**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**

**-----□□&□□-----**

**BÁO CÁO ĐỒ ÁN**

**MÔN HỌC: ĐÁNH GIÁ HIỆU NĂNG HỆ THỐNG MẠNG MÁY TÍNH**

**Năm học: 2021 – 2022**



**Management SNMP With GNS3 And MIB Browser**

**Lớp: NT531.M21**

**Giảng viên hướng dẫn: Nguyễn Khánh Thuật**

**Nhóm 3:**

|  |  |
| --- | --- |
| Nguyễn Mỹ Báo  Bùi Duy Long | 19521250  19520692 |
| Lương Nguyễn Mai Ly | 19520707 |

**TP. Hồ Chí Minh, tháng 05 năm 2022**

**MỤC LỤC**

[LỜI MỞ ĐẦU 6](#_Toc104127923)

[1. Lí do chọn đề tài 6](#_Toc104127924)

[2. Mục tiêu đề tài 6](#_Toc104127925)

[3. Nội dung đề tài 6](#_Toc104127926)

[DANH MỤC HÌNH ẢNH, BẢNG BIỂU 8](#_Toc104127927)

[CHƯƠNG 1: GIAO THỨC QUẢN LÝ MẠNG SNMP 9](#_Toc104127928)

[1.1. SNMP là gì ? 9](#_Toc104127929)

[1.2. Các thành phần trong SNMP 9](#_Toc104127930)

[*1.2.1.* Kiến trúc của SNMP 9](#_Toc104127931)

[*1.2.2.* Object ID 11](#_Toc104127932)

[*1.2.3.* *Object Access* 12](#_Toc104127933)

[*1.2.4.* *Management Information Base* 13](#_Toc104127934)

[1.3. Các phiên bản và phương thức của SNMP 14](#_Toc104127935)

[1.3.1. SNMPv1 14](#_Toc104127936)

[1.3.1.1. GetRequest 15](#_Toc104127937)

[1.3.1.2. GetNextRequest 15](#_Toc104127938)

[1.3.1.3. SetRequest 15](#_Toc104127939)

[1.3.1.4. GetResponse 16](#_Toc104127940)

[1.3.1.5. Trap 16](#_Toc104127941)

[1.3.2. SNMPv2c 17](#_Toc104127942)

[1.3.3. SNMPv2u 18](#_Toc104127943)

[1.3.4. SNMPv3 18](#_Toc104127944)

[**1.4.** **Các cơ chế bảo mật cho SNMP** 19](#_Toc104127945)

[1.4.1. Community String 19](#_Toc104127946)

[*1.4.2.* *View* 20](#_Toc104127947)

[*1.4.3.* *SNMP access control list* 20](#_Toc104127948)

[**1.5.** **Cấu trúc bản tin của SNMP** 21](#_Toc104127949)

[CHƯƠNG 2: TỔNG QUAN VỀ PHẦN MỀM GNS3 22](#_Toc104127950)

[**2.1.** **Giới thiệu GNS3** 22](#_Toc104127951)

[**2.2.** **Cài đặt** 23](#_Toc104127952)

[*2.1.1.* *Cài đặt GNS3* 23](#_Toc104127953)

[*2.1.2.* *Thiết lập Dynamips và IOS* 25](#_Toc104127954)

[**2.3.** **Làm việc trên GNS3** 27](#_Toc104127955)

[*2.3.1.* *Các công cụ làm việc* 27](#_Toc104127956)

[*2.3.2.* *Thiết lập Router* 29](#_Toc104127957)

[*2.3.3.* *Tinh chỉnh các Interface* 30](#_Toc104127958)

[*2.3.4.* *Tinh chỉnh Idle PC* 32](#_Toc104127959)

[*2.3.5.* *Thiết lập kết nối* 32](#_Toc104127960)

[*2.3.6.* *Cấu hình Router thông qua giao diện console* 34](#_Toc104127961)

[*2.3.7.* *Giao tiếp với PC thật* 37](#_Toc104127962)

[Chương 3: Tổng quan về MIB Browser 40](#_Toc104127963)

[**3.1.** **Giới thiệu chung** 40](#_Toc104127964)

[**3.2.** **Tính năng chính** 40](#_Toc104127965)

[**3.3.** **Yêu cầu cài đặt** 41](#_Toc104127966)

[**3.4.** **Tải và chạy MIB Browser** 41](#_Toc104127967)

[**3.5.** **Giao diện** 42](#_Toc104127968)

[KẾT LUẬN 73](#_Toc104127969)

LỜI MỞ ĐẦU

1. Lí do chọn đề tài

Công Nghệ mạng Internet đang phát triển mạnh mẽ, và xu hướng tích hợp các mạng không đồng nhất để chia sẻ thông tin cũng xuất hiện ngày càng nhiều. Việc đảm bảo hệ thống mạng phức tạp, có quy mô lớn hoạt động một cách tin cậy và có hiệu năng cao đòi hỏi phải có hệ quản trị mạng để thu thập và phân tích một số lượng lớn dữ liệu một cách hiệu quả. Tuy nhiên, thông tin quản trị mang lại phải truyền trên môi trường Internet, có nguy cơ bị thất thoát, thay đổi, giả mạo, vì vậy thông tin cần phải được bảo vệ.

Việc nghiên cứu các giải pháp đảm bảo tính xác tực, tính bảo mật của các thông điệp quản trị mạng là hết sức cần thiết. Giao thức SNMPv3 đã ra đời nhằm đáp ứng một phần yêu cầu cấp bách này. Tuy nhiên, việc lựa chọn mô hình thực thi vẫn còn nhiều vấn đề cần giải quyết. Phần mềm quản lý mạng GNS3 Network Monitor theo dõi và ngưng hoạt động hệ thống cho phép người dùng ngay lập tức phát hiện lỗi hoặc chậm hệ thống. Người dùng có thể tạo các báo cáo cho các giá trị và thời gian chết, kiểm tra băng thông, hiệu suất sử dụng, uptime và khả năng sẵn có, các vấn đề báo cáo và cảnh báo, giao diện web.

Vậy nên mục tiêu của đề tài “Tìm hiểu về phần mềm giám sát mạng GNS3 Network Monitor” là giúp chúng to hiểu rõ hơn về giao thức SNMP, trên cơ sở đó nghiên cứu về phần mềm sử dụng SNMP là GNS3 và MIB Browser.

1. Mục tiêu đề tài

Mục tiêu nghiên cứu đề tài của nhóm bao gồm các điểm sau:

* Tìm hiểu giao thức quản trị mạng SNMP.
* Tổng quan về phần mềm GNS3, MIB Browser.
* Cài đặt và sử dụng phần mềm GNS3, MIB Browser.

1. Nội dung đề tài

Đề hoàn thành các mục tiêu trên, nhóm tập trung nghiên cứu các nội dung sau:

* Nghiên cứu cơ sở lý thuyết về giao thức SNMP.
* Tìm hiểu phần mềm giám sát và quản lý mạng SNMP.
* Tiến hành mô phỏng và cài đặt phần mềm GNS3, MIB Browser.

DANH MỤC HÌNH ẢNH

**Danh mục hình ảnh**

[Hình 1. Kiến trúc của SNMP 12](#_Toc104128009)

[Hình 2. Hoạt động của SNMP agent 13](#_Toc104128010)

[Hình 3. Quá trình lấy sysName 14](#_Toc104128011)

[Hình 4. MIB tree 15](#_Toc104128012)

[Hình 5. Các phương thức của SNMPv1 19](#_Toc104128013)

[Hình 6. Các phương thức của SNMPv2c 20](#_Toc104128014)

[Hình 7. Cấu trúc bản tin SNMP 23](#_Toc104128015)

[Hình 8. Giao diện chính GNS3 26](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128016)

[Hình 9.Thiết lập Dynamips 27](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128017)

[Hình 10. Cấu hình IOS 29](#_Toc104128018)

[Hình 11. Thanh menu 29](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128019)

[Hình 12. Thanh công cụ 29](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128020)

[Hình 13. Cửa sổ thiết bị 30](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128021)

[Hình 14. Cửa sổ Topology 30](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128022)

[Hình 15. Cửa sổ Capture 30](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128023)

[Hình 16. Cửa sổ Console 31](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128024)

[Hình 17. Thiết lập Router 32](#_Toc104128025)

[Hình 18. Tinh chỉnh Interface 33](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128026)

[Hình 19. Thiết lập kết nối 34](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128027)

[Hình 20. Cấu hình R1 36](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128028)

[Hình 21. Cấu hình R2 37](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128029)

[Hình 22. Kiểm tra kết nối 38](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128030)

[Hình 23. Cấu hình Cloud 39](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128031)

[Hình 24. Chọn card VMWare 39](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128032)

[Hình 25. Thiết lập kết nối R1-C1 40](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128033)

[Hình 26. Cấu hình IP f0/1 của R1 40](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128034)

[Hình 27. Cấu hình IP VMNet1 41](file:///C:\Users\longb\Desktop\baocaocuoiky_NT531-NHOM3.docx#_Toc104128035)

[Hình 28. Giao diện chính của MIB Browser 44](#_Toc104128036)

[Hình 29. Hộp thoại Set 46](#_Toc104128037)

[Hình 30. Get Subtree 47](#_Toc104128038)

[Hình 31. Result Table 48](#_Toc104128039)

[Hình 32. Cửa sổ TrapSender 49](#_Toc104128040)

[Hình 33. Chọn đối tượng trong MIB Tree 50](#_Toc104128041)

[Hình 34. Add to Watches 51](#_Toc104128042)

[Hình 35. Configure Actions 51](#_Toc104128043)

[Hình 36. Cài đặt MIB Watches 52](#_Toc104128044)

[Hình 37. Port View 52](#_Toc104128045)

[Hình 38. Switch Port Mapper 53](#_Toc104128046)

[Hình 39. Device Snapshot 54](#_Toc104128047)

[Hình 40. Cisco Device Snapshot 55](#_Toc104128048)

[Hình 41. Ping 56](#_Toc104128049)

[Hình 42. Trace Route 56](#_Toc104128050)

[Hình 43. Network Discovery 57](#_Toc104128051)

[Hình 44. Manage SNMPv3 USM Users 57](#_Toc104128052)

[Hình 45. Compare Devices 58](#_Toc104128053)

[Hình 46. Options 59](#_Toc104128054)

[Hình 47. General Tabs 61](#_Toc104128055)

[Hình 48. Agent Tab 61](#_Toc104128056)

[Hình 49. MIB Files Tab 62](#_Toc104128057)

[Hình 50. Toolbar 62](#_Toc104128058)

[Hình 51. MIB Tree Pane 63](#_Toc104128059)

[Hình 52. Tree Pane 64](#_Toc104128060)

[Hình 53. Result Pane 65](#_Toc104128061)

[Hình 54. Trap Receiver 66](#_Toc104128062)

[Hình 55. Add Trap Rule 67](#_Toc104128063)

[Hình 56. Trap Receiver General Setting 68](#_Toc104128064)

[Hình 57. Trap Receiver SMTP Setting 70](#_Toc104128065)

[Hình 58. Trap Receiver Email Template Setting 71](#_Toc104128066)

[Hình 59. Trap Receiver SNMPv3 Setting 72](#_Toc104128067)

CHƯƠNG 1: GIAO THỨC QUẢN LÝ MẠNG SNMP

1. SNMP là gì ?

SNMP (Simple Network Management Protocol) là một giao thức tầng ứng dụng được Hội đồng Kiến trúc Internet (IAB) xác định trong RFC1157 để trao đổi thông tin quản lý giữa các thiết bị mạng. Nó là một phần của Transmission Control Protocol/Internet Protocol (TCP/IP).

