MỤC LỤC

[MỤC LỤC 1](#bookmark0)

DANH MỤC CÁC HÌNH 4

DANH MỤC CÁC BẢNG 5

DANH MỤC CÁC TỪ VIẾT TẮT YÀ MỘT số KHÁI NIỆM 6

[ĐẶT VẤN ĐỂ 9](#bookmark3)

[Chương 1 : Tổng Quan Về Giao Thức TCP/IP 10](#bookmark4)

1. - Mô hình tham chiếu mở OSI ( Open System Interconnection).... 10
2. - Giới thiệu mô hình OSI 10
3. - Các nguyên tắc khi xây dựng mô hình tham chiếu OSI 10
4. — Chức năng của các tầng trong mô hình tham chiếu OSI 10
5. - Bộ giao thức TCP/IP 11
6. -Giới thiệu về TCP IIP 11
7. Kiến trúc của TCP IIP 12
8. - Giao thức liên mạng TCP (Tranmission Control Protocal) 13
9. - Giao thức truyền thông IP (Internet Protocol) 15
10. - Giao thức thông báo điều khiển liên mạng (ICMP) 18
11. - Một số ứng dụng của TCPHP 20

[Chương 2 : Tìm hiểu về công nghệ IPSec 22](#bookmark5)

1. - IPSec (Internet protocal security) 22
2. - Tổng quan về IPSec 22
3. - Các ứng dụng của IPSec 22
4. — Mục đích của IPSec 23
5. - Hiện trạng của IPSec 23
6. - Lợi ích của IPSec 24
7. - Các ứng dụng định tuyến 25
8. — Kiến trúc IP Security 25
9. - Các tài liệu vềIPSec 25
10. - Các dịch vụ IPSec 27
11. - Các liên kết an toàn 28
12. - Header xác thực (Authentication Header) 31
13. - Dịch vụ Anti-Replay 32
14. - Giá trị kiểm tra tính toàn vẹn (Integrity Check Value): 33
15. — Các phương thức Transport và Tunnel: 34
16. - Khối an toàn tóm lược ịEncapsulating Security Payload) 35
17. -Dạng ESP 35
18. - Thuật toán mã và xác thực 36
19. - Padding: 36
20. - Quá trình xử lý gói tin trong ESP 36
21. - Các phương thức Transport và Tunnel 38
22. — Quản lý khóa 41
23. Giao thức xác định khóa Oakley (Oakley Key Determination

Protocol) 42

1. - ISAKMP(Internet Security Association and Key Management Protocol) 45
2. - Dạng header của ISAKMP: 45
3. - Các kiểu ISAKMP payload: 46
4. - Các trao đổi ISAKMP 48
5. - IKE (Internet Key Exchange) 51
6. - Giới thiệu về IKE 51
7. - Sự trao đổi IKE ịIKE Exchanges) 51
8. — Chế độ trao đổi chính của IKE (Main Mode Exchange) 51
9. - Chế độ trao đổi nỗ lực (Aggressive Mode Exchange) 51
10. - Chế độ trao đổi nhanh (Quick Mode Exchange) 51
11. - Chế độ trao đổi khác (Other IKE Exchanges) 52

[Chương 3 : ứng dụng của IPSec trong bảo mật thông tin trên mạng 53](#bookmark6)

1. - Tìm hiểu về bộ phần mềm OpenS/wan 53
2. - OpenS/wan 53
3. - Chức năng của OpenS/wan 53
4. - Cấu trúc của OpenS/wan 53
5. -PLUTO 53
6. KLIP 56
7. - ipsecXInterfaces 57
8. - Gối Caching đầu tiên 57
9. — Nghiên cứu dẫn đường MTU 57

3.13.2.4-KLIPS’Downside 58

1. - Cách cài đặt OpenS/wan 60
2. - ứng dụng của OpenS/wan để thiết lập mạng an toàn 67
3. - ứng dụng trong bảo mật mạng lan-to-lan 67
4. -Cài đặt OpenS/wan 67
5. - Chạy OpenS/wan 69
6. - Kiểm tra trạng thái cài đặt OpenS/wan 70
7. - Sửa một số lối trạng thái phổ biến 70
8. - Các bước cấu hình VPN (Sử dụng khóa RSA) 71
9. - ứng dụng trong bảo mật mạng Road Warrior 84

3.2.2.ỉ - Road Warrior 84

1. — Lấy leftrsasigkey 84
2. - Lấy rỉghtrsasỉgkey 84
3. - Tùy chỉnh file letclipsec.conf. 85
4. - Bắt đầu kết nối 86
5. - MASQ hoặc NAT những gói packet trong đường hầm 86
6. - Kiểm tra kết nối 86

[KẾT LUẬN 87](#bookmark8)

[TÀI LIỆU THAM KHẢO 88](#bookmark9)

Hình 1.1 : Mô hình tham chiếu OSI 10

Hình 1.2 : Mô hình TCP/IP 12

Hình 1.3 : Mô hình hoạt động của TCP/IP 13

Hình 1.4 : Định dạng segment - TCP 14

Hình 1.5 : Đinh dạng của một IP datagram 16

Hình 1.6 : Chi tiết header của IP packet 16

Hình 1.7 : Định dạng Flags 17

Hình 1.8 : Đinh dạng của thông báo ICMP 19

Hình 2.1: Mạng an toàn dùng IPSec 23

Hình 2.2 : Hiện trạng của IPSec 24

Hình 2.3 : Tổng quan tài liệu IPSec 26

Hình 2.4: Header xác thực IPSec 31

Hình 2.5 : Cơ chế Anti-Replay 32

Hình 2.6: Xác thực end-to-end và end-to-intermediate 34

Hình 2.7 : Cấu truc gói IP với tunnel mode AH 34

Hình 2.8 : Cấu trúc gói IP với Transport mode AH 35

Hình 2.9 : Cấu trúc của IPSec ESP..! 35

Hình 2.10 : Xử lý gói tin đi của giao thức ESP 37

Hình 2.11: Xử lý gói tin đến của giao thức ESP 38

Hình 2.12 : An toàn với phương thức transport 38

Hình 2.13 : an toàn với phương thức tunnel 39

Hình 2.14 :Cấu trúc gói IP trong tunnel mode ESP 39

Hình 2.15 : Cấu trúc gói IP trong tranport mode ESP 40

Hình 2.16 : Ví dụ về trao đổi khoá Aggressive Oakley 45

Hình 2.17 :Các dạng ISAKMP 7.7. 46

Hình 3.1 : Mô hình hoạt động của PLUTO 54

Hình 3.2 : Sơ đồ hoạt động của các module 56

Hình 3.3 : Sơ đồ hoạt động của KLIP 59

Hình 3.4 : Mô hình mạng lan-to-lan 67

Hình 3.5 : Khởi tạo một đường hầm mới 76

Hình 3.6 : Kiểm tra kết nối trên Client 1 82

Hình 3.7 : Kiểm tra kết nối trên máy client 2 82

Bảng 2.1: Các dịch vụ IPSec 27

Bảng 2.2 : Chức năng của phương thức Tunnel và transport 31

Bảng 2.3 : Các kiểu trao đổi ISAKMP 50

Bảng 3.1 : Các module chính trong PLUTO và KLIP 55

Bảng 3.2 : Các thông số trong file cấu hình /etc/ipsec.conf 72

|  |  |  |
| --- | --- | --- |
| Ký hiệu | Tên đầy đủ | Giải thích |
| TCP/IP | Transmission Control Protocol/Intemet Protocol | Giao thức điều khiển truyền Thông/Giao thức Internet |
| IPSec | Internet Protocol Security | Giao thức bảo mật Internet |
| VPN | Vitrual Protocol Network | Giao thức mạng riêng ảo |
| OSI | Open System Interconnection | Mô hình kết nối các hệ thống mở |
| ISO | International Standar Organization | Tổ chức Tiêu chuẩn hóa quốc tế: Được thành lập từ năm 1947, có trụ sở đặt tại Geneva - Thụy sĩ. ISO là một hội đoàn toàn cầu của hơn 150 các cơ quan tiêu chuẩn quốc gia (mỗi thành viên của ISO là đại diện cho mỗi quốc gia của mình), Tổng cục Tiêu chuẩn Đo lường Chất lượng là thành viên chính thức của ISO từ năm 1977 |
| NCP | Network Control Protocol | Giao thức điều khiển mạng |
| ICMP | Internet Control Message Protocol | Giao thức điều khiển thông báo Internet |
| IGMP | Internet Group Management Protocol | Giao thức quản lý nhóm Internet : là cơ chế truyền thông giữa trạm con và router gắn trực tiếp vói mạng đó. |
| GGP | Gateway-to-Gateway  Protocol | Giao thức gateway tói gateway |
| EGP | Exterior Gateway Protocol | Giao thức bên ngoài gateway |
| UDP | User Datagram Protocol | là một trong những giao thức cốt lõi của giao thức TCP/IP. Dùng UDP, chương trình trên mạng máy tính có thể gửi những dữ liệu ngắn được gọi là datagram tói máy khác. |
| ESP | Encap Security Payload | Khối an toàn tóm lược cung cấp các dịch vụ bí mật, bao gồm bí mật nội dung thông báo và bí mật luồng traffic |
| AH | Authentication Header | Header xác thực cho phép đảm bảo sự toàn vẹn dữ liệu và xác thực các gói IP |

|  |  |  |
| --- | --- | --- |
| FTP | File Tranfer Protocol | Giao thức truyền file : là một dịch vụ cho phép sao chép file từ một hệ thống máy tính này đến hệ thống máy tính khác. |
| DNS | Domain Name Server | Dịch vụ tên miền |
| SMTP | Simple Mail Transfer Protocol | Dịch vụ chuyển thư điện tử. |
| ISAKMP | Internet Security Association and Key Management Protocol | Hiệp hội bảo mật Internet và giao thức quản lý khóa |
| IKE | Internet Key Exchange | Trao đổi khóa trên Internet |
| LAN | Local Area Network | Mạng cục bộ |
| WAN | Wide Area Network | Mạng diện rộng |
| SPI | Security Parameters Index | SPI là một từ khóa xác dinh được thêm vào phần đầu. |
| MAC | Media Access Control | Là địa chỉ duy nhất để xác định một máy tính(thiết bị) trên mạng internet |
| DOI | Domain of Interpretation | Tên miền của sự giải thích |

Bảng danh mục các ký hiệu, các từ viết tắt

TÓM TẮT ĐỂ TÀI

Đề tài: NGHIÊN cứu TÌM HIỂU CÔNG NGHỆ BẢO MẬT GÓI IP  
(IPSEC) YÀ ÚNG DỤNG TRONG BẢO MẬT THÔNG TIN TRÊN

MẠNG MÁY TÍNH.

