Progetto Sgr 2000/01

Definizione di un MIB SNMP(simile a RMON) per l'implementazione di un sistema di cattura pacchetti (sniffing) remoto da attivare da parte di uno o più manager quando si rilevano particolari problemi di sicurezza.

Irene Santerini
D.U. Informatica

Introduzione:

Il progetto richiede lo sviluppo di un MIB SNMP per l'implementazione di un sistema di cattura pacchetti (sniffing) remoto da attivare da parte di uno o più manager quando si rilevano particolari problemi di sicurezza. Come scelta di progetto ho supposto che i problemi di sicurezza da rilevare siano i seguenti cinque errori presenti in un possibile pacchetto:

- ➜ *Pacchetti bad*: sono pacchetti che hanno un giusto framing e sono quindi riconosciuti come pacchetti, ma contengono errori dentro il pacchetto o hanno una lunghezza non valida. Ad esempio su Ethernet pacchetti bad sono quelli che hanno una lunghezza o maggiore di 1518 octets o minore di 64 octects;
- ➜ *Pacchetti mal formattati*: sono pacchetti che non hanno un giusto framing;
- ➜ *Pacchetti troppo lunghi*: pacchetti che superano una tot lunghezza;
- ➜ *Invalid packets:* pacchetti TCP con flag strani/errati;
- ➜ *Pacchetti "portscan"*.

In seguito per i riferimenti a questi errori utilizzerò i seguenti alias:

- ⇒ Errore 1: per i pacchetti bad.
- ⇒ Errore 2: per i pacchetti mal formattati.
- ⇒ Errore 3: per i pacchetti troppo lunghi.
- ⇒ Errore 4: per i pacchetti TCP con flag strani/errati.
- ⇒ Errore 5: per i pacchetti "portscan".

Il caso di studio:

SNMP data la sua semplicità si presta per la realizzazione di software di monitoraggio di oggetti remoti tramite analisi di alcune variabili MIB. È possibile ad esempio monitorare i router per vedere il carico sulla loro CPU, per verificare l'efficienza dei settaggi della rete. Con alcuni programmi è addirittura possibile monitorare più router nello stesso istante per vedere come reindirizzare il traffico per ottenere prestazioni migliori.

In questo caso è possibile realizzare un sistema per la cattura di pacchetti da attivare da parte di uno o più manager, utilizzando dei contatori per calcolare il numero di pacchetti contenenti gli errori sopra detti. Dato che via SNMP si leggono solamente valori, si possono impostare dei valori di soglia (uno per ogni tipo di errore) oltre i quali venga generato un allarme software.

Il progetto utilizzerà uno o più network manager e uno o più agents distribuiti, strategicamente localizzati su ogni segmento di newtork.

Il manager esercita il controllo, avvia le operazioni di management tramite un management protocol. Questo compito è assegnato al manager per non appesantire l'agent, il quale deve essere semplice e veloce e deve eseguire esclusivamente operazioni atomiche, così da poter essere utilizzato da più manager che, magari, fanno cose diverse e usano protocolli eterogenei o di altro livello. Il manager riceve messaggi dall'agent e, eventualmente, li gira alle applicazioni o agli utenti, inoltre, il manager può filtrare i messaggi che riceve per individuare errori e malfunzionamenti, così da non inviare, alle applicazioni o agli amministratori una serie lunghissima di notifiche di errori dovute ad un unico problema. Nel progetto il manager avrà il compito di attivare o disattivare lo sniffing effettuato dai driver di cattura e di ricevere eventuali trap dovute al superamento del valore soglia del numero di pacchetti catturati contenenti un determinato errore.

L'agent esegue l'azioni sul MIB, per conto del manager, ed implementa quest'ultimo e le MIB accedendo alle risorse reali. Dopo aver ricevuto una richiesta, l'agent esegue l'operazione e invia una risposta al manager che gliel' ha commissionata, ma anche verso terze parti, se è previsto che queste vengano informate dell'accaduto. L'agent, infine, protegge i M.O. da accessi imprevisti o non autorizzati, causati da manomissioni dall'esterno, da malfunzionamenti o operazioni non consentite, anche in base al momento storico in cui si verificano. Nel progetto l'agent (un driver di sniffing) ha il compito di catturare pacchetti e di collezionare statistiche dal network in tempo reale e comunica con il manager tramite un esclusivo protocollo IP/UDP per fornire i dati di network, e con l'"utility" Trap via SNMP per provvedere allarmi al manager (dovuti al superamento di una determinata soglia).