Giao thức SNMP là một trong những giao thức mạng được chấp nhận rộng rãi để quản lý và giám sát các phần tử mạng. Hầu hết các thiết bị mạng được cung cấp đi kèm với SNMP agent. Các agent này phải được kích hoạt và cấu hình để giao tiếp với các công cụ giám sát mạng hoặc hệ thống quản lý mạng (NMS).

SNMP dùng để quản lý, nghĩa là có thể theo dõi, có thể lấy thông tin, có thể được thông báo, và có thể tác động để hệ thống hoạt động như ý muốn. VD một số khả năng của phần mềm SNMP :

* Theo dõi tốc độ đường truyền của một router, biết được tổng số byte đã truyền/nhận.
* Lấy thông tin máy chủ đang có bao nhiêu ổ cứng, mỗi ổ cứng còn trống bao nhiêu.
* Tự động nhận cảnh báo khi switch có một port bị down
* Điều khiển tắt (shutdown) các port trên switch.

SNMP dùng để quản lý mạng, nghĩa là nó được thiết kế để chạy trên nền TCP/IP và quản lý các thiết bị có nối mạng TCP/IP. Các thiết bị mạng không nhất thiết phải là máy tính mà có thể là switch, router, firewall, adsl gateway, và cả một số phần mềm cho phép quản trị bằng SNMP. Giả sử bạn có một cái máy giặt có thể nối mạng IP và nó hỗ trợ SNMP thì bạn có thể quản lý nó từ xa bằng SNMP. SNMP là giao thức đơn giản, do nó được thiết kế đơn giản trong cấu trúc bản tin và thủ tục hoạt động, và còn đơn giản trong bảo mật (ngoại trừ SNMP version 3). Sử dụng phần mềm SNMP, người quản trị mạng có thể quản lý, giám sát tập trung từ xa toàn mạng của mình.

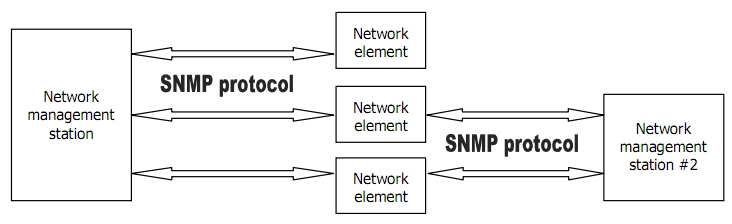
1. Các thành phần trong SNMP
2. Kiến trúc của SNMP

Theo **RFC1157** , kiến trúc của SNMP bao gồm 2 thành phần : các trạm quản lý mạng (network management station) và các thành tố mạng (network element).

Network  management  station  thường  là  một  máy  tính  chạy  phần  mềm  quản  lý  SNMP  (SNMP management application), dùng để giám sát và điều khiển tập trung các network element.

**Network element** là các thiết bị, máy tính, hoặc phần mềm tương thích SNMP và được quản lý bởi network management station. Như vậy element bao gồm device, host và application.

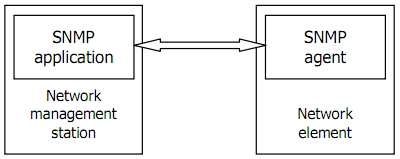
Một management station có thể quản lý nhiều element, một element cũng có thể được quản lý bởi nhiều management station. Vậy nếu một element được quản lý bởi 2 station thì điều gì sẽ xảy ra ? Nếu station lấy thông tin từ element thì cả 2 station sẽ có thông tin giống nhau. Nếu 2 station tác động đến cùng một element thì element sẽ đáp ứng cả 2 tác động theo thứ tự cái nào đến trước.



Hình 1. Kiến trúc của SNMP

Ngoài ra còn có khái niệm SNMP agent. SNMP agent là một tiến trình (process) chạy trên network element, có nhiệm vụ cung cấp thông tin của element cho station, nhờ đó station có thể quản lý được element. Chính xác hơn là application chạy trên station và agent chạy trên element mới là 2 tiến trình SNMP trực tiếp liên hệ với nhau. Các ví dụ minh họa sau đây sẽ làm rõ hơn các khái niệm này :

* Để dùng một máy chủ (= station) quản lý các máy con (= element) chạy HĐH Windows thông qua SNMP thì bạn phải : cài đặt một phần mềm quản lý SNMP (=application) trên máy chủ, bật SNMP service (= agent) trên máy con.
* Để dùng một máy chủ (= station) giám sát lưu lượng của một router (= element) thì bạn phải : cài phần mềm quản lý SNMP (= application) trên máy chủ, bật tính năng SNMP (=agent) trên router.



Hình 2. Hoạt động của SNMP agent

1. Object ID

Một thiết bị hỗ trợ SNMP có thể cung cấp nhiều thông tin khác nhau, mỗi thông tin đó gọi là một object. Ví dụ:

* Máy tính có thể cung cấp các thông tin : tổng số ổ cứng, tổng số port nối mạng, tổng số byte đã truyền/nhận, tên máy tính, tên các process đang chạy.
* Router có thể cung cấp các thông tin : tổng số card, tổng số port, tổng số byte đã truyền/nhận, tên router, tình trạng các port của router, ….

Mỗi object có một tên gọi và một mã số để nhận dạng object đó, mã số gọi là Object ID (OID). Ví dụ:

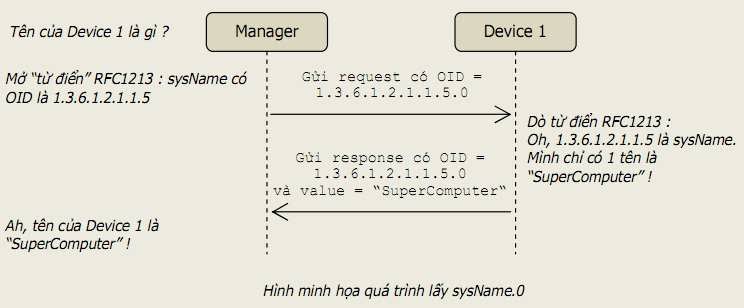
* Tên thiết bị được gọi là sysName, OID là 1.3.6.1.2.1.1.5
* Tổng số port giao tiếp (interface) được gọi là ifNumber, OID là 1.3.6.1.2.1.2.1.
* Địa chỉ Mac Address của một port được gọi là ifPhysAddress, OID là 1.3.6.1.2.1.2.2.1.6.
* Số byte đã nhận trên một port được gọi là ifInOctets, OID là 1.3.6.1.2.1.2.2.1.10.

Tuy nhiên nếu một thiết bị lại có nhiều tên thì làm thế nào để phân biệt ? Lúc này người ta dùng thêm 1 chỉ số gọi là “scalar instance index” (cũng có thể gọi là “sub-id”) đặt ngay sau OID. Ví dụ :

* Tên thiết bị được gọi là sysName, OID là 1.3.6.1.2.1.1.5; nếu thiết bị có 2 tên thì chúng sẽ được gọi là sysName.0 & sysName.1 và có OID lần lượt là 1.3.6.1.2.1.1.5.0 & 1.3.6.1.2.1.1.5.1.
* Địa chỉ Mac address được gọi là ifPhysAddress, OID là 1.3.6.1.2.1.2.2.1.6; nếu thiết bị có 2 mac address thì chúng sẽ được gọi là ifPhysAddress.0 & ifPhysAddress.1 và có OID lần lượt là 1.3.6.1.2.1.2.2.1.6.0 & 1.3.6.1.2.1.2.2.1.6.1.
* Tổng số port được gọi là ifNumber, giá trị này chỉ có 1 (duy nhất) nên OID của nó không có phân cấp con và vẫn là 1.3.6.1.2.1.2.1.

OID của các object phổ biến có thể được chuẩn hóa, OID của các object do bạn tạo ra thì bạn phải tự mô tả chúng. Để lấy một thông tin có OID đã chuẩn hóa thì SNMP application phải gửi một bản tin SNMP có chứa OID của object đó cho SNMP agent, SNMP agent khi nhận được thì nó phải trả lời bằng thông tin ứng với OID đó.

**Ví dụ:** Muốn lấy tên của một PC chạy Windows, tên của một PC chạy Linux hoặc tên của một router thì SNMP application chỉ cần gửi bản tin có chứa OID là 1.3.6.1.2.1.1.5.0. Khi SNMP agent chạy trên PC Windows, PC Linux hay router nhận được bản tin có chứa OID 1.3.6.1.2.1.1.5.0, agent lập tức hiểu rằng đây là bản tin hỏi sysName.0, và agent sẽ trả lời bằng tên của hệ thống. Nếu SNMP agent nhận được một OID mà nó không hiểu (không hỗ trợ) thì nó sẽ không trả lời.



Hình 3. Quá trình lấy sysName

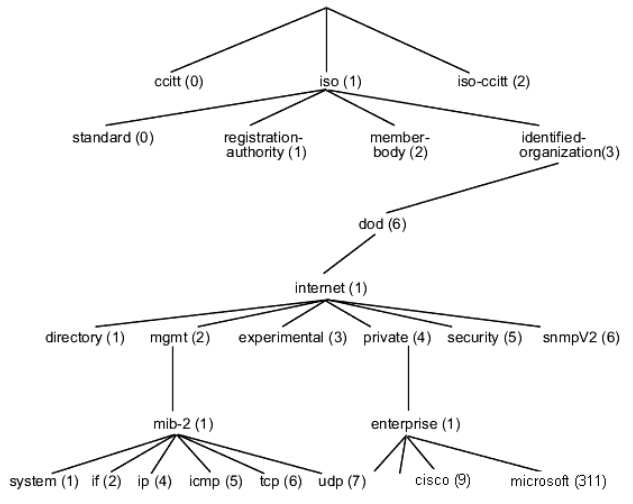
Một trong các ưu điểm của SNMP là nó được thiết kế để chạy độc lập với các thiết bị khác nhau. Chính nhờ việc chuẩn hóa OID mà ta có thể dùng một SNMP application để lấy thông tin các loại device của các hãng khác nhau.