Mục tiêu :

* Tìm hiểu về giao thức TCP/IP, công nghệ IPSec, các cơ chế mã hóa dữ liêu trong IPSec. Tìm hiểu về kiến trúc của IPSec, ứng dụng của nó trong thực tiễn - nghiên cứu về bộ phần mềm OpenS/wan.

Yêu cầu :

* Hoàn thành bản báo cáo đầy đủ.
* Sản phẩm : cài đặt và chạy được gói phần mềm OpenS/wan, kết nối được hai subnet thông qua OpenS/wan.
* Thời gian hoàn thành : 05/06/2009.

Nhiệm vụ :

* Tìm hiểu về giao thức TCP/IP.
* Tìm hiểu về công nghệ IPSec.
* Tìm hiểu về gói phần mềm OpenS/wan.
* Thực hiện cài đặt gói phần mềm OpenS/wan trên hai server (cài hệ điều hành linux) và cấu hình sao cho hai mạng subnet ở phía sau hai server có thể kết nối được với nhau.

Kết quả đạt được :

* Hiểu được về bộ giao thức TCP/IP và chức năng của các tầng trong giao thức TCP/IP.
* Hiểu về công nghệ IPSec và tầm quan trọng của nó trong việc truyền tải dữ liệu trên mạng.
* Cài đặt và cấu hình thành công được gói phần mềm OpenS/wan theo mô hình lan-to-lan.

ĐẶT VẤN ĐỂ

•

Từ khi có mạng Internet, nhu cầu truyền và trao đổi thông tin từ noi này đến nơi khác của con người được đáp ứng. Mạng Internet đã giúp con người liên lạc với nhau thuận lợi hơn, họ có thể chia sẻ thông tin vói nhau và sử dụng mạng internet để phục vụ nhiều nhu cầu khác như quảng cáo, truyền dữ liệu... Do mạng internet là một mạng công cộng, mọi thông tin truyền đi đều có thể có nguy cơ bị rò rỉ hoặc mất mát. Con ngưòi đã nghĩ đến việc tạo ra những kênh liên lạc riêng để trao đổi thông tin vói nhau mà không bị mất mát hoặc rò rỉ. Để làm điều đó thì đã có rất nhiều nghiên cứu được thực hiện và đạt được những thành công nhất định. Trên hệ điều hành Windows của Microsoft thì đã có vitrual private network (VPN) đảm nhiệm việc tạo ra những kênh liên lạc riêng sử dụng mạng Internet làm môi trường truyền - gọi là các đường hầm (Tunnel). Nhưng đối với hệ điều hành Linux thì việc tạo ra các kênh liên lạc riêng biệt là một vấn đề khó và mất nhiều thời gian nghiên cứu. Chính vì thế, OpenS/wan ra đời đã giúp giải quyết vấn đề VPN trên hệ điều hành Linux. OpenS/wan đã giúp tạo ra những kênh truyền riêng biệt nhằm bảo về dữ liệu riêng tư của người dùng khi được truyền tải trên mạng công cộng và OpenS/wan đã áp dụng tốt công nghệ IPSec trên hệ điều hành Linux.

Với những lợi ích mà OpenS/wan đem lại, đồ án này được thực hiện vói mục đích nghiên cứu về công nghệ IPSec và chức năng của gói phần mềm OpenS/wan.

Với những yêu cầu tìm hiểu, nghiên cứu ở trên đồ án của em trình bày gồm có ba chương :

Chương I : Tổng quan về giao thức TCP/IP : Chương này giói thiệu tổng quan về bộ giao thức TCP/IP. Kiến trúc và chức năng của các tầng trong giao thức TCP/IP.

Chương II : Tìm hiểu về công nghệ IPSec : Chương này trình bày về công nghệ IPSec, sự trao đổi khóa, các dịch vụ và lọi ích của IPSec đem lại.

Chương III : úng dụng của IPSec trong bảo mật thông tin trên mạng : Chương này giới thiệu về gói phần mềm OpenS/wan và ứng dụng cài đặt nó trên hệ thống mạng.

Hà Nội, ngày ... tháng ... năm 2009  
Sinh viên thực hiện

Thái Trung Thắng

Chương 1: Tổng Quan về Giao Thức TCP/IP.

1. - Mô hình tham chỉếu mở OSI (Open system Interconnection).
2. - Giới thiệu mô hình OSI.

Năm 1984 tổ chức tiêu chuẩn thế giới ISO (International Standar Organization ) đã đưa ra mô hình tham chiếu OSI hay thường gọi là mô hình 7 tầng cố cấu trúc như hình vẽ sau:

|  |  |  |
| --- | --- | --- |
| Application |  | ứna dune |
|  |
| Presentation |  | Trình diễn |
|  |
| Session |  | Phién |
|  |
| Transport |  | Giao vận |
|  |
| Network |  | Mạng |
|  |
| Data Link |  | Liên kết DL |
|  |
| Physical |  | Vặt lý |
|  |

Đườne truyẻn vật lý

Hình 1.1 : Mô hình tham chiếu OSỈ

1. Cắc nguyên tắc khi xây dựng mô hình tham chiếu OSI.

+ Để đơn giản cần hạn chế số lượng các tầng .

+ Mỗi lớp cần thực hiện các chức năng được định nghĩa rõ ràng.

+ Việc chọn chức năng cho mỗi lớp cần chú ý tới việc định nghĩa các quy tắc chuẩn hoá quốc tế.

+ SỐ mức phải đủ lán để các chức năng tách biệt không nằm trong cùng một lớp và đủ nhỏ để mô hình không quá phức tạp.

+ Một lớp có thể được phân thành các lớp nhỏ nếu cần thiết.

+ Các lớp con có thể lại bị loại bỏ khi không cần thiết.

+ Hai hệ thống khác nhau có thể truyền thông với nhau nếu chứng bảo đảm những nguyên tắc chung (cài đặt cùng một giao thức truyền thông).

+ Các chức năng được tổ chức thành một tập các tầng đồng mức cung cấp chức năng như nhau. Các tầng đồng mức phải sử dụng một giao thức chung.

1. - Chức năng của cấc tầng trong mồ hình tham chiếu OSỈ.

a- Tầng vật lý :

Là tầng thấp nhất, có chức năng truyền dòng bit không có cấu trúc qua đường truyền vật lý,truy cập đường truyền vật lý nhờ các phương tiện cơ, điện, hàm, thủ tục V V ..

Lớp vật lý làm việc với các thiết bị vật lý, truyền tói dòng bít 0 và 1, từ nơi nhận đến nơi gửi.

b - Tầng liên kết dữ liệu :

Cung cấp các phương tiện truyền thông tin qua liên kết vật lý đảm bảo tin cậy, gửi các khối dữ liêu với các cơ chế đổng bộ hoá, kiểm soát lỗi, kiểm soát luồng dữ liệu cần thiết.

Tại đây dữ liệu được tổ chức thành các khung (frame). Phần đầu khung chứa địa chỉ và thông tin điều khiển, phần cuối khung dành cho việc phát hiện lỗi.

c - Tầng mạng :

Thực hiện việc chọn đường và chuyển tiếp thông tin với các công nghệ chuyển mạch thích hợp, thực hiện việc kiểm soát luồng dữ liệu và cắt hợp thông tin khi cần thiết.

d - Tầng giao vận :

Thực hiện việc truyền dữ liệu giữa hai đầu mút, kiểm soát lỗi, kiểm soát luồng dữ liệu giữa hai đầu mút, thực hiện việc ghép kênh, cắt hợp dữ liệu khi cần thiết.

e - Tầng phiên :

Cung cấp phương tiện quản lý truyền thông giữa các ứng dụng, thiết lập duy trì, đồng bộ hoá và huỷ bỏ các phiên truyền thông giữa các ứng dụng.

/ - Tầng trình diễn :

Chuyển đổi cấu trúc dữ liệu để đáp ứng nhu cầu truyền dứ liệu của các phương tiện truyền thông trên môi trường OSI.

g - Tầng ứng dụng:

Cung cấp các phương tiện để người dùng có thể truy cập vào môi trường OSI, đồng thời cung cấp các dịch vụ thông tin phân tán.

1. - Bộ giao thức TCP/IP.
2. - Giới thiệu về TCP IIP.