Impostazione del progetto:

Il progetto si avvale di tre componenti principali:

Manager: è la parte centrale di un sistema distribuito per la cattura di pacchetti quando si rilevano particolari problemi di sicurezza.

Driver di cattura: la parte più importante: cattura il traffico di network dal "filo metallico", lo filtra per il particolare traffico voluto, e poi memorizza i dati in un buffer.

Buffer: una volta che i pacchetti sono catturati dal network, vengono memorizzati in un buffer.

Tabelle Utilizzate:

Il progetto si articola principalmente su due tabelle:

snifControlTable: una tabella dei pacchetti catturati da un particolare dispositivo di cattura (al quale è associato un canale). Questa tabella contiene una serie di parametri per il controllo dei pacchetti.

captureBufferTable: i pacchetti catturati sono posizionati in questa tabella. Queste entries sono associate con la "snifControlTable" sulla quale, per conto di essa, sono memorizzati.

Entries della tabella "*snifControlTable*":

SnifControlEntry ::= SEQUENCE {
indiceDiControllo
INTEGER (1...65535),
indiceCanale
INTEGER (1...65535),
statosePieno
INTEGER,
azionesePieno
INTEGER,
numeroPacchettiCatturati
Counter,
numeroPacchettiContenentiErrore_1
Counter,
numeroPacchettiContenentiErrore_2
Counter,
numeroPacchettiContenentiErrore_3
Counter,
numeroPacchettiContenentiErrore_4
Counter,
numeroPacchettiContenentiErrore_5
Counter,
maxNumPaccContenentiErrore_1
Unsigned32,
maxNumPaccContenentiErrore_2
Unsigned32,
maxNumPaccContenentiErrore_3
Unsigned32,

```
        maxNumPaccContenentiErrore_4
            Unsigned32,
        maxNumPaccContenentiErrore_5
            Unsigned32,
        proprietario
            Stringaproprietario,
        seSniffingAttivo
            INTEGER
    }
```

- **indiceDiControllo**: identifica una entry nella tabella, cioè un dispositivo di cattura pacchetti.
- **indiceCanale**: identifica un particolare canale da dove verranno catturati i pacchetti.
- **statosePieno**: indica se il driver ha spazio per accettare nuovi pacchetti o se è pieno. Se lo stato è "*spazioDisponibile (1)*", il driver accetta i nuovi pacchetti normalmente. Se lo stato è "*pieno (2)*" e l'associata variabile "*azionesePieno*" è settata a "*LockQuandoPieno (1)*", vengono cancellati i pacchetti più vecchi per far posto a quelli nuovi; altrimenti se la variabile "*azionesePieno*" è settata a "*wrapQuandoPieno (2)*" il driver termina di collezionare i pacchetti.
- **azionesePieno**: controlla l'azione del driver quando la variabile "*statosePieno*" raggiunge "*pieno (2)*" come descritto sopra.
- **numeroPacchettiCatturati**: indica il numero di pacchetti presenti nel buffer "*captureBufferTable*".
- **numeroPacchettiContenentiErrore_1**: indica il numero di pacchetti presenti nel buffer "*captureBufferTable*" contenenti l'errore 1.
- **numeroPacchettiContenentiErrore_2**: indica il numero di pacchetti presenti nel buffer "*captureBufferTable*" contenenti l'errore 2.
- **numeroPacchettiContenentiErrore_3**: indica il numero di pacchetti presenti nel buffer "*captureBufferTable*" contenenti l'errore 3.
- **numeroPacchettiContenentiErrore_4**: indica il numero di pacchetti presenti nel buffer "*captureBufferTable*" contenenti l'errore 4.
- **numeroPacchettiContenentiErrore_5**: indica il numero di pacchetti presenti nel buffer "*captureBufferTable*" contenenti l'errore 5.
- **maxNumPaccContenentiErrore_1**: indica il numero massimo di pacchetti che contenenti l'errore 1. Quando "*numeroPacchettiContenentiErrore_1*" supera questo valore di soglia l'agent invia una Trap al Manager.
- **maxNumPaccContenentiErrore_2**: indica il numero massimo di pacchetti che contenenti l'errore 2. Quando "*numeroPacchettiContenentiErrore_2*" supera questo valore di soglia l'agent invia una Trap al Manager.