1. *Object Access*

Mỗi object có quyền truy cập là READ\_ONLY hoặc READ\_WRITE. Mọi object đều có thể đọc được nhưng chỉ những object có quyền READ\_WRITE mới có thể thay đổi được giá trị. VD : Tên của một thiết bị (sysName) là READ\_WRITE, ta có thể thay đổi tên của thiết bị thông qua giao thức SNMP. Tổng số port của thiết bị (ifNumber) là READ\_ONLY, dĩ nhiên ta không thể thay đổi số port của nó.

1. *Management Information Base*

MIB (cơ sở thông tin quản lý) là một cấu trúc dữ liệu gồm các đối tượng được quản lý (managed object), được dùng cho việc quản lý các thiết bị chạy trên nền TCP/IP. MIB là kiến trúc chung mà các giao thức quản lý trên TCP/IP nên tuân theo, trong đó có SNMP. MIB được thể hiện thành 1 file (MIB file), và có thể biểu diễn thành 1 cây (MIB tree). MIB có thể được chuẩn hóa hoặc tự tạo.



Hình 4. MIB tree

Một node trong cây là một object, có thể được gọi bằng tên hoặc id. Ví dụ;

* Node iso.org.dod.internet.mgmt.mib-2.system có OID là 1.3.6.1.2.1.1, chứa tất cả các object liên quan đến thông tin của một hệ thống như tên của thiết bị (iso.org.dod.internet.mgmt.mib-2.system.sysName hay 1.3.6.1.2.1.1.5).
* Các OID của các hãng tự thiết kế nằm dưới iso.org.dod.internet.private.enterprise. Ví dụ : Cisco nằm dưới iso.org.dod.internet.private.enterprise.cisco hay 1.3.6.1.4.1.9, Microsoft nằm dưới iso.org.dod.internet.private.enterprise.microsoft hay 1.3.6.1.4.1.311. Số 9 (Cisco) hay 311 (Microsoft) là số dành riêng cho các công ty do IANA cấp. Nếu Cisco hay Microsoft chế tạo ra một thiết bị nào đó, thì thiết bị này có thể hỗ trợ các MIB chuẩn đã được định nghĩa sẵn (như mib-2) hay hỗ trợ MIB được thiết kế riêng. Các MIB được công ty nào thiết kế riêng thì phải nằm bên dưới OID của công ty đó.

1. Các phiên bản và phương thức của SNMP

SNMPv1 có 5 phương thức, tuy nhiên các version khác sau này được bổ sung thêm một số phương thức mới và sẽ khác nhau ở cấu trúc các bản tin.

1. SNMPv1

Giao thức SNMPv1 có 5 phương thức hoạt động, tương ứng với 5 loại bản tin như sau: :

|  |  |
| --- | --- |
| **Bản tin/phương thức** | **Mô tả tác dụng** |
| GetRequest | Manager gửi GetRequest cho agent để yêu cầu agent cung cấp thông tin nào đó dựa vào ObjectID (trong GetRequest có chứa OID) |
| GetNextRequest | Manager gửi GetNextRequest có chứa một ObjectID cho agent để yêu cầu cung cấp thông tin nằm kế tiếp ObjectID đó trong MIB |
| SetRequest | Manager gửi SetRequest cho agent để đặt giá trị cho đối tượng của agent dựa vào ObjectID. |
| GetResponse | Agent gửi GetResponse cho Manager để trả lời khi nhận được GetRequest/GetNextRequest |
| Trap | Agent tự động gửi Trap cho Manager khi có một sự kiện xảy ra đối với một object nào đó trong agent. |

Bảng 1. Tác dụng của các phương thức SNMPv1

Mỗi bản tin đều có chứa OID để cho biết object mang trong nó là gì. OID trong GetRequest cho biết nó muốn lấy thông tin của object nào. OID trong GetResponse cho biết nó mang giá trị của object nào. OID trong SetRequest chỉ ra nó muốn thiết lập giá trị cho object nào. OID trong Trap chỉ ra nó thông báo sự kiện xảy ra đối với object nào.

1. GetRequest

Bản tin GetRequest được manager gửi đến agent để lấy một thông tin nào đó. Trong GetRequest có chứa OID của object muốn lấy. Ví dụ: Muốn lấy thông tin tên của Device1 thì manager gửi bản tin GetRequest OID=1.3.6.1.2.1.1.5 đến Device1, tiến trình SNMP agent trên Device1 sẽ nhận được bản tin và tạo bản tin trả lời.

1. GetNextRequest

Bản tin GetNextRequest cũng dùng để lấy thông tin và cũng có chứa OID, tuy nhiên nó dùng để lấy thông tin của object nằm kế tiếp object được chỉ ra trong bản tin. Một MIB bao gồm nhiều OID được sắp xếp thứ tự nhưng không liên tục, nếu biết một OID thì không xác định được OID kế tiếp. Do đó ta cần GetNextRequest để lấy về giá trị của OID kế tiếp. Nếu thực hiện GetNextRequest liên tục thì ta sẽ lấy được toàn bộ thông tin của agent.

1. SetRequest

Bản tin SetRequest được manager gửi cho agent để thiết lập giá trị cho một object nào đó. Ví dụ:

* Có thể đặt lại tên của một máy tính hay router bằng phần mềm SNMP manager, bằng cách gửi bản tin SetRequest có OID là 1.3.6.1.2.1.1.5.0 (sysName.0) và có giá trị là tên mới cần đặt.
* Có thể shutdown một port trên switch bằng phần mềm SNMP manager, bằng cách gửi bản tin có
* Chỉ những object có quyền READ\_WRITE mới có thể thay đổi được giá trị.

1. GetResponse

Mỗi khi SNMP agent nhận được các bản tin GetRequest, GetNextRequest hay SetRequest thì nó sẽ gửi lại bản tin GetResponse để trả lời. Trong bản tin GetResponse có chứa OID của object được request và giá trị của object đó.

1. Trap

Bản tin Trap được agent tự động gửi cho manager mỗi khi có sự kiện xảy ra bên trong agent, các sự kiện này không phải là các hoạt động thường xuyên của agent mà là các sự kiện mang tính biến cố. Ví dụ : Khi có một port down, khi có một người dùng login không thành công, hoặc khi thiết bị khởi động lại, agent sẽ gửi trap cho manager.

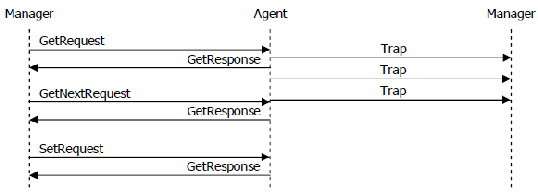
Tuy nhiên không phải mọi biến cố đều được agent gửi trap, cũng không phải mọi agent đều gửi trap khi xảy ra cùng một biến cố. Việc agent gửi hay không gửi trap cho biến cố nào là do hãng sản xuất device/agent quy định.

Phương thức trap là độc lập với các phương thức request/response. SNMP request/response dùng để quản lý còn SNMP trap dùng để cảnh báo. Nguồn gửi trap gọi là Trap Sender và nơi nhận trap gọi là Trap Receiver. Một trap sender có thể được cấu hình để gửi trap đến nhiều trap receiver cùng lúc.

Có 2 loại trap : trap phổ biến (generic trap) và trap đặc thù (specific trap). Generic trap được quy định trong các chuẩn SNMP, còn specific trap do người dùng tự định nghĩa (người dùng ở đây là hãng sản xuất SNMP device). Loại trap là một số nguyên chứa trong bản tin trap, dựa vào đó mà phía nhận trap biết bản tin trap có nghĩa gì.

Theo SNMPv1, generic trap có 7 loại sau : coldStart(0), warmStart(1), linkDown(2), linkUp(3), authenticationFailure(4), egpNeighborloss(5), enterpriseSpecific(6). Giá trị trong ngoặc là mã số của các loại trap. Ý nghĩa của các bản tin generic-trap như sau :

* **coldStart** : thông báo rằng thiết bị gửi bản tin này đang khởi động lại (reinitialize) và cấu hình của nó có thể bị thay đổi sau khi khởi động.
* **warmStart** : thông báo rằng thiết bị gửi bản tin này đang khởi động lại và giữ nguyên cấu hình cũ.
* **linkDown** : thông báo rằng thiết bị gửi bản tin này phát hiện được một trong những kết nối truyền thông (communication link) của nó gặp lỗi. Trong bản tin trap có tham số chỉ ra ifIndex của kết nối bị lỗi.
* **linkUp** : thông báo rằng thiết bị gửi bản tin này phát hiện được một trong những kết nối truyền thông của nó đã khôi phục trở lại. Trong bản tin trap có tham số chỉ ra ifIndex của kết nối được khôi phục.
* **authenticationFailure** : thông báo rằng thiết bị gửi bản tin này đã nhận được một bản tin không được chứng thực thành công (bản tin bị chứng thực không thành công có thể thuộc nhiều giao thức khác nhau như telnet, ssh, snmp, ftp, …). Thông thường trap loại này xảy ra là do user đăng nhập không thành công vào thiết bị.
* **egpNeighborloss** : thông báo rằng một trong số những “EGP neighbor” của thiết bị gửi trap đã bị coi là down và quan hệ đối tác (peer relationship) giữa 2 bên không còn được duy trì.
* **enterpriseSpecific** : thông báo rằng bản tin trap này không thuộc các kiểu generic như trên mà nó là một loại bản tin do người dùng tự định nghĩa.



Hình 5. Các phương thức của SNMPv1

Đối với các phương thức Get/Set/Response thì SNMP Agent lắng nghe ở port UDP 161, còn phương thức trap thì SNMP Trap Receiver lắng nghe ở port UDP 162.