Họ giao thức TCP/IP được phát triển từ những năm 1970 bởi hai tác giả Vint Cerf và Robert Kahn, ban đầu cùng tổn tại vói giao thức NCP (Network Control Protocol) nhưng tới năm 1983 thì TCP/Ip đã thay thế hoàn toàn giao thức NCP.

• Có một số ưu điểm của TCP/IP như sau:

* Giao thức chuẩn mở thoải máỉ và sẵn sàng phát triển độc lập với phần cứng và hệ điều hành. Bởi vì nó được hỗ trợ bởi nhiều nhà cung cấp.
* TCP/IP lý tưởng cho việc hợp nhất phần cứng và phần mềm khác nhau, ngay cả khi truyền thông trên Internet. Sự độc lập rành mạch với phần cứng vật lý của mạng cho phểp TCP/EP hợp nhất các mạng khác nhau.
* TCP/IP có thể chạy trên mạng Ethernet, mạng Token ring, mạng quay số (Dial-Up line), mạng X.25 mạng ảo và mọi loại môi trường vật lý truyền thông.
* Một sơ đồ địa chỉ dùng chung cho phép mỗi thiết bị TCP/IP có duy nhất một địa chỉ trên mạng ngay cả khỉ đó là mạng toàn cầu Internet.
* HỖ trợ mô hình client-server, mô hình mạng bình đẳng, Hỗ trợ kỹ thuật dẫn đường động.

1. Kiến trúc của TCP HP.

Kiến trúc phân tầng của TCP/IP cũng tuân theo nguyên tắc phân tầng của mô hình tham chiếu OSI. TCP/IP được phân ra thành 4 tầng tương ứng với mô hình OSI như sau:

* Tầng mạng (Network Layer)
* Tầng Internet (Internet Layer)
* Tầng giao vận (Transport Layer)
* Tầng ứng dụng (Application Layer)

Sơ đồ tương ứng :

Mô hình OSI Mô hình TCP/IP

|  |  |  |
| --- | --- | --- |
| Application |  |  |
| Presentation |  | Application |
| Session |  |  |
| Transport |  | Transport |
| Network |  | Internet |
| Data Link |  | Network |
| Physical |  |  |

Hình 1.2 : Mồ hình TCP IIP • Sự hoạt động của TCP Ị IP:

Cũng giống như ttong mô hình tham chiếu OSI, dữ liệu gửi từ tầng Application đi xuống ngăn xếp, mỗi tầng có những định nghĩa riêng về dữ liệu mà nó sử dụng. Tại nơi gửi, mỗi tầng coi gói tin của tầng trên gửi xuống là dữ liệu của nó và thêm vào gối tin các thông tín điều khiển của mình sau đó chuyển tiếp xuống tầng dưới. Tại nơi nhận, quá trình diễn ra ngược lại mỗi tầng lại tách thông tín điều khiển của mình ra và chuyển dữ liệu lên tầng ưên. Sơ đồ sau miêu tả rõ điều đó:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| APPLICATION | |  |  | DATA |
|  | |  |  | Q |
| TRANSPORT | |  | TCP he ader | DATA |
|  | |  |  |  |
| INTERNET | | IP header | TCP header | DATA |
|  | |  |  | £ |
| NETWORK | Ethernet header | IP header | TCP header | DATA |

Hình 13 : Mô hình hoạt động của TCP IIP

1. — Giao thức liên mạng TCP (Tranmission Control Protocal).

a - Khái niệm.

Một kết nối TCP sẽ được thực hiện khi ứng dụng ở một host truyền và nhận dữ liệu đến một host khác. TCP cung cấp khả năng truyền song cụng (full-duplex) giữa hai ứng dụng ở hai đầu kết nối.

TCP phải có nhiệm vụ chuyển dữ liệu của lớp ứng dụng thành các đơn vị dữ liệu có thể truyền để có thể đóng gói thành packet ở lớp Internet, ứng dụng chuyển dữ liệu đến TCP và TCP đặt vào bộ đệm gửi. TCP chia nhỏ dữ liệu và thêm phần tiêu đề (header) tạo thành đơn vị dữ liệu gọi là segment Kích thước của segment phải luôn được diều chỉnh ở mức tối ưu với tài nguyên hiện có ưên mạng. TCP sẽ chờ cho đến khi nhận đủ dữ liệu từ lớp trên trước khi tạo một segment có kích thước phù hợp.

Một máy khách phải xác định được loại dịch vụ yêu cầu từ máy chủ. Điều này được thực hiện bằng việc sử dụng cặp địa chỉ IP và số hiệu cổng TCP. Cổng TCP nẩm ưòng khoảng từ 0 đến 65535. Từ 0 đến 1023 là cac cổng cho những dịch vụ thông thường.

Sợ kết hợp giữa địa chỉ IP và số hiệu cổng tạo thành cặp địa chỉ socket. Một kết nối TCP giữa hai đầu cuối được nhận diện hay phân biệt nhờ địa chỉ socket này. Trong header của packet chứa thông tin địa chỉ nguồn và địa chỉ đích, số hiệu cổng nằm trong segment cùa TCP.

TCP là một giao thức connection-Oriented nên để truyền được dữ liệu thì trước đó nó phải thiết lập kết nối rồi duy trì kết nối và sau khi hết dữ liệu cần gửi, nó phải giải phóng kết nối. Trong quá trình truyền dữ liệu có sử dụng cơ chế điều khiển luồng (flow control) và diều khiển lỗi.

b - Định dạng dữ liệu của TCP.

Mỗi segment cùa giao thức TCP bao gồm phần tiêu đề (header) và phần dữ liệu (data).

0 12 3

01234667890123456789012345678901

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Source Port | | | | | | | | Destination Port |
| Sequence Mum ber | | | | | | | | |
| Ac k nowl edgment N urn her | | | | | | | | |
| Data  Offset | Reset | u  R  G | A  c  K | p  &  H | R  s  T | 1 1 | F  1  N | Window |
| Checksum | | | | | | | | Urgent Pointer |
| Options | |  |  |  |  |  |  | ..L Padding |
|  |  |  |  | Data Bytes | | | | |

Hình 1.4 : Định dạng segment - TCP

Trong đó :

* Source port (16 bit) và Destination port (16 bit) : số hiệu cổng của host nguồn và đích.
* Sequence Number (32 bit) : Số hiệu xác định vị trí byte đầu tiên của segment khi bit SYN không được thiết lập. Nừu bit SYN được thiết lập thì đây là số hiệu tuần tự khỏi đầu của dữ liệu.
* Acknowledgment Number (32 bit): ký hiệu là ACK, là số hiệu của segment kế tiếp trong dòng dữ liệu mà bên nhận đang chở. Data offset (4 bit) : chỉ kích thước của phần header TCP tính theo đơn vị từ 32 bit. Trường này đồng thời cũng xác đinh vị trí bắt đầu của phần dữ liệu.
* Reserved (6 bit) : Trường này hiện vẫn dự phòng và luôn bằng 0.
* Flags (6 bit) : là các bít cờ có ý nghĩa như sau :

s URG : Bằng 1 nếu có dữ liệu khẩn. Dữ liệu khẩn sẽ được chỉ ra ừong trường Urgent Pointer. Ngược lại thì bằng 0.

'S ACK : Bằng 0 nếu là segment khởi đầu và khi đó trường ACK number mái có hiệu lực. s PSH : Thông báo dữ liệu cần chuyển đi ngay. s RST : Xác định lỗi, đồng thời để khcâ động lại kết nối. s SYN : Bằng 1 khi thiết lập kết nối.

'S FIN : Bằng 1 khi trạm nguồn hết thông tin.

* Window (16 bit) : Đây là số lượng các byte dữ liệu, bắt đầu từ byte được chi ra trong trường ACK Number mà trạm nguồn sẵn sàng để nhận.
* Checksum (16 bit) : Mã kiểm soát lỗi theo phương pháp CRC của toàn bộ Segment.
* Urgent Pointer (16 bit) : Đây là con trỏ tói số hiệu tuần tự của byte đi sau dữ liệu khẩn, cho phép bên nhận biết được độ dài của dữ liệu khẩn. Trường hợp này có hiệu lực khi bit URG được thiết lập 1.
* Padding (độ dài thay đổi) : Phần mềm chèn thêm vào header để đảm bảo header luôn kết thúc ở một mốc 32 bit. Phần chèn thêm này luôn bằng 0.
* Data (độ dài không xác định) : Chứa dữ liệu cần gửi đi của lớp trên TCP.
* Options (độ dài thay đổi) : Khai báo các tùy chọn cuat TCP, trong đó có độ dài tối đa của vùng TCP data trong một segment.

1. - Giao thức truyền thông IP (Internet Protocol).

Giao thức IP là một giao thức lóp mạng, được sử dụng phổ biến cho các mạng tham gia internet. Thực chất, internet là mạng của các mạng nối vói nhau qua bộ định tuyến (Router). Mục đích ra đời của IP là để thống nhất việc sử dụng các máy chủ và router từ các hãng sản xuất khác nhau. Cho nên, IP cho phép kết nối nhiều loại mạng có đặc điểm khác nhau mà không gián đoạn hoạt động của mạng và kết nối với internet.

Giao thức IP có ba nhiệm vụ chính đó là :

* Thứ nhất : Giao thức IP định nghĩa đơn vị cơ sở của lóp Internet.
* Thứ hai : Thực hiện chức năng định tuyến (Routing), chọn ra con đường đi tối ưu mà dữ liệu cần gửi qua.
* Thứ ba : Điều khiển và xử lý lỗi.

a - Định dạng IP.