- **MaxNumPaccContenentiErrore_3**: indica il numero massimo di pacchetti che contenenti l'errore 3. Quando "*numeroPacchettiContenentiErrore*_3" supera questo valore di soglia l'agent invia una Trap al Manager.
- **maxNumPaccContenentiErrore_4**: indica il numero massimo di pacchetti che contenenti l'errore 4. Quando "*numeroPacchettiContenentiErrore*_4" supera questo valore di soglia l'agent invia una Trap al Manager.
- **maxNumPaccContenentiErrore_5**: indica il numero massimo di pacchetti che contenenti l'errore 5. Quando "*numeroPacchettiContenentiErrore*_5" supera questo valore di soglia l'agent invia una Trap al Manager.
- **proprietario**: l'entità che ha configurato questa entry (cioè il manager a cui dovranno essere inviate eventuali trap). (Per il tipo di "*Stringaproprietario*" vedere le textual convention nell'apposito paragrafo sotto).
- **seSniffingAttivo**: indica se il dispositivo è abilitato/disabilitato per lo sniffing.

Entries della tabella "*captureBufferTable*":

```
CaptureBufferEntry ::= SEQUENCE {
                        indiceDiSnifControlTable
                              INTEGER (1...65535),
                        indiceDiControllo
                              INTEGER (1... 2147483647),
                        lunghezzaPacc
                              INTEGER,
                        tipoPacc
                              StringaTipo,
                        indirizzoMittente
                              StringaMittente,
                        statoPacc
                              INTEGER,
                  }
```

- **indiceDiSnifControlTable**: indica l'indice della snifControlEntry con il quale questo pacchetto è associato.
- **indiceDiControllo**: identifica una entry nella tabella, quindi un particolare pacchetto.
- **lunghezzaPacc**: indica la lunghezza del pacchetto memorizzato in una particolare entry.
- **tipoPacc**: indica il tipo del pacchetto (es. *SMTP*). (Per il tipo "*StringaTipo*" vedere le textual convention nell'apposito paragrafo sotto).

- ➜ **indirizzoMittente**: indica il mittente del pacchetto. (Per il tipo "*StringaMittente*" vedere le textual convention nell'apposito paragrafo sotto).
- ➜ **statoPacc**: indica lo stato di errore del pacchetto.

Textual Convention:

Le Textual Convention utilizzate sono le seguenti:

Stringaproprietario: questa stringa contiene l'entità che ha configurato una entry della tabella "*snifControlTabel*" . Questo nome contiene le seguenti informazioni: l'indirizzo IP dell'entità, la locazione dell'entità e il nome del network manager.

StringaTipo: questa stringa contiene il tipo del pacchetto (es. SMTP).

StringaMittente: questa stringa contiene l'indirizzo IP del mittente che ha spedito un particolare pacchetto.

TRAP:

Le trap non sono altro che una notifica eseguita dall'agent al manager quando il contatore "*numeroPacchettiContenentiErrore_**" supera la soglia "*maxNumPaccContenentiErrore_**". Pertanto le trap sono cinque, una per ogni tipo di errore.

Conclusioni:

Le eventuali migliorie da apportare al progetto dipendono soprattutto dall'utilizzo del MIB e, dato che lo sniffing dei pacchetti avviene quando il manager rileva particolari problemi di sicurezza, bisogna analizzare tali problemi, per poi apportare al progetto perfezionamenti e maggiori specifiche.

Variazioni al progetto potrebbero essere effettuate per:

- ⇒ ottenere statiche più precise: ad esempio controllando i pacchetti TCP con flag strani/errati è possibile analizzare da dove provengono e prendere provvedimenti nel caso in cui questi sono stati inviati principalmente da un particolare host.
- ⇒ analizzare i pacchetti non soltanto per trovare errori al suo interno (cioè nel suo formato). È possibile analizzare la provenienza per evitare accessi "non autorizzati" a risorse che una macchina particolare dovrebbe fornire. Per esempio, un web server dovrebbe fornire le pagine web a qualsiasi

utente che le richieda. Tuttavia, quel calcolatore centrale non dovrebbe fornire l'accesso alla shell di comando senza essere sicuro che la persona richiedente abbia l'autorizzazione, quale un amministratore locale.