1. SNMPv2c

SNMP version 2 chia thành 2 phiên bản khác nhau ở cơ chế bảo mật, trong đó phiên bản vẫn sử dụng cơ chế bảo mật dựa vào community string như ở SNMPv1 gọi là Community-based SNMPv2 hay SNMPv2c. Một số tài liệu đã ghi chú không đúng rằng “SNMPv2c bổ sung thêm cơ chế community string so với SNMPv1”, thực sự SNMPv2c và SNMPv1 đều có cơ chế xác thực đơn giản bằng community giống nhau.

Khác biệt của SNMPv2c so với SNMPv1 là :

* Có nhiều phương thức hơn so với SNMPv1.
* Cấu trúc bản tin Trap PDU khác so với SNMPv1.
* Có thêm bản tin Bulk PDU với cấu trúc riêng.

SNMPv2c có 8 phương thức gồm : GetRequest, GetNextRequest, Response, SetRequest, GetBulkRequest, InformRequest, Trap và Report. Như vậy so với SNMPv1 thì v2c có thêm các phương thức GetBulk, Inform và Report.

Graphical user interface, table

Description automatically generated

Hình 6. Các phương thức của SNMPv2c

1. SNMPv2u

Đây là phiên bản SNMPv2 sử dụng cơ chế bảo mật có chứng thực bằng băm1 và mã hóa đối xứng2 dữ liệu, gọi là User-based SNMPv2 hay SNMPv2u. Sau này phiên bản SNMPv3 ra đời đã thay thế hoàn toàn SNMPv2u và người ta không còn ưu tiên dùng SNMPv2u nữa. Do đó SNMPv2u sẽ không được trình bày trong tài liệu này mà SNMPv3 sẽ được trình bày chi tiết. Trong thực tế rất khó tìm thấy một thiết bị còn hỗ trợ SNMPv2u.

1. SNMPv3

Phiên bản bảo mật nhất của SNMP sử dụng mô hình bảo mật dựa trên người dùng (User-based security model) với các cơ chế chứng thực bằng băm (MD5, SHA) và mã hóa (DES, AES) hiện đại. Việc lập trình ứng dụng hỗ trợ được SNMPv3 phức tạp hơn, do đó hầu hết các phần mềm SNMP manager phiên bản có hỗ trợ SNMPv3 đều có tính phí, trong khi phiên bản miễn phí chỉ hỗ trợ SNMPv1 và SNMPv2.

1. **Các cơ chế bảo mật cho SNMP**

SNMP management station có thể quản lý/giám sát nhiều SNMP element, thông qua hoạt động gửi request và nhận trap. Tuy nhiên một SNMP element có thể được cấu hình để chỉ cho phép các SNMP management station nào đó được phép quản lý/giám sát mình. Các cơ chế bảo mật đơn giản này gồm có : community string, view và SNMP access control list.

1. *Community String*

Community string là một chuỗi ký tự được cài đặt giống nhau trên cả SNMP manager và SNMP agent, đóng vai trò như “mật khẩu” giữa 2 bên khi trao đổi dữ liệu. Community string có 3 loại : Read-community, Write-Community và Trap-Community.

Khi  manager  gửi GetRequest,  GetNextRequest đến agent  thì  trong bản tin gửi  đi  có  chứa  Read- Community. Khi agent nhận được bản tin request thì nó sẽ so sánh Read-community do manager gửi và Read-community mà nó được cài đặt. Nếu 2 chuỗi này giống nhau, agent sẽ trả lời; nếu 2 chuỗi này khác nhau, agent sẽ không trả lời.

Write-Community được dùng trong bản tin SetRequest. Agent chỉ chấp nhận thay đổi dữ liệu khi write- community 2 bên giống nhau.

Trap-community nằm trong bản tin trap của trap sender gửi cho trap receiver. Trap receiver chỉ nhận và lưu trữ bản tin trap chỉ khi trap-community 2 bên giống nhau, tuy nhiên cũng có nhiều trap receiver được cấu hình nhận tất cả bản tin trap mà không quan tâm đến trap-community.

Community string có 3 loại như trên nhưng cùng một loại có thể có nhiều string khác nhau. Nghĩa là một agent có thể khai báo nhiều read-community, nhiều write-community.

Trên hầu hết hệ thống, read-community mặc định là “public”, write-community mặc định là “private” và trap-community mặc định là “public”.

Community string chỉ là chuỗi ký tự dạng cleartext, do đó hoàn toàn có thể bị nghe lén khi truyền trên mạng. Hơn nữa, các community mặc định thường là “public” và “private” nên nếu người quản trị không thay đổi thì chúng có thể dễ dàng bị dò ra. Khi community string trong mạng bị lộ, một người dùng bình thường tại một máy tính nào đó trong mạng có thể quản lý/giám sát toàn bộ các device có cùng community mà không được sự cho phép của người quản trị.

1. *View*

Khi manager có read-community thì nó có thể đọc toàn bộ OID của agent. Tuy nhiên agent có thể quy định chỉ cho phép đọc một số OID có liên quan nhau, tức là chỉ đọc được một phần của MIB. Tập con của MIB này gọi là view, trên agent có thể định nghĩa nhiều view. Ví dụ : agent có thể định nghĩa view interfaceView bao gồm các OID liên quan đến interface, storageView bao gồm các OID liên quan đến lưu trữ, hay AllView bao gồm tất cả các OID.

Một view phải gắn liền với một community string. Tùy vào community string nhận được là gì mà agent xử lý trên view tương ứng. Ví dụ : agent định nghĩa read-community “inf” trên view interfaceView, và “sto” trên storageView; khi manager gửi request lấy OID ifNumber với community là “inf” thì sẽ được đáp ứng do ifNumber nằm trong interfaceView; nếu manager request OID hrStorageSize với community “inf” thì agent sẽ  không  trả  lời  do  hrStorageSize  không  nằm  trong  interfaceView;  nhưng  nếu  manager  request hrStorageSize với community “sto” thì sẽ được trả lời do hrStorageSize nằm trong storageView.

Việc định nghĩa các view như thế nào tùy thuộc vào từng SNMP agent khác nhau. Có nhiều hệ thống không hỗ trợ tính năng view.

1. *SNMP access control list*

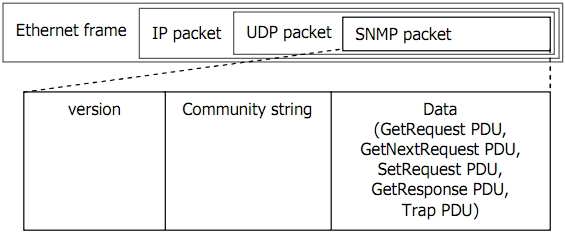
Khi manager gửi không đúng community hoặc khi OID cần lấy lại không nằm trong view cho phép thì agent sẽ không trả lời. Tuy nhiên khi community bị lộ thì một manager nào đó vẫn request được thông tin. Để ngăn chặn hoàn toàn các SNMP manager không được phép, người quản trị có thể dùng đến SNMP access control list (ACL).

SNMP ACL là một danh sách các địa chỉ IP được phép quản lý/giám sát agent, nó chỉ áp dụng riêng cho giao thức SNMP và được cài trên agent. Nếu một manager có IP không được phép trong ACL gửi request thì agent sẽ không xử lý, dù request có community string là đúng.

Đa số các thiết bị tương thích SNMP đều cho phép thiết lập SNMP ACL

1. **Cấu trúc bản tin của SNMP**

SNMP chạy trên nền UDP. Cấu trúc của một bản tin SNMP bao gồm : version, community và data.



Hình 7. Cấu trúc bản tin SNMP

* Version : v1 = 0, v2c = 1, v2u = 2, v3 = 3.

Phần Data trong bản tin SNMP gọi là PDU (Protocol Data Unit). SNMPv1 có 5 phương thức hoạt động tương ứng 5 loại PDU. Tuy nhiên chỉ có 2 loại định dạng bản tin là PDU và Trap-PDU; trong đó các bản tin Get, GetNext, Set, GetResponse có cùng định dạng là PDU, còn bản tin Trap có định dạng là Trap-PDU.

CHƯƠNG 2: TỔNG QUAN VỀ PHẦN MỀM GNS3

1. **Giới thiệu GNS3**

GNS3 là một chương trình giả lập mạng có giao diện đồ họa (Graphical Network Simulator) cho phép chúng ta có thể dễ dàng thiết kế các mô hình mạng và sa đó chạy giả lập trên chúng. Tại thời điểm hiện tại GNS3 chỉ hỗ trợ các IOS của Router, ATM/Frame Relay/Ethernet Switch và Hub. Ta có thể mở rộng mạng của mình bằng cách kết nối nó vào mạng ảo này.

Để làm được điều này , GNS3 đã dựa trên Dynamips và một phần của Dynagen. Nó được phát triển bằng Python và thông qua PyQt, phần giao diện đồ họa thì sử dụng thư viện Qt, vốn rất nổi tiếng về tính hữu dụng trong dự án KDE. GNS3 cũng sử dụng kỹ thuật SVG (Scalable Vector Graphic) để cung cấp các biểu tượng chất lượng cao cho việc thiết kế mô hình mạng.

Dynamips là một chương trình mô hỏng Router Cisco được viết bởi Christophe Fillot. Nó mô phỏng các dòng 1700, 2600, 3600 và 7200, sử dụng các IOS image chuẩn. Phần mềm này có thể được sử dụng cho:

* + - Làm một công cụ để thực tập, với phần mềm sử dụng trong thế giới thực. Nó cho phép mọi người làm quen hơn với các thiết bị của Cisco (hiện đang là công ty hàng đầu trên thế giới về kỹ thuật mạng).
    - Thử nghiệm và làm quen với các đặc tính của Cisco IOS.
    - Kiểm tra nhanh chóng các cấu hình để triển khai sau này trên các Router thật.

Dĩ nhiên, phần mềm mô phỏng này không thể thay thế các Router thật, nó chỉ đơn giản là một công cụ bổ sung cho các bài lab thực tế của các nhà quản trị mạng Cisco hoặc cho những ai muốn vượt qua các kỳ thi CCNA/CCNP/CCIE.

Dynagen là một giao tiếp dựa trên nền văn bản (text-base) dành cho Dynamips, cung cấp một bộ OOP API riêng được sử dụng bởi GNS3 để tương tác với Dynamips.

1. **Cài đặt**
2. *Cài đặt GNS3*

GNS3 chạy trên Windows, Linux và Mac OS X và đòi hỏi các thành phần sau đây đã được cài đặt sẵn trong máy nếu muốn cài đặt từ mã nguồn:

* Qt >= 4.3
* Python >= 2.4
* Sip >= 4.5
* PyQt >= 4.1

Để thuận tiện hơn trên Windows, chúng ta có thể cài đặt từ gói tích hợp tất cả các phần mềm cần thiết bao gồm: WinPcap, Dynamips, và một phiên bản đã được biên dịch của GNS3, giúp chúng ta không cần phải cài Python, PyQt, Qt và Sip.