Trên một mạng vật lý, đơn vị truyền dữ liệu là một frame bao gồm phần đầu và phần dữ liệu, với phần đầu cung cấp địa chỉ nguồn và địa chỉ đích (vật lý). Internet gọi đơn vị truyền dữ liệu của nó là IP datagram hoặc datagram (có những tài liệu thì lại gọi là packet). Cũng giống như một frame trong mạng vật lý, một datagram bao gồm 2 phần :

'S Phần tiêu đề (header)

'S Phần dữ liệu (data)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| header | data |  | |  |
| base IP da | tagrarn... |
| physical network header | | | IP datagram as data |
| encapsulated within the physical network's frame  3376'33 7ỡF 2ŨG | | | | |

Hình 1.5 : Định dạng của một IP datagram

Sau đây chứng ta sẽ tìm hiểu chỉ tiết nội dung từng trường một trong header của IP datagram.

112 3

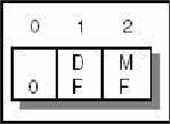
0 4 6 6 9 4 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| VERS | HLEN | Service  Type | Total Length | | |
| ED | | | F LG | Fragment  OEfirét | |
| TIL | | Protocol | Header  Checksum | | |
| Sowoe IP Address | | | | | |
| De-acinattoil IP Address | | | | | |
| IP Options | | | | | Peddìng |
|  |  | Data . |  |  |  |
|  |  |  |  |  |  |

Hình 1.6 : Chi tiết header của IP packet.

Trong đó:

* VERS (4 bit) : Chỉ phiên bản hiện hành của IP được sử dụng. Với IP thông thường là 4, thế hệ IP tiếp theo là 6.
* HLEN (4 bit) : Chỉ độ dài phần tiêu đề của datagram tính theo đơn vị từ (32 bit). Độ dài tối thiểu là 5 (20 octet).
* Service Type : Là chỉ số chất lượng dịch vụ yêu cầu cho IP datagram.
* Total Length : Xác định độ dài của toàn bộ datagram header và data.
* Identification : Cùng với các tham số khác như Source IP address, Destination IP address dùng để định danh duy nhất cho một datagram trong khoảng thời gian gói tin tồn tại trên mạng, dùng để tập hợp fragmented datagram.
* Flags : Liên quan đến sự phân đoạn của datagram. Trong đó :



Hình 1.7: Định dạng Flags S 0 : chưa sử dụng và luôn bằng 0.

s DF (do not Fragment) : bằng 0 có nghĩa là phép phân mảnh, bằng 1 là không cho phép phân mảnh.

s MF (More Fragments) : bằng 0 đây là đoạn phân mảnh cuối cùng (the last fragment), bằng 1 thì đây là đoạn phân mảnh tiếp theo (more fragments).

* Fragment Offset (13 bit) : Chỉ vị trí của đoạn (fragment) ưong datagram ban đầu, tính theo đơn vị 8 octet. Mỗi đoạn (trừ đoạn cuối cùng) phải chứa vùng dữ liệu là bội số của 8 octet.
* Time to Live (8 bit): Quy định thời gian tồn tại (tính bằng giây) của datagram ưên mạng để ttánh tình trạng datagram không đến được đích và cứ đi lòng vòng trên mạng. Thời gian này được thiết lập bởi trạm gửi và giảm đi 1 mỗi khỉ datagram đi qua một nứt mạng. TTL=Q gói dữ liệu sẽ bị discard.
* Protocol Number : nó báo mức cao hơn của giao thức IP có thể chuyển ừong gối này. Bao gồm :

S 0 : Dự trữ

s 1 ; Internet Control Message Protocol (ICMP) s 2 : Internet Group Management Protocol ỢGMP) s 3 : Gateway-to-Gateway Protocol (GGP) s 4 : IP (IP encapsulation) v' 5 : Dòng (Stream)

s 6 : Transmission Control Protocol (TCP) s 8 : Exterior Gateway Protocol (EGP)

J 9 : Private Interior Routing Protocol s 17 : User Datagram Protocol (UDP)

J 41 : IP Version 6 (Ipv6) s 50 : Encap Security Payload for Ipv6 (ESP) s 51 : Authentication Header for Ipv6 (AH) s 89 : Open Shortest Path First

* Source IP address (32 bit): Địa chỉ IP của trạm gửi.
* Destination IP Address (32 bit): Địa chỉ IP của trạm nhận
* Header Checksum (16 bit) : Mã kiểm soát lỗi 16 bit theo phương pháp CRC, chi áp dụng cho vùng header. Trường này luôn được cập nhật khi một gói tin đi qua router trung gian.
* Options : Khai báo các tùy chọn do noi gửi yêu cầu. Trường option không bắt buộc phải có trong mọi datagram và chủ yếu dùng để kiểm tea lỗi trên mạng. Option là một phần quan trọng của giao thức IP nên mọi tiêu chuẩn thực hiện phải dựa ữên IP phải bao gồm tiến trình xử lý trường này. Độ dài của trường Option thay đổi tùy thuộc vào các tham số đi kèm. Khi các Option xuất hiện trong datagram, nó sẽ kéo dài liên tục mà không có sự ngắt quãng.

b - Định tuyến.

Một trong những chức năng của giao thức IP có khả năng thiết lập kết nối giữa các mạng vật lý khác nhau. Đó chính là đinh tuyến. Một hệ thống thực hiện chức năng này được gọi là IP router.

c - Điều khiển và xử lý lỗi.

Như đã biết, giao thức IP cung cấp dịch vụ “unreliable”, chuyển dữ liệu connectionless bằng cách dàn xếp cho mỗi bộ đinh tuyến chuyển dữ liệu. Mỗi packet sẽ di chuyển từ bộ định tuyến này đến bộ đinh tuyến khác cho đến bộ định tuyến mà có thể chuyển packet trực tiếp đến đích cuối cùng của nó. Nếu một bộ định tuyến không thể gửi một packet, hay nếu nó phát hiện một dấu hiệu không bình thường có ảnh hưởng đến việc truyền dữ liệu ( ví dụ : nghẽn mạch trên mạng),bộ định tuyến cần phải thông báo cho noi xuất phát packet, để tránh hoặc khắc phục lỗi. Do đó cần phải có một cơ chế để thông báo lỗi cho bên gửi gói tin.

Giao thức thông báo điều khiển liên mạng ICMP (Internet Control Message Protocol) ra đời để giải quyết vấn đề trên. ICMP cũng giúp cho các host định tuyến trên mạng và cho phép các nhà quản lý mạng theo dõi tình trạng các node trên mạng. Tất cả các host và router đều phải có khả năng tạo và xử lý các thông báo ICMP nhận được.

1. - Giao thức thông báo điều khiển liên mạng (ICMP).

Giao thức IP hoạt động tại lớp Network được sử dụng bởi IP cho nhiều dịch vụ khác nhau. Thông báo ICMP được mang trực tiếp trong gói tin IP vói trường Protocol Number bằng 1.

Có rất nhiều trường hợp khiến cho gói tin IP bị loại bỏ: Đường truyền có sự cố, trường Time-to-Live hết hạn, không phân mảnh được gói tin kích thước lớn hơn MTU cho phép...Khi một gói tin cần loại bỏ, thông báo ICMP được sử dụng để thông báo về địa chỉ gửi gói tin.

Tuy nhiên, không phải trường hợp nào ICMP cũng cần phải báo lỗi. Sau đây là một số trường hợp mà khi xảy ra sự cố, ICMP không cần báo lỗi:

* Định tuyến hay chuyển giao thông báo ICMP.
* Phát quảng bá hay phát theo nhốm gói tin IP.
* Các phân đoạn gối tín khác vứi phân đoạn đầu tiên.
* Thông báo cố địa chỉ nguồn không xác định một host duy nhất (ví dụ: 127.0.0.1, ¿0.0.0)/

Định dạng của thông báo IGMP như sau :

0 ô 16 31

type

code

checksum

IGMP data ; depending on the type ol mess

Hình 1.8 : Định dạng của thông báo ICMP

Thông báo ICMP được mang ừong phần dữ liệu của gói tin IP. Mặc dù mỗi bản tin ICMP cố dạng riêng của nố, nhưng chứng đềiu bắt đầu với ba trường sau:

s TYPE (8 bit): là một số nguyên 8 bit để xác định thông điệp, v' CODE (8 bit): cung cấp thêm thông tin về kiểu thông điệp, v' CHECKSUM (16 bit) : ICMP sử dụng thuật toán giải checksum như IP, nhung ICMP checksum chỉ tính đến thông điệp ICMP.

Hơn nữa, các thông đỉệp ICMP thông báo lỗi luôn luôn bao gồm phần đầu và 64 bit đầu tiên của packet gây nên lỗi. Lý do có thêm phần đầu này cùng với phần đầu packet là để cho phép nơi nhận xác định chính xác hơn những giao thức nào và chương trình ứng dụng có trách nhiệm đối với packet.

Trường TYPE của ICMP xác đinh ý nghĩa của thông điệp cũng như định dạng của nố. Các kiểu bao gồm : s 0 : Echo reply s 3 : Destination unreachable s 4 : Source quench s 5 : Redirect J 8 : Echo

s 9 : Router advertisement s 10 : Router solicitation s 11: Time exceeded s 12 : Parameter problem s 13 : Timestamp request s 14 : Timestamp reply s 15 : Information request (obsolete) s 16 : Information reply (obsolete)

'S 17 : Address mask request 'S 18 : Address mask reply 'S 30 : Traceroute 'S 31 : Datagram conversion error 'S 32 : Mobile host redữect 'S 33 : Ipv6 1s 34 : Ipv6

'S 35 : Mobile registeration request 'S 36 : Mobile registeration reply 'S 37 : Domain name request 'S 38 : Domain name reply 'S 39 : SKIP s 40 : Photuris

1. — Một số ứng dụng của TCPUP.

a - FTP (File Tranfer Protocol).