⇒ ecc.

Molti sono i casi da analizzare: le limitazioni e le migliorie del progetto risiedono, quindi, nelle specifiche del progetto stesso. Esistono un ampia gamma di modifiche da eseguire in base alle necessità di chi utilizza tale MIB e l'assenza di tali specifiche porta sì ad una mancanza del progetto ma ha il vantaggio di una generalizzazione dello stesso per poter esser applicato a problema diversi ma analoghi.

Riferimenti:

⇒ "*Sistemi di Elaborazione dell'Informazione: Gestione di Rete*" di J. Shöenwälder – L.Deri
Lucidi del corso.
Anno Accademico 2000/ 2001

⇒ *"RFC 1757 - Remote Network Monitoring Management Information Base"*

```
SNIFFING-MIB DEFINITIONS ::= BEGIN

IMPORTS

        MODULE-IDENTITY, OBJECT-TYPE, mib-2, Counter, Integer, Unsigned32,
                FROM SNMPv2-SMI

        TEXTUAL-CONVENTION, DisplyString
                FROM SNMPv2-TC

        OBJECT GROUP
                FROM SNMPv2-CONF

        snmpTraps
                FROM SNMPv2-MIB



-- textual convention

StringaProprietario ::= DisplayString
StringaTipo  ::= DisplayString
StringaMittente ::= DisplayString


sniffingMIB MODULE-IDENTITY
     LASTUPDATE        "0109131807Z"
     ORGANIZATION      "Università di Pisa"
     CONCTACT-INFO     "Irene Santerini - E-mail: santeri@cli.di.unipi.it"
     DESCRIPTION       "Questo documento descrive un MIB SNMP per lo
                        sniffing da parte di uno o più manager quando si rilevano
                        particolari problemi di sicurezza."
  ::= {private 34}


sniffing OBJECT IDENTIFIER ::= { sniffingMIB 1 }

-- MONITORAGGIO

-- TABELLA snifControlTable

snifControlTable OBJECT-TYPE
  SYNTAX     SEQUENCE OF snifControlEntry
  ACCESS     not-accessible
  STATUS     mandatory
  DESCRIPTION "Tabella contenente più entries di controllo"
  ::= {sniffing 1}

snifControlEntry  OBJECT-TYPE
  SYNTAX    SnifControlEntry
```

ACCESS not-accessible
STATUS mandatory
DESCRIPTION "Una serie di parametri per il controllo dei pacchetti"
INDEX { agentControlIndex }
::= { snifControlTable 1}

```
SnifControlEntry ::= SEQUENCE {
                        indiceDiControllo
                            INTEGER (1...65535),
                        indiceCanale
                            INTEGER (1...65535),
                        statosePieno
                            INTEGER,
                        azionesePieno
                            INTEGER,
                        numeroPacchettiCatturati
                            Counter,
                        numeroPacchettiContenentiErrore_1
                            Counter,
                        numeroPacchettiContenentiErrore_2
                            Counter,
                        numeroPacchettiContenentiErrore_3
                            Counter,
                        numeroPacchettiContenentiErrore_4
                            Counter,
                        numeroPacchettiContenentiErrore_5
                            Counter,
                        maxNumPaccContenentiErrore_1
                            Unsigned32,
                        maxNumPaccContenentiErrore_2
                            Unsigned32,
                        maxNumPaccContenentiErrore_3
                            Unsigned32,
                        maxNumPaccContenentiErrore_4
                            Unsigned32,
                        maxNumPaccContenentiErrore_5
                            Unsigned32,
                        proprietario
                            Stringaproprietario,
                        seSniffingAttivo
                            INTEGER
                }
```

indiceDiControllo **OBJECT-TYPE**
SYNTAX INTEGER (1...65535)
ACCESS read-only
STATUS mandatory
DESCRIPTION "Indice che identifica una entry nella tabella"
::= { snifControlEntry 1 }