Phiên bản mới nhất ành cho Win ows tính đến thời điểm này (12/2012) là

0.8.3.1 bao gồm cả Dynamips, Qemu/Pemu, Putty và WinPcap. Địa chỉ download: <http://www.gns3.net/download/>

Sau khi download, ta tiến hành cài đặt GNS3. Chọn tất cả các option trong quá trình cài đặt. Khởi động lại máy nếu có yêu cầu.

Khởi động chương trình GNS3:

Graphical user interface, application

Description automatically generated

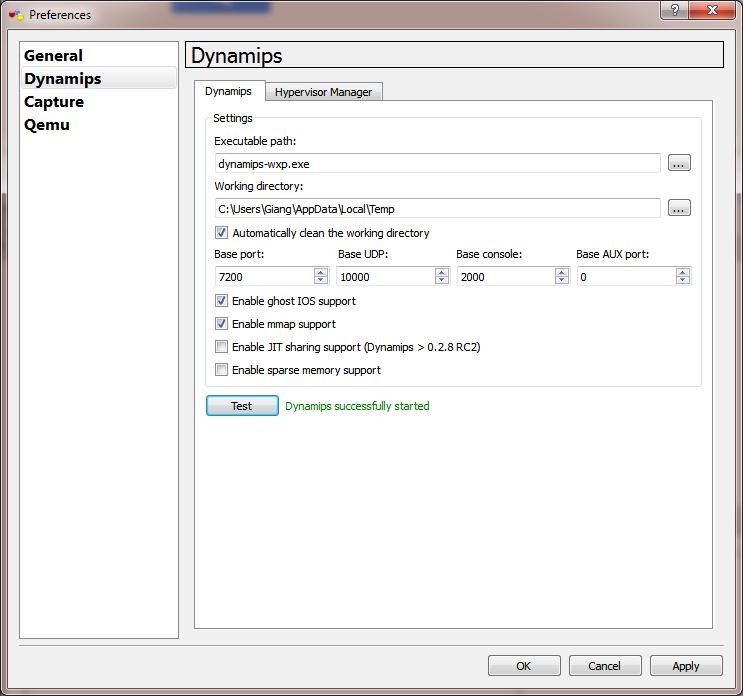
Hình 8. Giao diện chính GNS3

1. *Thiết lập Dynamips và IOS*

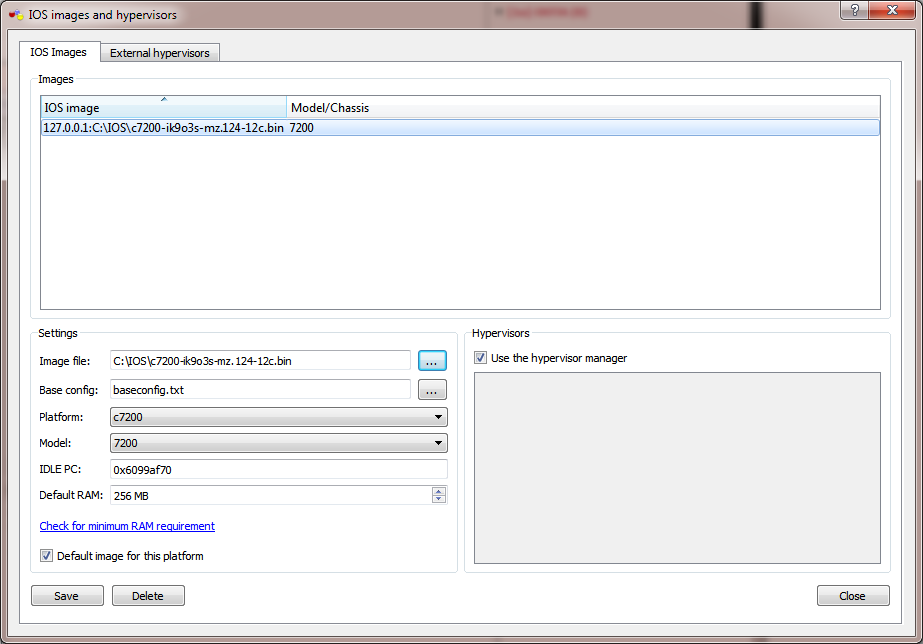
Để GNS3 có thể làm việc được với Dynamips và các IOS, ta cần thiết lập các cấu hình sau:

* Cấu hình với Dynamips: Vào menu Edit – chọn Preferences – chọn Dynamips ở bảng bên trái. Do mặc định D nami s được cài chung trong bộ công cụ GNS3 nên ta không cần kiểm tra đường dẫn, chỉ cần kiểm tra chương trình chạy có lỗi không: Nhấp vào nút Test  Successfully

Hình 9.Thiết lập Dynamips



* Cấu hình các IOS:
* Trước tiên ta download các file IOS của Router Cisco về, và cất vào một thư mục nào đó, ví dụ: C:\IOS
* Trong GNS3, vào menu Edit – IOS images and hypervisors
* Tại đây, với mỗi “Platform” và “model”, ta trỏ đường dẫn tới image tương ứng trong C:\IOS.
* Hình minh họa làm cho Router có Platform ‘c7200” và Model “7200”



Hình 10. Cấu hình IOS

* Làm tương tự cho các Platform 2600 và 3600 (Model 3640). Sau khi làm xong ấn nút Save rồi thoát ra.

1. **Làm việc trên GNS3**
2. *Các công cụ làm việc*

* Thanh menu: Chứa các đường dẫn đến các chức năng tinh chỉnh GNS3 và các thành phần làm việc



Hình 11. Thanh menu

* Thanh công cụ: Chứa các công cụ để thao tác với mô hình mạng



Hình 12. Thanh công cụ

* Cửa sổ thiết bị: Chứa các thiết bị mạng ảo để sử dụng cho mô hình mạng.

Graphical user interface, application

Description automatically generated

Hình 13. Cửa sổ thiết bị

* Cửa sổ Topology: Hiển thị các thông tin sơ bộ về thiết bị và các liên kết trong mạng



Hình 14. Cửa sổ Topology

* Cửa sổ Capture: Chức các thông tin về việc bắt gói tin trên mô hình mạng. Tuy nhiên sau này chúng ta sẽ dùng Wireshark thay thế cho cửa sổ này.



Hình 15. Cửa sổ Capture

* Cửa sổ Console: Cho phép GNS3 thao tác trực tiếp với Dynagen

Graphical user interface, application, Word

Description automatically generated

Hình 16. Cửa sổ Console

1. *Thiết lập Router*

* Để thiết lập một Router, ta khởi tạo một mô hình đơn giản chỉ bao gồm 2 Router: kéo 2 Router c7200 từ cửa sổ thiết bị vào khung làm việc của chúng ta:

Graphical user interface, application

Description automatically generated

Hình 17. Thiết lập Router

1. *Tinh chỉnh các Interface*

* Bản thân một Router sau khi kéo vào khung làm việc chỉ có một khung rỗng (không có các interface), không thể giao tiếp với các thiết bị khác nên ta phải thiết lập các interface này:
* Nhấp chuột phải lên R1, chọn Configure, chọn thẻ Slot.Ta thiết lập 2 interface cho R1:
* C7200-IO-2FE: Cung cấp 2 cổng Fast Ethernet cho thiết bị
* C7200-PA-4T+: Cung cấp 4 cổng Serial cho thiết bị

Graphical user interface

Description automatically generated with medium confidence

Hình 18. Tinh chỉnh Interface

* Sau đó nhấn OK và thoát ra.
* Sau khi đã tinh chỉnh cho các Router xong, ta start lần lượt từng Router lên (thời gian start từ 30s đến 5 phút tùy theo cấu hình PC của bạn)
* Start Router: Chuột phải lên Router, nhấp chọn “Start”

1. *Tinh chỉnh Idle PC*

* Nếu như chúng ta để GNS3 tinh chỉnh mặc định các thông số CPU thì sẽ dễ dàng làm cho CPU của hệ thống hoạt động tới 100% và sẽ luôn nằm ở ngưỡng đó. Điều này là vì Dynamips không biết khi nào Router ảo đang rỗi và khi nào nó đang th c thi công việc. Lệnh Idle PC sẽ giúp ta th c hiện điề nà để GNS3 không chiếm dụng quá nhiều tài nguyên của PC.
* Cách thực hiện: Chuột phải lên Router, chọn Idle PC  Yes  chọn một dòng bất kỳ (nếu có dòng được đánh dấu \* thì ta chọn dòng đó)
* Chúng ta tinh chỉnh Idle PC cho tất cả các thiết bị Router trong mô hình mạng.

1. *Thiết lập kết nối*

* Trước khi thiết lập kết nối, chúng ta nên bật tùy chọn Show Interface Label trên thanh công cụ để dễ dàng hơn trong quá trình làm việc với Router:



* Nhấ vào “A a link” tr n thanh công cụ, l ta chọn loại liên kết muốn thiết lập. Với bài minh họa này, ta chọn “Fast Ethernet”

Graphical user interface, application, Word

Description automatically generated

Hình 19. Thiết lập kết nối

* Nhấn 2 đầu kết nối vào 2 Router. Mặc định GNS3 sẽ tự chọn cổng kết nối trên các Interface chúng ta đã chọn ở mục 3.3.
* Ở 2 đầu của Interface có 2 nút tròn, biểu hiện cho 2 cổng kết nối trên 2 thiết bị, với tên cổng trên đó.