FTP là một dịch vụ cho phép sao chép file từ một hệ thống máy tính này đến hệ thống máy tính khác ftp bao gồm thủ tục và chương trình ứng dụng, và là một trong những dịch vụ ra đời sớm nhất trên Internet.

FTP có thể được dùng ở mức hệ thống, trong Web browser hay một số tiện ích khác. Fpt vô cùng hữu ích cho những người dùng Internet, bởi vì khi sục sạo trên Internet, bạn sẽ tìm thấy vô số những thư viện phần mềm có ích về rất nhiều lĩnh vực và bạn có thể chép chúng về để sử dụng.

b — Dịch vụ tên miền DNS (Domain Name Server).

Việc định vị các máy tính trên mạng bằng các địa chỉ IP có nhiều lọi điểm như đã trình bày ở phần trên, tuy nhiên với người sử dụng, việc nhớ các con số đó là một việc tẻ nhạt không thể chịu nổi. Hơn nữa, địa chỉ IP không mang thông tin về địa lý, tổ chức hay người dùng. Vì thế, người ta xây dựng hệ thống đặt tên gọi là Domain Name Server để cung cấp cho người dùng cách đặt tên cho các máy tính với cách đặt tên thông thường quen thuộc.

Một domainame thông thường có dạng:

Tên\_người\_dùng@Tên\_miền

Với tên miền được phân làm các cấp nối vói nhau bởi dấu Tên miền được NIC cung cấp. Tên miền cao nhất là cấp quốc gia được đặt bỏi 2 chữ cái:

Việt nam là VN , Pháp là FR .... nếu không có gì thì được hiểu như thuộc USA Mức tiếp theo chỉ lãnh vực hoạt động: edu: giáo dục , gov: chính phủ , com: thương mại, mil: quân sự .

Sau đó có thể là tên công ty và tên máy tính. Một máy tính có thể có nhiều tên nhưng trên mạng, mỗi tên là duy nhất. Việc ánh xạ địa chỉ IP vào tên miền được thực hiện bởi các Name Server cài đặt tại máy Server và Name Resolver cài đặt trên máy trạm.

c - Thư điện tử (Electronic Mail).

Đây là một dịch vụ phổ biến nhất trên Internet Bằng dịch vụ này, mọi người sử dụng máy tính kết nối với Internet đều có thể trao đổi thông tin vói nhau.

Dịch vụ này sử dụng giao thức SMTP (Simple Mail Transfer Protocol) trong họ giao thức TCP/IP. Một điểm manh của thư điện tử là nó là phưng thức trao đổi thông tin nhanh chóng và thuận tiện.

Người sử dụng có thể trao đổi những tin ngắn hay dài chỉ bằng một phưng thức duy nhất. Rất nhiều người sử dụng thường truyền tập tin thông qua thư điện tử chứ không phi bằng các chưng trình truyền tập tin thông thường. Đặc điểm của dịch vụ thư điện tử là không tức thời (off-line) - tất c các yêu cầu gửi đi không đòi hỏi phi được xử lý ngay lập tức. Khi ngưòi sử dụng gửi một bức thư, hệ thống sẽ chuyển thư này vào một vùng riêng (gọi là spool) cùng với các thông tin về người gửi, ngưòi nhận, địa chỉ máy nhận... Hệ thống sẽ chuyển thư đi bằng một chưng trình không đồng bộ (background). Chương trình gửi thư này sẽ xác định địa chỉ IP máy cần gửi tói, tạo một liên kết vói máy đó. Nếu liên kết thành công, chưng trình gửi thư sẽ chuyển thư tói vùng spool của máy nhận. Nếu không thể kết nối với máy nhận thì chưng trình gửi thư sẽ ghi lại những thư chưa được chuyển và sau đó sẽ thử gửi lại một lần nó hoạt động. Khi chương trình gửi thư thấy một thư không gửi được sau một thời gian quá lâu (ví dụ 3 ngày) thì nó sẽ tr lại bức thư này cho người gửi. Mọi thư trên Internet đều tuân theo một dạng chuẩn. Bao gồm phần header chứa địa chỉ người gửi, địa chỉ người nhận dạng domain name và sau đó là phần nội dung thư. Có hai phần đều là các ký tự ASCII chuẩn. Thư chuyển trên mạng và đến được đích là nhờ vào thông tin chứa trong phần header của thư. Ban đầu thư điện tử chỉ nhằm mục đích trao đổi các thông báo (thực chất là các tệp văn bản) giữa người sử dụng với nhau. Dần dần người ta đã phát triển thêm các biến thể trên nó để phục vụ người sử dụng tốt hơn hoặc dùng cho những mục đích riêng biệt. Đó là các dịch vụ thông tin dựa trên thư điện tử. Thực chất của các dịch vụ này là sử dụng thư có nội dung tuân theo một cú pháp đặc biệt thể hiện yêu cầu của người sử dụng. Các thư này được gửi tói một ngưòi sử dụng đặc biệt là các server, các server này phân tích nội dung thư, thực hiện các yêu cầu rồi gửi tr lại kết qu cho người yêu cầu cũng dưới dạng thư điện tử. Có hai server phổ biến trong hoạt động này là ã name server cung cấp dịch vụ fra cứu địa chỉ trên mạng ã archive server cho phép người sử dụng tìm kiếm và lấy về những tệp tin dùng chung.

d - Dịch vụ WEB.

WEB là dịch vụ Internet ra đời gần đây nhất, nhưng phát triển nhanh nhất hiện nay. Nó cung cấp một giao diện vô cùng thân thiện vói người dùng, dễ sử dụng, vô cùng thuận lợi và đơn giản để tìm kiếm thông tin. Web liên kết thông tin dựa trên công nghê hyper-link (siêu liên kết), cho phép các trang Web liên kết với nhau trực tiếp qua các địa chỉ của chúng.

Chương 2 : Tìm hiểu về công nghệ IPSec

1. - IPSec (Internet protocal security).
2. - Tổng quan về IPSec.

Năm 1994, uỷ ban kiến trúc Internet (Internet Architecture Board- IAB) giới thiệu một bản báo cáo với tên gọi “An toàn trong kiến trúc Internet” (RFC 1636). Thông báo được bắt đầu với sự lưu ý rằng Internet cần có sự an toàn nhiều hơn và tốt hơn. Cần phải đảm bảo an toàn cho các dữ liệu được truyền tíên mạng và các thông tin về các luồng traffic bằng cơ chế mã hoá và xác thực. Những mối quan tâm này đã được chứng minh là đúng.

Năm 1997, trong báo cáo hàng năm, “Đội đáp ứng tình trạng khẩn cấp máy tính CERT” (Computer Emergency Response Team) liệt kê trên 2500 vụ việc an toàn làm ảnh hưỏng đến 150,000 sites. Các kiểu tấn công nghiêm trọng nhất bao gồm giả địa chỉ IP (IP spoofing), trong đó kẻ lạ mặt tạo ra các gói với địa chỉ IP sai và khai thác ứng dụng dựa trên IP và một vài dạng nghe trộm và soi gói, trong đó kẻ tấn công đọc các thông tin được truyền bao gồm thông tin logon và nội dung cơ sở dữ liệu.

Để đối phó lại những vấn đề trên, IAB đã gắn các đặc trưng bí mật và xác thực vào cấu trúc gói IP.

1. - Các ứng dụng của IPSec.

IPSec cung cấp khả năng truyền thông an toàn qua một mạng LAN, mạng WAN và Internet.

Các ví dụ của việc dựng IPSec như:

* An toàn giữa các chi nhánh cơ quan khi kết nối với nhau qua internet:

Một công ty có thể xây dựng một mạng riêng ảo an toàn qua Internet hoặc

qua mạng WAN chung. Điều này cho phép việc truyền thông dựa trên Internet và giảm tiết kiệm chi phí xây dựng và quản lý mạng.

* An toàn truy cập từ xa qua Internet:

Một người dùng đầu cuối mà hệ thống của họ được trang bị các giao thức IP có thể truy nhập an toàn tới mạng công ty thông qua các nhà cung cấp dịch vụ Internet

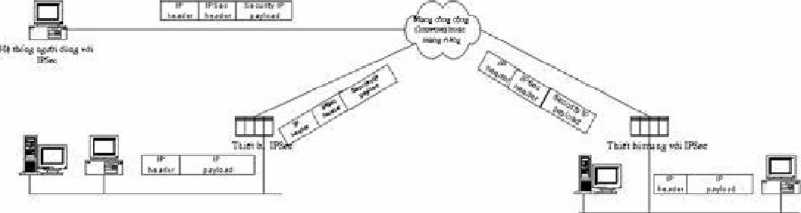
* Thiết lập các kết nối với các đối tác:

IPSec có thể được dùng để truyền thông an toàn vói các tổ chức khác, vói các dịch vụ bí mật, xác thực.

* Cho phép an toàn thương mại điện tử:

IPSec có khả năng hoàn thiện các dịch vụ an toàn trong các trang WEB và các ứng dụng thương mại điện tử.

Đặc trưng chính của IPSec là cho phép nó hỗ trợ các ứng dụng khác nhau để mã và xác thực tất cả các traffic tại mức IP. Như vậy tất cả các ứng dụng phân tán bao gồm logon từ xa, client/server, e-mail, truyền tệp, truy nhập WEB,... có thể an toàn.