indiceCanale **OBJECT-TYPE**
SYNTAX INTEGER (1...65535)
ACCESS read-write
STATUS mandatory
DESCRIPTION "Indice che identifica il canale su cui passano i pacchetti da catturare"
::= { snifControlEntry 2 }

statosePieno **OBJECT-TYPE**
SYNTAX INTEGER {
spazioDisponibile(1),
pieno (2)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION "Questa opzione mostra se il dispositivo ha spazio per accettare nuovi pacchetti o se è pieno"
::= { snifControlEntry 3 }

azionesePieno **OBJECT-TYPE**
SYNTAX INTEGER {
LockQuandoPieno (1),
wrapQuandoPieno (2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION "Controlla l'azione del dispositivo quando raggiunge lo stato di pieno"
::= { snifControlEntry 4 }

numeroPacchettiCatturati **OBJECT-TYPE**
SYNTAX Counter
ACCESS read-only
STATUS current
DESCRIPTION "Numero di Pacchetti catturati"
::= { snifControlEntry 5}

numeroPacchettiContenentiErrore_1 **OBJECT-TYPE**
SYNTAX Counter
ACCESS read-only
STATUS current
DESCRIPTION "Il numero delle volte che un pacchetto contenente il 1° errore è stato ricevuto"
::= { snifControlEntry 6}

numeroPacchettiContenentiErrore_2 **OBJECT-TYPE**
SYNTAX Counter
ACCESS read-only
STATUS current
DESCRIPTION "Il numero delle volte che un pacchetto contenente il 2° errore è stato ricevuto"
::= { snifControlEntry 7}

numeroPacchettiContenentiErrore_3 **OBJECT-TYPE**
SYNTAX Counter
ACCESS read-only
STATUS current
DESCRIPTION "Il numero delle volte che un pacchetto contenente il 3° errore è stato ricevuto"
::= { snifControlEntry 8}

numeroPacchettiContenentiErrore_4 **OBJECT-TYPE**
SYNTAX Counter
ACCESS read-only
STATUS current
DESCRIPTION "Il numero delle volte che un pacchetto contenente il 4° errore è stato ricevuto"
::= { snifControlEntry 9}

numeroPacchettiContenentiErrore_5 **OBJECT-TYPE**
SYNTAX Counter
ACCESS read-only
STATUS current
DESCRIPTION "Il numero delle volte che un pacchetto contenente il 5° errore è stato ricevuto"
::= { snifControlEntry 10}

-- SOGLIE

maxNumPaccContenentiErrore_1 **OBJECT-TYPE**
SYNTAX Unsigned32
ACCESS read-write
STATUS current
DESCRIPTION "Soglia massima di pacchetti catturati contenenti il 1° errore "
::= { snifControlEntry 11}

maxNumPaccContenentiErrore_2 **OBJECT-TYPE**
SYNTAX Unsigned32
ACCESS read-write
STATUS current
DESCRIPTION "Soglia massima di pacchetti catturati contenenti il 2° errore "
::= { snifControlEntry 12}

maxNumPaccContenentiErrore_3 **OBJECT-TYPE**
SYNTAX Unsigned32
ACCESS read-write
STATUS current
DESCRIPTION "Soglia massima di pacchetti catturati contenenti il 3° errore "
::= { snifControlEntry 13}

