with Ganglia*. In: IEEE International Conference on Cluster Computing, Proceedings, 2003:pp. 289–298.
(http://ganglia.info/papers/Sacerdoti03Monitoring.pdf).

[Bar06] Wolfgang Barth: *Nagios - System And Network Monitoring*, 2006.
[p. 353-375]

RFCs

RFC 792 Internet Control Message Protocol (ICMP);

RFC 1157 A Simple Network Management Protocol (SNMP);

RFC 1441 Introduction to version 2 of the Internet-standard Network Management Framework;

RFC 3410 Introduction and Applicability Statements for Internet Standard Management Framework;

RFC 3411 Standard 62 - An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks;

RFC 3412 Standard 62 - Message Processing and Dispatching for the Simple Network Management Protocol (SNMP);

RFC 3413 Standard 62 - Simple Network Management Protocol (SNMP) Application;

RFC 3414 Standard 62 - User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3);

RFC 3415 Standard 62 - View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP);

RFC 3416 Standard 62 - Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP);

RFC 3417 Standard 62 - Transport Mappings for the Simple Network Management Protocol (SNMP);

RFC 3418 Standard 62 - Management Information Base (MIB) for the Simple Network Management Protocol (SNMP);

RFC 3584 Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework;

RFC 3512 Configuring Networks and Devices with Simple Network Management Protocol (SNMP);

RFC 1213 Management Information Base for Network Management of TCP/IP-based internets: MIB-II.