1. *Cấu hình Router thông qua giao diện console*

* Để cấu hình Router, ta nhấp chuột phải lên Router và chọn Console.
* Khi bật console lúc Router mới khởi động xong, ta nên chọn “no” tất cả để đỡ mất thời gian chỉnh các thông tin trên Router.
* Ở đây, ta sẽ cấu hình đơn giản cho 2 cổng trên 2 Router R1 và R2 để chúng có thể nhìn thấy nhau (ping được với nhau).
* Trên R1:

Text

Description automatically generated

Hình 20. Cấu hình R1

* Trên R2:

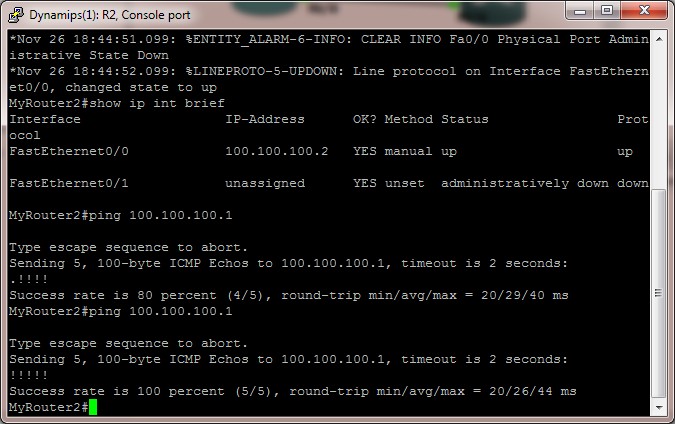
Text

Description automatically generated

Hình 21. Cấu hình R2

* Kiểm tra thông số các cổng và ping kiểm tra:

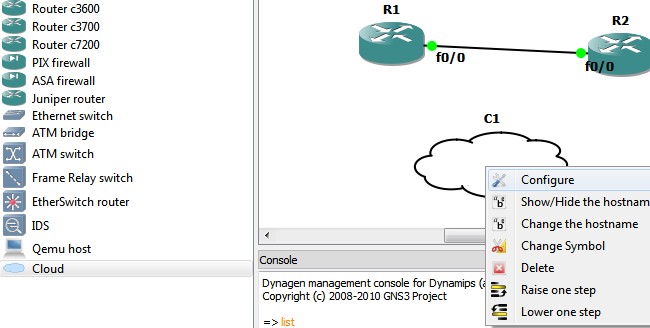
Hình 22. Kiểm tra kết nối



1. *Giao tiếp với PC thật*

* Dynamips có thể tạo cầu nối giữa Interface trên Router ảo với interface trên máy thật, cho phép mạng ảo giao tiếp với mạng thật. Để sử dụng tính năng này, bạn phải tạo thêm một thiết bị “Cloud”:
* Kéo thả một “Cloud” từ cửa sổ công cụ vào khung làm việc của chúng ta, sau đó nhấp chuột phải lên C1, chọn Configure:

Hình 23. Cấu hình Cloud



* Trong phần “ IO Ethernet”, ta chọn loại thiết bị để GNS3 giao tiếp với máy thật. Để tránh xung đột với mạng bên ngoài, ta nên chọn 1 card cho VMWare hỗ trợ.
* Trong ví dụ này ta chọn VMWare VMNet 1, nhấp chọn Add rồi OK:

Graphical user interface, text, application, email

Description automatically generated

Hình 24. Chọn card VMWare

* Bây giờ Cloud đã có Interface để giao tiếp với mô hình (cụ thể là ta mượn Interface “VMNet1” của chương trình VMWare.
* Thiết lập kết nối: Chọn một Link dạng “Fast Ethernet” từ thanh công cụ, nối từ R1 đến C1:

Graphical user interface, application

Description automatically generated

Hình 25. Thiết lập kết nối R1-C1

* Text

  Description automatically generatedCấu hình IP cho Interface f0/1 của R1:

Hình 26. Cấu hình IP f0/1 của R1

* Cấu hình IP cho card mạng VMNet1:

Graphical user interface, text, application

Description automatically generated

Hình 27. Cấu hình IP VMNet1

Chương 3: Tổng quan về MIB Browser

1. **Giới thiệu chung**

The iReasoning MIB browser là một công cụ mạnh mẽ và dễ sử dụng được cung cấp bởi iReasoning SNMP API. MIB browser là một công cụ không thể thiếu cho các kỹ sư để quản lý các thiết bị mạng SNMP kích hoạt các thiết bị mạng và ứng dụng. The iReasoning MIB browser cho phép người dùng đưa ra các yêu cầu về SNMP ( SNMP request ) để lấy thông tin dữ liệu của SNMP Agent hoặc thực hiện các thay đổi thông tin của các Agent.

1. **Tính năng chính**

* Giao diện trực quan.
* Hỗ trợ đầy đủ SNMPv1, v2c và v3 (USM và VACM).
* Hỗ trợ đầy đủ SNMPv3 USM, bao gồm cả các thuật toán HMAC-MD5, HMAC-SHA, CBC-DES, CFB128-AES-128, CFB128-AES-192, CFB128-AES-256.
* Bộ phân tích SMIv1/SMIv2 MIB mạnh mẽ.
* Hỗ trợ IPv6.
* Trap Receiver (nơi nhận Trap) với công cụ có quy tắc để xử lý trap và kích hoạt các hoạt động nếu các điều kiện nhất định được thỏa mãn.
* Trap Sender (nơi gửi Trap).
* Log window để hiển thị ứng dụng đăng nhập và các gói tin SNMP được trao đổi giữa browser và agents.
* Hiển thị dạng bảng cho MIB tables.
* Quản lý người dùng SNMPv3 USM.
* Hiển thị port cho thẻ giao diện mạng.
* Công cụ biểu đồ hiệu suất để theo dõi các giá trị số OID.
* Thiết bị chụp.
* Thiết bị chụp Cisco.
* Công cụ ping và traceroute.
* Công cụ phát hiện mạng.
* So sánh các SNMP Agents.
* Đánh dấu.
* Chạy trên Windows, Mac OS X, Linux và các nền tảng UNIX khác.

1. **Yêu cầu cài đặt**

* Windows, Mac OS X, Linux và các nền tảng UNIX khác.
* Nếu trên Linux / Unix, SUN JRE 1.5 hoặc phiên bản sau này phải được cài đặt và lệnh java phải nằm trong đường dẫn hệ thống.

1. **Tải và chạy MIB Browser**

* Tải MIB browser installer hoặc file zip. Trên Windows, chạy setup.exe để cài đặt MIB browser. Trên Mac OS X, Linux và các nền tảng UNIX khác, giải nén nó vào thư mục mong muốn.
* Chạy MIB Browser:
  + Trên Windows, hãy nhấp vào biểu tượng MIB Browser để bắt đầu
  + Trên Linux / Unix, nhập vào thư mục MIB Browser và chạy browser.sh để bắt đầu.
  + Trên Mountain Lion, kéo thư mục "mibbrowser" vào thư mục ứng dụng của bạn, hoặc nhấp đúp chuột vào biểu tượng ứng dụng "MIB Browser" trong thư mục "mibbrowser" để bắt đầu MIB Browser, hoặc mở một thiết bị đầu cuối và nhập thư mục ireasoning / mibbrowser sau đó chạy tập lệnh browser.command để bắt đẩu nó. Trong các phiên bản trước đó của Mac OS X, nhập thư mục ireasoning / mibbrowser và nhấp đúp chuột vào biểu tượng browser.command để bắt đầu MIB browser.

Trên Linux / Unix / Mac OS X, nếu bạn đăng nhập như một người dùng non-root và cần phải chạy trap receiver tại cổng UDP 162, bắt đầu MIB browser bằng cách sử dụng lệnh “sudo browser.sh”.

1. **Giao diện**

Graphical user interface, application

Description automatically generated

Hình 28. Giao diện chính của MIB Browser

**Menu (Trình đơn)**

* + *File menu*
  + *Load MIB*
* Để khởi động một hoặc nhiều file dạng MIB, bạn cần phải giữ phím CTRL nếu muốn chọn nhiều file.
  + *Unload MIB*
  + *MIB Modules*
* Danh sách các thuộc tính của tất cả các module MIB đã tải.
  + *Open Session*
* Mở một tập tin giao dịch đã lưu trước đó.
  + *Save Session*
* Lưu các thiết lập hiện tại của các thẻ chứa tập tin đang giao dịch, mà có thể được mở ra sau đó để khôi phục lại các thẻ.
  + *Exit*
* Thoát trình duyệt.

**Edit menu**

* + *Find in MIB Tree*
* Tìm một nút trong cây MIB.
  + *Find in Result Table*
* Tìm một chuỗi trong bảng kết quả.

**Operations menu**

* + *Get*
* Thực hiện phương thức GetRequest của SNMP (Manager gửi GetRequest cho agent để yêu cầu agent cung cấp thông tin nào đó dựa vào ObjectID).
  + *Get Next*
* Thực hiện phương thức GetNextRequest của SNMP (Manager gửi GetNextRequest có chứa một ObjectID cho agent để yêu cầu cung cấp thông tin nằm kế tiếp ObjectID đó trong MIB).
  + *Get Bulk*
* Thực hiện phương thức GetBulkRequest. GetBulk là phương thức có từ  SNMPv2, nó có thể  lấy thông tin của nhiều object cùng lúc chỉ  bằng 1 bản tin; ví dụ thay vì bạn gửi 10 GetRequest thì bạn chỉ  cần gửi 1 GetBulk.
  + *Set*
* Thực hiện phương thức SetRequest của SNMP (Manager gửi SetRequest cho agent để đặt giá trị cho đối tượng của agent dựa vào ObjectID).

Lưu ý, Trong panel bên trái, những object có hình lá cây là read-only, hình chiếc bút là read-write, chỉ có thể thực hiện phương thức SetRequest với object dạng read-write. Để thiết lập nhiều biến số, bạn có thể giữ phím CTRL và chọn nhiều hàng trong cửa sổ kết quả, sau đó chọn SET operation.

Sau khi chọn Set, hộp thoại sau sẽ xuất hiện:

Graphical user interface, application

Description automatically generated

Hình 29. Hộp thoại Set

Trong hộp thoại [SNMP SET] bạn nhập giá trị  cần thay đổi vào [Value] và nhấn [OK], nếu write community là đúng thì thao tác Set sẽ được thực hiện thành công.

Lưu ý:

1. Đối với loại BITS dữ liệu, bạn cần phải sử dụng ký hiệu toán học cho một tập hợp các số nguyên, ví dụ {1, 3, 8}.

2. Các định dạng cho chuỗi thập lục phân là (0x [0-9A-Fa-f] [0-9A-Fa-f]) +. Ví dụ, 0x12 0xA1 0x30.

* + *Get Subtree:* Thực hiện lấy tất cả các object nằm dưới một node. Ví dụ bạn đang ở node mib-2.system, khi dùng GetSubTree thì phần mềm sẽ lấy tất cả các OID nằm dưới một cấp. GetSubTree không phải là một phương thức chuẩn của SNMP mà là tính năng của riêng phần mềm, bản chất của việc thực hiện GetSubTree là thực hiện các GetNextRequest liên tục đến khi nào nhận được object cùng cấp với object khởi đầu.