Hình 2.1: Mạng an toàn dàng ỈPSec

Hình 2.4 là hình ảnh của mạng dùng IPSec. Một tổ chức quản lý các LAN tại các vị trí phân tán. Traffic IP không an toàn được quản lý tiên mỗi LAN. '

Các traffic đi qua một vài mạng WAN công cộng được bảo vệ bởi các giao thức IPSec. Các giao thức này thao tác ừong các thiết bị mạng như router, firewall, chúng kết nối các LAN với các mạng bên ngoài.Thiết bị mạng IPSec sẽ mã và nén tất cả các ttaffic trước khỉ rời các mạng LAN để đỉ vào mạng WAN và giải mã, giải nén các traffic đi vào từ mạng WAN. Các thao tác này là ừong suốt với các trạm và các server trên mạng LAN. Truyền thông an toàn cũng có thể thực hiện với những người dùng riêng lẻ kết nối vào WAN. Các ừạm của các người dùng như vậy phải cài đặt các giao thức IPSec để cung cấp các dịch vụ an toàn.

1. - Mục đích của ỈPSec.

Được dùng để bảo mật dữ liệu cho các chuyển giao thông tin qua mạng. Admin có thể xác lập một hoặc nhiều chuỗi các Rules, gọi là IPSEC Policy, những rules này chứa các Filters, có trách nhiệm xác định những loại thông tín lưu chuyên nào yêu cầu được mã hốa (Encryption), xác nhân (digital signing), hoặc cả hai. Sau đó, mỗi Packet, được Computer gửi đi, sẽ được xem xét cố hay không gặp các điều kiện của chính sách. Neu gặp những điều kiện này, thì các Packet có thể được mã hốa, được xác nhận số, theo những quy định từ Policy. Quy trình này hòan toàn vô hình với User và Application.

Kích hoạt truyền thông tín trên Mạng.

Do IPSEC được chứa bên ưong mễỉ gối IP chuẩn, cho nên có thể dùng IPSEC qua Network, mà không yêu cầu những cấu hình đặc biệt trên thiết bị hoặc giữa 2 Computer.

Tuy nhiên, IPSEC không tiến hành mã hốa một vài loại giao tiếp mạng như: Broadcast, Multicast, các packet dừng giao thức xác thực Kerberos.

1. - Hiện trạng của ỈPSec.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| RFC 2405 I |  | \  RFC24K |  | DSEÏliBiHI  Prwscîl |  | SCEf/E ' |
| Internet Secut)1 Ajïïsladtfi ard iffy ManjPÏAl | ftinffeEïttiKiiik  \*or ueiçfMiûr ard | Interet Key Emdiinjc  IIKÊj | 1 toi. paris nf | Me K. Omar  tarin |  | Attisât»! Saute Kif Mtanje MMsm-îr |
| PiàocN(BWHPi | tey estera |  | (but là rot |  | Internet |
| Dan Kỉ [Km | MpQdaitOR) |  |  |
| MỉiạpSíl.eỉ.ắ  Nwtsss |  | ûMeCanel N?/19» | neœpfCMS | | | Hugo irawcsyK fit» 1995 |

stDỊMẼm ETES IKEnSAKMP «n tiiespsct b Pìiase 2

RFCHI2

(MLEVKey

C24ID Ỉ-\*.

yULLEKrypUon PJwllTfrâiiü tì IJ65 \*nr -P53C

E'PDEjvEC

Ç^rAJgtrilim

MKBfMiV

cnayiMatoi

N. CÓt3S\*3tny

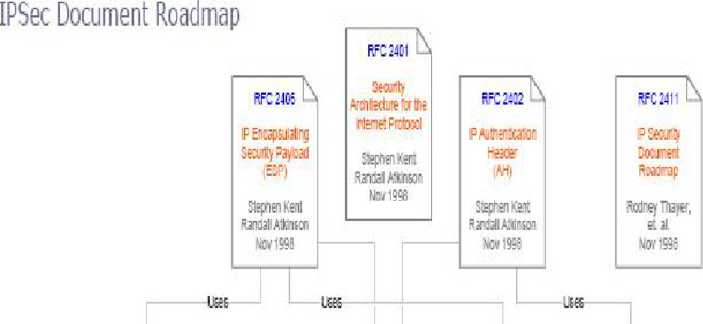
NSV m

RF

Interet p SecurtỊ  
Doutant lf  
Intepœlaio\*  
|Sffl|1tĩlSAKMP

DỉrĩỄí Ptpsỉ Nw19»

Hình 2.2 : Hiện trạng của ỈPSec

**

Celas ỉomed

ttìỉ VÍ.UỈ5

# iTN

RFC2451

ESP COõde SlỊterAlỊptnĩ

ftoÿPaea Roi Adams M199S

fîcdGieiïi

StepreriKBT!

HDV1993

ÏFC24D7 \

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| -¿3  §  ir; |  |  |  | RFC21C4 \\ |
| US-tfMAC- |  | ÜHüfHIftfr |  | HWAjC: Hiajfed- |
| UD5-K WfltUR |  | SMPSfdHn |  | Kaîltgfcr |
| ESP and JW |  | ESP and JW |  | MỈSHQỈ |
|  |  |  |  | AJiMcaKn |
| ctnerçi Ittfeffl |  | Glwjl Maison |  |  |
| RofcGIsnn |  | Rob Gtan |  | tCnwE^eLU |
| NM IBM |  | IDF1SK |  | F6M997 |

- AJtr enD catoi AigomnTỄ--

IPsec là một phần bắt buộc của IPv6 , nhưng đối với IPv4 có thể được lựa chọn. Các giao thức IPsec được định nghĩa từ RFC 1825-1829 vào những năm 1995 , Năm 1998 được nâng cấp lên với các phiên bản RFC 2401-2412 , tháng 12 năm 2005 thế hệ thứ 3 ra đời

1. - Lợi ích của ỈPSec.

- Khi IPSec được cài đặt trong Fkewall hoặc router, nó cung cấp khả năng an toàn cao tới tất cả các ttaffic qua bỉẽn. Các traffic trong các mạng LAN không phải cần phải can thiệp các thao tác xử lý an toàn.

* IPSec nằm dưới mức vận tải (TCP hoặc UDP) nên trong suốt với các ứng dụng. Không cần phải thay đổi phần mềm trên hệ thống người dùng hoặc hệ thống server khi IPSec được cài đặt ừong firewall hoặc router. Thậm chí nếu IPSec được cài đặt trong các hê thống đầu cuối, phần mềm mức ừên bao gồm các ứng dụng không bị tác động
* IPSec có thể trong suốt với người dùng đầu cuối. Không cần phải huấn luyện người dùng về cơ chế an toàn, cũng như các tài liệu khoá cho người dùng hoặc huỷ các tài liệu khoá khi ngưòi dùng ròi cơ quan.
* IPSec có thể cung cấp an toàn cho các người dùng đơn lẻ khi cần thiết. Nó có lợi cho những người làm việc xa cơ quan hoặc thiết lập một mạng con ảo an toàn trong một cơ quan cho các ứng dụng nhạy cảm.

1. - Các ứng dụng định tuyến..

Để hỗ trợ người dùng và bảo vệ các hệ thống biên và mạng biên, IPSec có thể đóng vai trò quan trọng trong kiến trúc đinh tuyến được đòi hỏi cho liên mạng. IPSec có thể đảm bảo rằng:

* Một quảng cáo router (một router mới quảng cáo sự tồn tại của nó) đến từ một router được uỷ quyền
* Một quảng cáo láng giềng (một router tìm kiếm để thiết lập hoặc quản lý một quan hệ láng giềng với một router trong vùng định tuyến khác) đến từ một router được uỷ quyền.
* Một thông báo chuyển hướng đến từ router mà gói khỏi tạo được gửi

tới đó.

* Việc cập nhật định tuyến không bị giả mạo

Nếu không có các biện pháp an toàn như vậy, một đối thủ có thể phá vỡ các truyền thông hoặc làm lệch hướng một vào Traffic.

1. - Kiến trúc IP Security.

IPSec có cấu trúc phức tạp. Để có thể thấy được kiến trúc tổng quan, chúng ta bắt đầu bằng việc tham khảo các tài liệu về IPSec. Sau đó chúng ta sẽ thảo luận các dịch vụ IPSec và giới thiệu khái niệm về liên kết an toàn.

1. - Các tài liệu vềIPSec.

Tháng 8 năm 1995, IETF đề nghị 5 chuẩn về các khả năng an toàn tại mức Internet:

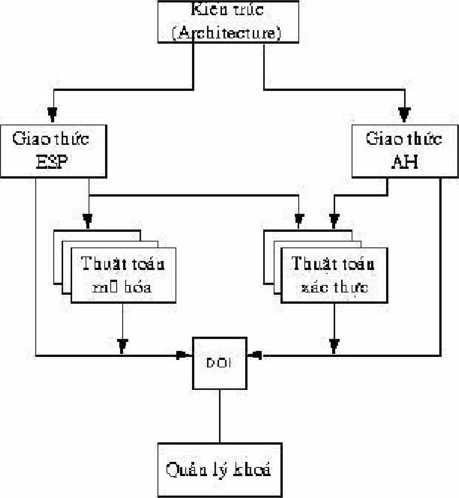
RFC 1825: Tổng quan về một kiến trúc an toàn

RFC 1826 :Mô tả việc xác thực gói IP

RFC 1828: Một cơ chế xác thực đặc trưng

RFC 1827:Mô tả việc mã gói IP

RFC 1829: Một cơ chế mã đặc trưng



Hình 2.3 : Tổng quan tài liệu IPSec

Các đặc trưng này là bắt buộc cho IPv6 và tuỳ chọn cho IPv4. Trong cả hai trường hợp các đặc trưng an toàn được cài đặt là những header mở rộng theo sau header IP chính. Header mở rộng cho xác thực được gọi là header xác thực (Authentication header), header mở rộng cho mã được gọi là Khối an toàn tóm lược (Encapsulating Security Payload - ESP).