maxNumPaccContenentiErrore_4 **OBJECT-TYPE**
SYNTAX Unsigned32
ACCESS read-write
STATUS current

```
  DESCRIPTION "Soglia massima di pacchetti catturati contenenti il 4° errore "
  ::= { snifControlEntry  14}

maxNumPaccContenentiErrore_5  OBJECT-TYPE
  SYNTAX      Unsigned32
  ACCESS      read-write
  STATUS      current
  DESCRIPTION "Soglia massima di pacchetti catturati contenenti il 5° errore "
  ::= { snifControlEntry  15}

proprietario  OBJECT-TYPE
  SYNTAX      StringaProprietario
  ACCESS      read-write
  STATUS      current
  DESCRIPTION "L'entità che ha configurato questa entry e che quindi usa le risorse
assegnate ad esso"
  ::= { snifControlEntry  16}

seSniffingAttivo OBJECT-TYPE
   SYNTAX INTEGER { disabilitato(1), abilitato(2) }
   ACCESS read-only
   STATUS mandatory
   DESCRIPTION "Se lo sniffing è abilitato o disabilitato"
   ::= { snifControlEntry  17 }

-- TABELLA captureBufferTable

captureBufferTable OBJECT-TYPE
  SYNTAX      SEQUENCE OF captureBufferEntry
  ACCESS      not-accessible
  STATUS      mandatory
  DESCRIPTION "Lista dei pacchetti catturati da un dispositivo"
  ::= { sniffing 2}

captureBufferEntry OBJECT-TYPE
  SYNTAX     CaptureBufferEntry
  ACCESS     not-accessible
  STATUS     mandatory
  DESCRIPTION "Un pacchetto catturato da un dispositivo"
  INDEX        { agentControlIndex }
  ::= { captureBufferTable 1}

CaptureBufferEntry ::= SEQUENCE {
                              indiceDiSnifControlTable
                                    INTEGER (1...65535),
                              indiceDiControllo
                                    INTEGER (1... 2147483647),
                              lunghezzaPacc
                                    INTEGER,
                              tipoPacc
                                    StringaTipo,
```

indirizzoMittente
StringaMittente,
statoPacc
INTEGER,
}

indiceDiSnifControlTable **OBJECT-TYPE**
SYNTAX INTEGER (1…65535)
ACCESS read-only
STATUS mandatory
DESCRIPTION "L'indice della snifControlEntry con il quale questo pacchetto è associato"
::= { captureBufferEntry 1 }

indiceDiControllo **OBJECT-TYPE**
SYNTAX INTEGER (1…2147483647)
ACCESS read-only
STATUS mandatory
DESCRIPTION " Indice che identifica una entry nella tabella associata con un particolare snifControlEntry "
::= { captureBufferEntry 2 }

lunghezzaPacc **OBJECT-TYPE**
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION "La lunghezza del pacchetto memorizzato in questa entry "
::= { captureBufferEntry 3 }

tipoPacc **OBJECT-TYPE**
SYNTAX StringaMittente
ACCESS read-only
STATUS mandatory
DESCRIPTION "Il tipo del pacchetto memorizzato in questa entry"
::= { captureBufferEntry 4 }

indirizzoMittente **OBJECT-TYPE**
SYNTAX StringaTipo
ACCESS read-only
STATUS mandatory
DESCRIPTION "L'indirizzo del mittente del pacchetto memorizzato in questa entry"
::= { captureBufferEntry 5 }

statoPacc **OBJECT-TYPE**
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION "Lo stato di errore del pacchetto memorizzato in questa entry"
::= { captureBufferEntry 6 }

-- **TRAP**

dispositivo **OBJECT IDENTIFIER** ::= { enterprises 2001 }

notificaMsgContenentiErrore_1 **TRAP-TYPE**
ENTERPRISE dispositivo
VARIABLES { indiceDiControllo }
DESCRIPTION "Trap inviata al manager quando il numero di pacchetti contenenti il 1°errore supera la soglia max prefissata"
::= { sniffing 3}

notificaMsgContenentiErrore_2 **TRAP-TYPE**
ENTERPRISE dispositivo
VARIABLES { indiceDiControllo }
DESCRIPTION "Trap inviata al manager quando il numero di pacchetti contenenti il 1°errore supera la soglia max prefissata"
::= { sniffing 4}

notificaMsgContenentiErrore_3 **TRAP-TYPE**
ENTERPRISE dispositivo
VARIABLES { indiceDiControllo }
DESCRIPTION "Trap inviata al manager quando il numero di pacchetti contenenti il 1°errore supera la soglia max prefissata"
::= { sniffing 5}

notificaMsgContenentiErrore_4 **TRAP-TYPE**
ENTERPRISE dispositivo
VARIABLES { indiceDiControllo }
DESCRIPTION "Trap inviata al manager quando il numero di pacchetti contenenti il 1°errore supera la soglia max prefissata"
::= { sniffing 6}

notificaMsgContenentiErrore_5 **TRAP-TYPE**
ENTERPRISE dispositivo
VARIABLES { indiceDiControllo }
DESCRIPTION "Trap inviata al manager quando il numero di pacchetti contenenti il 1°errore supera la soglia max prefissata"
::= { sniffing 7}