Graphical user interface, text, application, email

Description automatically generated

Hình 30. Get Subtree

* + *Walk:* Thực hiện GetNextRequest liên tục đến khi hết bảng mib. Chỉ cần một lần Walk bạn có thể lấy toàn bộ các object nhưng sẽ tốn nhiều thời gian. Walk cũng không phải là phương thức chuẩn của SNMP.
  + *Table View:* MIB Browser cho phép GET thông tin dưới dạng bảng , cung cấp cái nhìn trực quan hơn về SNMP Agent. Chẳng hạn cần xem bảng định tuyến của Router , bạn tìm đến mục ipRoute Table trong thư mục ip , sauđó click chuột phải vào ipRoute Table và chọn Table View. Kết quả sẽ xuất hiện ở Result Tabble.

Graphical user interface, application

Description automatically generated

Hình 31. Result Table

Nút trên thanh công cụ của table view:

|  |  |
| --- | --- |
| **Rotate button** | Xoay bảng một góc 90°. |
| **Refresh button** | Làm mới bảng ngay lập tức. |
| **Export** | Xuất bảng dữ liệu vào một tập tin CSV. |
| **Poll button** | Làm mới bảng theo định kỳ. |
| **SNMP SET** | Thực hiện thao tác SET cho cột được chọn trong bảng. |
| **Create Row** | Tự động tạo ra một hàng. |
| **Delete Row** | Tự động xóa một hàng. |

**Tools menu**

* *Trap sender*

Graphical user interface

Description automatically generated

Hình 32. Cửa sổ TrapSender

Cửa sổ này cho phép người dùng gửi đi các trap SNMPv1/v2c hoặc các thông báo. Cài đặt trap có thể được đánh dấu lại, do đó bạn sẽ không phải nhập lại lần sau. Đối với trap nút trong cây MIB, bạn có thể bấm chuột phải, chọn “Send Trap” để mở cửa sổ này, và các thông tin từ trap nút sẽ được dùng để điền vào các giá trị mặc định.

* *Watches và Add Watch:* Watch là tính năng cho phép thiết lập lịch cho hoạt động của MIB Brower.

Để hẹn giờ thực hiện lệnh cho một đối tượng, đầu tiên ta cần chọn đối tượng trong MIB Tree.

Graphical user interface, application

Description automatically generated

Hình 33. Chọn đối tượng trong MIB Tree

Sau đó chọn Menu Tool, vào Add Watch (hoặc chuột phải chọn Add Watch).Giá trị OID và NAME đã được điền sẵn, bạn chỉ cần chọn Operation mong muốn.

Graphical user interface, text, application, email

Description automatically generated

Hình 34. Add to Watches

Tiếp tục click vào Configue Actions để lựa chọn hình thức thông báo, bạn có thể chọn thông báo qua email, phát chuông cảnh báo khi SNMP Agent không có phản hồi hoặc báo chuông nếu Agent không ở trạng thái bình thường. Click OK để hoàn thành thiết lập Watch.

Graphical user interface, application

Description automatically generated

Hình 35. Configure Actions

Mặc định , các Watch sẽ được thực hiện 60 giây một lần. Để thay đổi thông số này, bạn vào Menu Tool, chọn Watch để xem các Watch đã được thiết lập. Click Setting, điền thời gian Query vào ô QueryInterval, mục “Remove data older than” để thiết lập thời gian xóa các Watch.

Graphical user interface, application

Description automatically generated

Hình 36. Cài đặt MIB Watches

* *Port View:* Tính năng này cho phép hiển thị băng thông Input/ Output và xác suất lỗi của tất cả các cổng thuộc SNMP Agent.

Graphical user interface, application

Description automatically generated

Hình 37. Port View

* *Switch Port Mapper:* Công cụ thể hiện bản đồ các thiết bị được kết nối với managed switch, lượng băng thông được sử dụng ở các cổng cũng được hiển thị.

Graphical user interface, application, table, Excel

Description automatically generated

Hình 38. Switch Port Mapper

* *Device Snapshot:* Hiển thị một bản chụp xem dữ liệu thiết bị có sẵn, bao gồm cả thông tin hệ thống, giao diện, các ổ đĩa, các quy trình, phần mềm cài đặt, …

Graphical user interface

Description automatically generated

Hình 39. Device Snapshot

* *Cisco Device Snapshot:* Hiển thị ảnh chụp của các thiết bị Cisco.

Graphical user interface

Description automatically generated

Hình 40. Cisco Device Snapshot

Nếu bạn có nhiều OIDs cần theo dõi, bạn có thể sử dụng màn hình cài đặt để thêm OIDs bổ sung. Các giá trị này sẽ được hiển thị trong panel “Basic Information”. Các biểu đồ được sử dụng có hai loại là biểu đồ tròn hoặc biểu đồ cột.

* *Log Window:* Mở log window để hiển thị thông tin đăng nhập của ứng dụng, để xóa các nội dung, click chuật phải vào cửa sổ này và chọn “Clear text” trong menu.
* *Ping:* Mở cửa sổ Ping bằng cách vào Menu Tool -> Ping. Sau đó điền địa chỉ IP vào ô Host, click Ping, kết quả sẽ hiển thị.

Graphical user interface, text, application, email

Description automatically generated

Hình 41. Ping

* *Trace Route:* Mở cửa sổTrace Route bằng cách vào Menu Tool -> Trace Route , điền địa chỉ IP đích vào ô Host, click Trace Route và xem kết quả.

Graphical user interface, text, application

Description automatically generated

Hình 42. Trace Route

* *Network Discovery:* Mở cửa sổ Network Discovery trong Menu Tool , sau đó điền địa chỉ mạng LAN (network address) cần khám phá ( chẳng hạn 192.168.1.0 ). Sau đó bấm Start . Tính năng này cho phép xem có những host nào đang hoạt động trong mạng LAN , nếu host đó là một SNMP Agent thì thông tin về hệ thống của host sẽ được hiển thị trên bảng kết quả.

Graphical user interface, application

Description automatically generated

Hình 43. Network Discovery

* *Manage SNMPv3 USM Users*

Table

Description automatically generated

Hình 44. Manage SNMPv3 USM Users

Cửa sổ này hiển thị một danh sách người dùng SNMPv3 hiện có. Sau đó bạn có thể chỉnh sửa các thuộc tính của người dùng hoặc xóa nó.

* *Compare Devices:* Công cụ này cho phép so sánh hai SNMP Agent với nhau. Bạn chỉ cần nhập IP của 2 Device cùng các tham số bổ sung trong cửa sổ Advanced. Sau đó chọn OID và Operation để so sánh, cuối cùng click OK để xem kết quả. Các thông số khác nhau sẽ được đổi màu chữ sang xanh để dễ phân biệt.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Hình 45. Compare Devices

* *Options*

Graphical user interface, text, application, email

Description automatically generated

Hình 46. Options

**General tab**

|  |  |
| --- | --- |
| **SNMP Retries** | Số lần thử lại cho các SNMP queries. |
| **SNMP Timeout** | Giá trị thời gian chờ cho SNMP queries, tính bằng giây. |
| **Lenient Parsing** | Nếu được đánh dấu, bộ phân tích MIB sẽ bỏ qua các lỗi syntax trong file MIBs. |
| **Logging Level** | Mức độ logging của logger. Nếu đó là "DEBUG", SNMP PDU(Protocol Data Unit: phần dât trong bản tin SNMP) sẽ được in ra Log window, nhưng nó sẽ làm giảm hiệu suất của MIB browser. |
| **Trap** **Port** | Số cổng của trap receiver. |
| **Max Token Number** | Số lượng lớn nhất các thẻ cho phép trong trường mô tả của trap receiver. Ví dụ, nếu giá trị là 6, thì thông báo ".iso.org.dod.internet.mgmt.mib-2.interfaces.ifTable.ifEntry.ifAdminStatus.3" sẽ được rút ngắn còn "Â¡Â.mib-2.interfaces.ifTable.ifEntry.ifAdminStatus.3". |
| **Single Tree Root** | Nếu được đánh dấu, MIB tree sẽ có một nút gốc duy nhất, đó là nút *.iso*. Nếu không, mỗi MIB module sẽ có một nút gốc của riêng nó. MIB browser cần khởi động lại để tùy chọn này có hiệu lực. |
| **Agent's Character Encoding** | Với một SNMP agent ở một hệ điều hành sử dụng ngôn ngữ địa phương, có thể sẽ trả về một kí tự mã hóa sai khác. Bạn có thể thay đổi các ký tự để hiển thị một cách chính xác giá trị trả lại từ các agent. |
| **Max Graph Data Count** | Số lượng tối đa của dữ liệu trong bộ nhớ của một đồ thị. |
| **Non Repeaters** | Không lắp lại giá trị của SNMP GET-BULK requests. |
| **Max Repetitions** | Giá trị lớn nhất lặp lại của SNMP GET-BULK requests. |
| **Minimize to system tray** | Nếu được đánh dấu, MIB browser sẽ thu nhỏ lại ở khay hệ thống khi bạn đóng nó. |

Graphical user interface

Description automatically generated

Hình 47. General Tabs

* Default Values tab: Tab này cho phép người dùng thiết lập các thuộc tính mặc định cho các SNMP agent mới.
* Agent tab:

Graphical user interface, application

Description automatically generated

Hình 48. Agent Tab

Bảng này lưu trữ các thuộc tính của các SNMP agent được truy cập. Bạn có thể thêm hoặc xóa các agent từ bảng này.

Đối với SNMPv1/v2c agents, các giá trị User, Authentication Protocol, Authentication Password, Privacy Protocol và Privacy Password được bỏ qua. Đối với SNMPv3 agents, các giá trị Read Community và Write Community được bỏ qua.

* MIB Files Tab:

Graphical user interface, text, application, email

Description automatically generated

Hình 49. MIB Files Tab

Bảng này lưu trữ tin các tập tin của MIBs đã tải. Bạn có thể thêm hoặc xóa MIBs từ bảng này

**Toolbar (thanh công cụ)**

Graphical user interface, application

Description automatically generated

Hình 50. Toolbar

* + *Address:* Điền địa chỉ IP của SNMP Agent.
  + *Advanced Button:* Bổ sung các tham số như Port, SNMP version.

Graphical user interface, text, application, email

Description automatically generated

* + *OID field:* Object identifier , được sử dụng để xác định đối tượng truy vấn, mỗi đối tượng truy vấn sẽ có một OID khác nhau
  + *Operations:* Xác định lệnh mà MIB sẽ thực hiện
  + *Go button:* Thực hiện lệnh

**MIB Tree Pane**

Graphical user interface, table

Description automatically generated

Hình 51. MIB Tree Pane

MIB Tree Pane được chia ra làm 2 phần là phần hình cây ở trên và phần định nghĩa bên dưới

* + *Tree Pane:* Hiển thị cây MIB. Nhấp chuột phải vào một nút thì một pop up menu sẽ xuất hiện. Tùy thuộc vào đặc điểm của từng nút, nếu click chuột phải vào nút gốc, pop up menu sẽ gồm: “Find”, “Expand” và “Unload MIB”. Ở các nút khác, sẽ chỉ có “Find” và “Expand”.