Từ một tập các tài liệu ban đầu, “Nhóm công tác giao thức an toàn IF’ đã xây dựng thành 7 nhóm tài liệu sau:

* Kiến trúc : Trình bầy những khái niệm chung, các đòi hỏi an toàn, các định nghĩa và các cơ chế xác định cổng nghệ IPSec
* Khối an toàn tóm lược (ESP) :Trình bầy dạng gói và các vấn đề chung có quan hộ đến việc dùng ESP cho mã và xác thực gói.
* Header xác thực (AH): Trình bầy dạng gói và các vấn đề liên quan có quan hệ tổi việc dùng AH cho xác thực gói.
* Thuật toán mã: Một tập các tài liệu mô tả các thuật toán mã được dùng cho ESP
* Thuật toán xác thực: Một tập các tài liệu mô tả các thuật toán xác thực được dùng cho AH và cho xác thực vởi ESP.
* Quản lý khoá: Các tàỉ liệu mô tả sơ đồ quản lý khoá
* Vùng diễn dịch(DOI): Chứa các giá trị cần thiết giúp cho các tài liệu quan hệ với nhau. Nó bao gồm các định danh cho các thuật toán mã, xác thực cũng như các tham số như thời gian sống của khoá.

2.1.22 - Các dịch vụ IPSec.

IPSec cung cấp các dịch vụ an toàn tại mức IP bằng việc cho phép một hệ thống lựa chọn các giao thức an toàn được đòi hỏi, xác đinh các thuật toán để dùng cho các dịch vụ và các khoá mã được đòi hỏi để cung cấp các dịch vụ được đòi hỏi.

Hai giao thức được dùng để cung cấp an toàn:

* Một giao thức xác thực được chọn lựa bởi header của giao thức, gọi là hader xác thực AH (Authentocation Header)
* Một giao thức kết hợp mã và xác thực được lựa chọn bỏi dạng của gói cho giao thức, khối an toàn tóm lược ESP (Encapsulating Security Payload).

Các dịch vụ bao gồm:

* Điều khiển truy nhập (Access control)
* Toàn vẹn không kết nối (Connectioness integrity)
* Xác thực nguồn gốc dữ liệu (Data origin authentication)
* Huỷ các gói được dùng lại (Rejection of replayed packets)
* Bí mật (Confidentiality)
* Bí mật luồng traffic (Limited traffic flow confidentiality)

Bảng 1 chỉ ra các dịch vụ được cung cấp bỏi các giao thức AH và ESP.

Với ESP có hai trường hợp: có xác thực hoặc không.

Cả hai AH và ESP có dịch vụ điều khiển truy nhập, dựa trên sự phân phối các khoá mật mã và quản lý các luồng traffic có quan hệ tói các giao thức an toàn.

|  |  |  |  |
| --- | --- | --- | --- |
|  | AH | ESP  (chỉ mã) | ESP (mã cộng xác thực) |
| Điều khiển truy nhập (Access control) | + | + | + |
| Toàn vẹn không kết nối (Connectionless integrity) | + |  | + |
| Xác thực nguồn gốc dữ liệu (Data origin authentication) | + |  | + |
| Hủy các gói bị dùng lại (Rejection of replayed packets) | + | + | + |
| Bí mật  (Confidentiality) |  | + | + |
| Bí mật luồng Traffic  (Limited traffic flow confidentiality) |  | + | + |
| Bảng 2.1: Các dịch vụ L | | PSec | |

1. - Các liên kết an toàn.

Một khái niêm quan trọng được dùng cho cả hai cơ chế xác thực và bí mật cho IP là liên kết an toàn (SA). Một liên kết là một quan hệ một chiều giữa người gửi và người nhận cho phép cung cấp các dịch vụ an toàn tói ừaffic dựa trên nó. Nếu muốn thiết lập một kênh trao đổi an toàn hai chiều thì hai cần có hai liên kết an toàn .

Các dịch vụ an toàn được cấp cho một SA được dùng bỏi AH hoặc ESP nhưng không cả hai.

Một liên kết an toàn được xác đinh duy nhất bỏi ba tham số:

* Chỉ số tham số an toàn (Security Parameters Index — SPI):Một xâu bit được gán tới SA và chi có ý nghĩa cục bộ. SPI được chứa ừong AH header và ESP header để cho phép hệ thống nhận lựa chọn SA mà các gói sẽ được nó xử lý.
* Địa chỉ đích IP (IP Destination Address): Là địa chỉ của điểm đích của SA, nó có thể là hệ thống người dùng đầu cuối hoặc một hệ thống mạng như Fiïewall, router.
* Định danh giao thức an toàn (Security Protocol Identifier): Chỉ ra liên kết là một liên kết an toàn AH hay ESP.

Từ đó trong gói IP bất kỳ, một liên kết an toàn được xác đinh duy nhất bỏi địa chỉ đích trong IPv4 và giá trị SPI trong header mở rộng (AH hoặc ESP).

a - Các tham sổ SA.

Trong hệ thống IPSec có một cơ sở dữ liệu liên kết an toàn xác đinh các tham số liên kết với mỗi SA. Một liên kết an toàn được xác định bỏi các tham số sau:

* Bộ đếm số liên tiếp ( Sequence Number Counter) : Một giá trị 32 bit được dùng để sinh trường số liên tiếp (Sequence Number) trong AH hoặc ESP header
* Tràn bộ đếm liên tiếp (Sequence Number Overflow):Một cờ chỉ ra việc tràn của bộ đếm liên tiếp và sinh ra một sự kiện có thể kiểm tra và chặn việc truyền các gói trên SA này.
* Cửa sổ Anti\_Replay:dùng để xác đinh xem các gói AH và ESP có bị dùng lại hay không
* Thông tin AH: Thuật toán xác thực, các khoá, thời gian sống của khoá và các tham số quan hệ được dùng vói AH
* Thông tin ESP:thuật toán mã và xác thực, các khoá, các giá trị khởi tạo, thời gian sống của khoá và các tham số liên quan được dùng vói ESP
* Thời gian sống của một liên kết an toàn :khoảng thời gian mà sau đó một SA phải được thay thế với một SA mới (và một SPI mới ) hoặc ngắt
* Phương thức giao thức IPSec (IPSec Protocol Mode):Tunnel, transport
* Path MTU :CỠ tối đa của một gói có thể truyền mà không cần phân mảnh.

Cơ chế quản lý khoá được dùng để phân phối khoá được đi với cơ chế xác thực và bí mật chỉ bằng con đường chỉ số tham số an toàn SPI.

Từ đó xác thực và bí mật đã được xác định một cách độc lập vói cơ chế quản lý khoá.

*b-SA* Selecter.

IPSec cung cấp cho người dùng sự mềm dẻo đáng kể trong cách thức mà các dịch vụ IPSec được gắn với IP traffic. IPSec cung cấp khả năng phân biệt các traffic được bảo vê bằng IPSec và các traffic không cần sử dụng IPSec.

Cách thức mà IP traffic quan hệ với các SA cụ thể (hoặc không cần SA trong trường họp traffic không cần IPSec) là Cơ sở dữ liệu chính sách an toàn

spdT

Trong trường hợp đơn giản nhất một SPD chứa các đầu vào, mỗi đầu vào định nghĩa một tập con của IP traffic và chỉ tói một SA cho traffic này. Trong các môi trường phức tạp hơn, có nhiều đầu vào quan hệ vói một SA đơn hoặc nhiều liên kết SA với một đầu vào SPD đơn. Mỗi đầu vào SPD được đinh nghĩa bởi một tập các giá trị trường giao thức IP và giao thức mức cao, được gọi là selectors.Trong thực tế, các selector này được dùng để lọc các traffic ra để gán nó với một SA đặc trưng.Việc xử lý các gói IP ra tuân theo các bước sau:

* So sánh các giá trị của các trường phù họp trong gói (Các trường selector) với SPD để tìm một đầu vào tương ứng, nó chỉ tói các SA hoặc không chỉ tới SA nào
* Quyết định SA gán cho gói
* Tiến hành các thao tác xử lý IPSec được đòi hỏi (Xử lý AH hoặc ESP).
* Các selector sau xác định một đầu vào SPD:
* Địa chỉ IP đích (Destination IP Address):Có thể là một địa chỉ IP đơn, một danh sách hoặc một vùng địa chỉ, một địa chỉ mask (cho phép nhiều hệ thống đích chia sẻ cùng một SA như các máy nằm sau firewall).
* Địa chỉ IP nguồn (Source IP Address) : Có thể là một địa chỉ IP đơn, một danh sách hoặc phạm vi địa chỉ, địa chỉ massk (cho phép nhiều hệ thống đích chia sẻ cùng một SA như các máy nằm sau firewall).
* Định dang người dùng (UserID):một định danh người dùng trong hệ điều hành.
* Mức nhạy cảm dữ liệu (Data Sensitivity Level): Được dùng cho các hệ thống cung cấp an toàn luồng thông tin
* Giao thức mức vận tải (Transport Layer Protocol): Nó có thể là một số giao thức riêng lẻ, một danh sách các số giao thức hoặc một phạm vi các số giao thức).
* Giao thức IPSec (IPSec Protocol) :bao gồm AH hoặc ESP hoặc AH/ESP

• Cổng nguồn và đích (Source and Destination ports): có thể là cổng TCP hoặc UDP, danh sách các cổng hoặc Wildcard por.

c — Các phương thức Transport và Tunnel (Transport and Tunnel mod). Cả AH và ESP đều hỗ trợ cả hai phương thức Transport và Tunnel.