Table

Description automatically generated

Hình 52. Tree Pane

* + *Node Properties Pane:* Các định nghĩa về object bao gồm Name, OID, MIB, Syntax, Access, Status. Phần mềm MIB Browser biết được các thông tin này chính là nhờ  vào RFC1213, mib file này được lưu ở đường dẫn cài đặt chương trình:

“..\ireasoning\mibbrowser\mibs\rfc1213”

**Result Pane**

Graphical user interface

Description automatically generated

Hình 53. Result Pane

Kết quả của các truy vấn SNMP sẽ được hiển thị trong bảng.

Các nút trong thanh công cụ bên phải:

|  |  |
| --- | --- |
| *Stop button* | Dừng cấp phát SNMP queries. |
| *Clear Table button* | Xóa bảng kết quả |
| *Raw Data button* | Mở một cửa sổ hộp thoại mới hiển thị các kết quả của SNMP queries. |
| *Find button* | Tìm các chuỗi trong cửa sổ kết quả. |
| *Save button* | Lưu các giá trị trong cửa sổ kết quả vào một file XML. |
| *Open button* | Tải các giá trị từ một tập tin XML để cửa sổ kết quả. |

TRAP RECEIVER

1. **Cửa sổ chính**

Trap Receiver có thể được mở từ cửa sổ trình duyệt MIB hoặc bắt đầu độc lập.

Graphical user interface, text, application, email

Description automatically generated

Hình 54. Trap Receiver

Cửa sổ trap receiver được chia làm 2 phần. Phần trên thể hiện tóm tắt về trap, phần dưới thể hiện chi tiết về trap được chọn.

|  |  |
| --- | --- |
| **Menu Items of Trap Receiver** | |
| **Trap Filter button** | Sử dụng bộ lọc cho tất cả các trap nhận được. Một trap có thể bị chặn dựa trên địa chỉ IP và OID. |
| **Start Trap Receiver button** | Chạy trap receiver. Nó sẽ được chuyển sang màu xám nếu trap receiver đang chạt. Trap receiver không tự động chạy khi cửa sổ mở ra. |
| **Stop button** | Dừng trap receiver. |
| **Add Trap Rule** | Thêm rule mới cho các trap thực thi. Nếu một dòng của trap table được chọn, thì hộp thoại sẽ populate các trường với dữ liệu từ các hàng này.  Tất cả rules sẽ được áp dụng cho trap đến, ngoại trừ rule specifies. Rules được sắp xếp bằng tên. Vì vậy bạn có thể thay đổi tên để thay đổi thứ tự thực hiện của nó. |

Graphical user interface, text, application, email

Description automatically generated

Hình 55. Add Trap Rule

**Rule Conditions:**

|  |  |
| --- | --- |
| *snmpTrapOID is* | SnmpTrapOID trong SNMPv2 traps, hoặc "entepriseOID + '.0.' + specific" cho SNMPv1 traps. |
| *Allowed trap source IPs* | Các IP của trap originator, ngăn cách bởi dấu phẩy. |
| *Raw values of variable bindingds contains* | Bạn có thể nhập nhiều từ khóa ở đâ, cách nhau bằng dấu gạch ngang hoặc dấu phẩy. Nếu một trong những từ khóa được tìm thấy, sau đó điều kiện được coi là đáp ứng. |
| *Manage Rules* | Quản lý rules, bao gồm chỉnh sửa và xóa rules. |
| *Clear Table button* | Xóa các nội dung trong trap table. |
| *Export Table to CSV* | Xuất bảng dữ liệu vào một tập tin CSV. |
| *Options* | Cài đặt trap receiver. |

**Tùy chọn**

* *General Tab*

Graphical user interface, text, application, email

Description automatically generated

Hình 56. Trap Receiver General Setting

|  |  |  |
| --- | --- | --- |
| **Trap** **Port** | Số cổng của trap receiver. | |
| **Forward Trap To** | Chuyển tiếp traps tới trap receiver trong SNMPv1 PDUs. | |
| **Save trap data to database** | Nếu được kích hoạt, received traps sẽ được lưu vào cơ sở dữ liệu gói. Sau đó nó sẽ được tải trở lại. | |
| **Show trap message window** | Nếu được kích hoạt, một cửa sổ thông báo trap sẽ xuất hiện xung quanh khay hệ thống khi có trap tới. | |
| **Play sound** | Mở tiếng khi trap đến. | |
| **Execute command** | Thực thi một lệnh khi có trap đến. | |
| **Tokens used in execute command** | |  |
| **$ip** | Địa chỉ IP của trap originator |  |
| **$oid** | Trap OID |  |
| **$ts** | Giá trị nhãn thời gian |  |
| **$sp** | Giá trị của trường trap SNMPv1 đặc thù. |  |
| **$vb.all** | Toàn bộ thông báo bằng định dạng HTML, sẽ được hiển thị tại bảng dưới của trap receiver. |  |
| **$vb.n** | Tất cả các biến được gán |  |

* *SMTP Tab*

Graphical user interface, text, application, email

Description automatically generated

Hình 57. Trap Receiver SMTP Setting

Thẻ này cho phép chuyển tiếp SNMP trap qua email. Nếu nó được kích hoạt thì máy chủ phải được cấu hình. Nếu máy chủ thứ cấp cũng được cấu hình thì nó sẽ được sử dụng để gửi email khi máy chủ SNMP bị lỗi.

* *Email Template Tab (chỉ có ở phiên bản doanh nghiệp).*

Graphical user interface, text, application, Word

Description automatically generated

Hình 58. Trap Receiver Email Template Setting

Nếu mẫu email được kích hoạt, tất cả các email gửi đi từ các trap module sẽ sử dụng mẫu.

|  |  |
| --- | --- |
| **Tokens used in template** | |
| **$ip** | Địa chỉ IP của trap originator |
| **$oid** | Trap OID |
| **$ts** | Giá trị nhãn thời gian |
| **$sp** | Giá trị của trường trap SNMPv1 đặc thù. |
| **$trap** | Toàn bộ thông báo bằng định dạng HTML, sẽ được hiển thị tại bảng dưới của trap receiver. |
| **$vb.all** | Tất cả các biến được gán |
| **$vb.n** | Biến thứ n được gán. Ví dụ, $vb.1 là biến đầu được gán, $vb.2 là biến thứ 2. |

* *SNMPv3 Trap Receiver Tab*

Graphical user interface

Description automatically generated

Hình 59. Trap Receiver SNMPv3 Setting

Bảng này lưu trữ các thuộc tính của SNMPv3 Trap Receiver. Bạn có thể thêm hoặc xóa các mục từ bảng này.

**Command Line Options (Tùy chọn dòng lệnh)**

* Command line tools được đặt tại đường dẫn “$INSTALL\_DIR\bin”.
* MIB browser có thể được mở từ các dòng lệnh sau:

|  |  |
| --- | --- |
| browser.bat -h (or Â¨Chelp, -?) | Print usage |
| -h <h> | Specify the host name or IP address of SNMP agent. |
| -p <p> | Specify the port number of the SNMP agent. Default value is 161. |
| -v <1|2|3> | Specify the SNMP version number. Possible values are 1, 2, and 3. |
| -c <c> | Specify the SNMP community name. Default value is public. |
| -o <o> | Specify OID. |
| -a <a> | Specify the SNMP action. Possible values are (get|getnext|getsubtree|walk|gettable). |
| -f <filename> | Output CSV file name. Used with gettable action. |
| -s <filename> | Open a saved session on startup. |
| -w | Open watch tab on startup. |

* Command Line Graph Tool: Lệnh *graph.bat* được sử dụng để ghi lại các giá trị của SNMP agent vào tập tin, có thể được mở trong MIB Browser bằng cách sử dụng "File/Open Graph Data" menu.
* Command Line SNMP Tools:
  + *snmpget.bat* script sử dụng để thực hiện SNMP GET request.
  + *snmpgetnext.bat* script sử dụng để thực hiện SNMP GET-NEXT request.
  + *snmpset.bat* script sử dụng để thực hiện SNMP SET request.
  + *trap.bat* script sử dụng để gửi SNMP traps.

**Phím tắt**

|  |  |
| --- | --- |
| CTRL-L | Tải thư mục MIB |
| CTRL-F | Tìm một nút MIB |
| CTRL-G | Thực hiện thao tác SNMP GET |
| CTRL-N | Thực hiện thao tác SNMP GET-NEXT |
| CTRL-E | Thực hiện thao tác Get-Subtree |
| CTRL-S | Thực hiện thao tác SNMP SET |
| CTRL-W | Thực hiện thao tác walk |
| CTRL-P | Dừng thao tác hiện tại |
| Enter | Nếu nút Operations hoặc Go đang có trọng tâm, nhấn phím "Enter" sẽ lặp lại hoạt động cuối cùng. |
| CTRL-T | Thực hiện Table view |
| CTRL-R | Thực hiện Plot graph. |
| CTRL-D | Thực hiện Add to watches |
| CTRL-O | Mở watches panel |
| CTRL-I | Mở trap receiver panel |

KẾT LUẬN

Qua bài báo cáo trên, chúng ta đã phần nào thấ được sức ảnh hưởng to lớn của việc áp dụng các nguyên tắc sáng tạo trong việc giải quyết các vấn đề trong tin học nói riêng, và trong các lĩnh vực khác của đời sống nói chung.

Xã hội ngày càng phát triển thì nhu cầu của con người cũng ngày càng cao. Để đáp ứng được nhu cầu đó, đòi hỏi chúng ta phải luôn cố gắng phát huy sáng tạo. Muốn làm được điều này, chúng ta không thể chỉ giải quyết vấn đề mà còn phải tìm ra được cách giải quyết vấn đề nhanh chóng và hiệu quả nhất. Các nguyên tắc sáng tạo cơ bản sẽ giúp chúng ta tăng cường khả năng sáng tạo của mình. Từ đó, chúng ta không chỉ cải tiến những sự vật sự việc đang tồn tại, mà còn có khả năng sáng tạo ra những cái mới hơn, làm cho cuộc sống ngày càng hoàn thiện hơn.