1. - Phương thức Transport.

Phương thức Transport cho phép bảo vệ phân đoạn tầng vận tải (TCP/UDP header và dữ liệu ứng dụng) hoặc các gói ICMP. Phương thức Transport được dùng để truyền thông end-to-end giữa hai host. Khi một host dùng AH hoặc ESP trong phương thức transport , payload là dữ liệu đi theo sau IP header. ESP trong phương thức trasport mã và xác thực (lựa chọn) IP payload (phân đoạn tầng vận tải hoặc các gói ICMP) nhưng không bảo vệ IP header.AH trong phương thức transport xác thực IP payload và các thành phần lựa chọn trong IP header.

1. - Phương thức Tunnel.

Phương thức Tunnel cho phép bảo vệ toàn bộ gói IP. Để thực hiện được điều này, sau khi trường AH và ESP được thêm tói gói IP, toàn bộ gói IP cộng với các trường an toàn được coi như payload của gói IP bên ngoài mới (new “outer” IP packet) với một IP header bên ngoài mói (new outer IP header). Toàn bộ gói IP ban đầu được chuyển qua một “tunnel” (đường hầm) từ một điểm của mạng IP tới điểm khác. Các router sẽ không kiểm tra header của gói IP bên trong (inner IP header). Bởi vì gói IP nguyên bản ban đầu được tóm lược (encapsulated), gói IP mới có thể có các địa chỉ nguồn và đích khác hoàn toàn đồng thời được bảo vệ an toàn.

Phương thức Tunnel được dùng khi một hoặc cả hai đầu của SA là một gateway an toàn, như firewall hoặc router cài đặt IPSec.

Các gói không được bảo vệ được sinh bỏi các host như vậy được đi qua một đường hầm thông qua các mạng bên ngoài bởi các SA phương thức Tunnel được cài đặt bởi phần mềm IPSec trong Firewall hoặc router an toàn tại biên của mạng cục bộ.

Sau đây là một ví dụ về sự làm việc của phương thức Tunnel.

Máy A trên một mạng sinh ra một gói IP vói địa chỉ đích của máy B trên mạng khác. Gói này được chuyển từ máy ban đầu tói Firewall hoặc router an toàn tại biên của mạng của A. Fứewall lọc tất cả các gói ra để xác đinh sự cần thiết cho việc xử lý IPSec. Nếu một gói từ A tói B đòi hỏi IPSec, Fkewall tiến hành xử lý IPSec và tóm lược gói vào một IP header bên ngoài. Địa chỉ nguồn của gói IP bên ngoài là địa chi của Firewall, địa chỉ đích có thể là địa chỉ của firewall tại biên của mạng của B, các router trung gian chỉ kiểm tra header của IP bên ngoài. Tại firewall của máy B, IP header bên ngoài được loại bỏ, gói IP bên trong được đi tới B.

ESP trong phương thức Tunnel mã và xác thực (tuỳ chọn) toàn bộ gói IP bên trong, bao gồm cả IP header. [[1]](#footnote-1)

AH trong phương thức Tunnel xác thực toàn bộ gói IP bên trong và các phần lựa chọn của IP header bên ngoài

Bảng 2 tóm tắt chức năng của các phương thức transport và tunnel

|  |  |  |
| --- | --- | --- |
|  | Transport Mode SA | Tunnel Mode SA |
| Ah | Xác thực IP payload và Phần lựa chọn của IP header | Xác thực toàn bộ gói IP bên ừong (gói nguyên bản ban đầu) cộng vái những phần lựa chọn của IP header ngoài ợp header mới) |
| Esp | Mã IP payload | Mã gói IP bên trong |
| Esp với xác thực | Mã IP payload và xác thực IP payload (không bao gồm IP header) | Mã gói IP bên trong Xác thực gói IP bên ừong |

Bảng 2.2 : Chức năng của phương thức Tunnel và transport

1. - Header xắc thực (Authentication Header).

Header xác thực cho phép đảm bảo sự toàn vẹn dữ liệu và xác thực các gối IP. Đặc trưng toàn vẹn dữ liệu đảm bảo rằng các thay đổi nội dung của một gói trong khi truyền sẽ bị phát hiện. Đặc trưng xác thực cho phép một hệ thống đầu cuối và thiết bị mạng xác thực người dùng hoặc ứng dụng và lọc ưa£fỉc đứng đắn, nố cũng chặn tấn cổng “giả địa chỉ” (address spoofing).

AH cũng đảm bảo chống lại tấn công replay. Xác thực dựa ttên việc dùng mã xác thực thông báo, hai đối tượng phải cố một khoá bí mật chia sẻ.

Bit: 0

16

31

Next Header Payload Length

RESERVED

Security Para maters Index (SPI)

Sequence Number

Authentication Data (variable)

Hình 2.4: Header xấc thực IPSec Header xác thực bao gồm các trường sau:

* Header tiếp theo (Next header - 8 bits):định danh kiểu của header theo ngay sau header này.
* Độ dài Payload (Payload length -8 bits): độ dài của header xác thực trong các từ 32 bits trừ 2. Ví dụ, độ dài ngầm định của trường dữ liệu xác thực là 96 bits, hoặc 3 từ 32 bits. Với một header cố định 3 từ, tổng số sẽ là 6 từ ừong header và trường độ dài payload cố giá trị là 4.
* Reserved (16 bits): dự trữ cho tương lai.
* Chỉ số tham số an toàn (Security Parameter Index - 32 bits) :định danh một liên kết an toàn.
* SỐ liên tiếp (Sequence Number - 32 bits): Một gía tri bộ đếm tăng đều đều
* Dữ liệu xác thực (độ dài thay đổi): một trường độ dài thay đổi (phải là số nguyên của các từ 32 bits) chứa “Giá tri kiểm ưa toàn vẹn - Integrity Check Value ICV) cho gói này.

1. - Dịch vụAntỉ-Repỉay.

Trong tán công replay, kẻ tấn công lấy một bản sao của gối đã được xác thực và sau đó truyền lại tới đích chủ ý. Việc nhận đúp dẫn đến một vài hậu quả không mong muốn. Trường số Hên tiếp (Sequence Number) được gán để chống lại việc tẩn công như vậy.

Đầu tiên, chứng ta thảo luận việc sinh số liên tiếp bỏti người gửi và sau đó chúng ta xem xét việc xử lý của người nhận.

:ở c LÊL aổ GO chah

rrrrmi

N-W

/

*TTT*

nxn

Đánh, dấu D.CU.  
gói đú.D£ âuạc nhận

\ H+-1

\_ \ > . Ktưõag đáahdãu aếu

gổi đĩi.ag thõng dub'G ahậa

m

Dịch, chuyên elra Bổ CƯÍÌI  
ảíLđiinỹ đuọe nhạ n i mip

phải elrat Bổ

Hình 2.5 : Cơ chếAnti-Replay

Khỉ một SA mái được thiết lập, người gửi khởi tạo một bộ đếm số liên tiếp. Mỗi lần một gói được gửi theo SA này, người gửi tăng bộ đếm và đặt giá trị vào trường “Số liên tiếp”. Như vậy giá tri đầu tiên được dùng là 1. Nếu Anti-Replay được cho phép (ngầm định), người gửi phải không cho phép “Số liên tiếp” quay vòng tròn theo chu kỳ khi vượt 232 -1 được ưở lại 0. Nói cách khác có nhiều gói đúng với cùng số liên tiếp. Nếu giới hạn 232 -1 đã đạt được, người gửi phải ngắt SA này và thoả thuận một SA với khoá mới.

Bởi vì IP là không kết nối, các dịch vụ không chắc chắn, giao thức không đảm bảo các gói sẽ được giao theo đúng thứ tự và không đảm bảo rằng tất cả các gói sẽ được giao. Từ đó tài liệu xác thực IPSec chỉ ra rằng ngưòi nhận phải cài đặt một cửa sổ cỡ w, vói ngầm đinh w =64. Mép phải của cửa sổ biểu diễn số liên tiếp cao nhất N, là số liên tiếp của một gói đã được nhận đúng.Với gói bất kỳ có số liên tiếp trong phạm vi từ N-W+1 tới N đã được nhận đúng (nghĩa là đã được xác thực đúng), khe tương ứng trong cửa sổ được đánh dấu.

* Nếu một gói rơi vào trong cửa sổ và là mới, MAC được kiểm tra. Nếu gói được xác thực, khe tương ứng trong cửa sổ được đánh dấu.
* Nếu gói nhận được nằm ở bên phải cửa sổ và là mói, MAC được kiểm tra. Nếu gói được xác thực, cửa sổ được tiến sao cho số liên tiếp này là biên phải của cửa sổ và khe tương ứng trong cửa sổ được đánh dấu.
* Nếu gói nhận được nằm ở bên trái cửa sổ hoặc nếu xác thực lỗi, gói bị huỷ

1. - Giá trị kiểm tra tính toàn vẹn (Integrity Check Value):

Trường dữ liệu xác thực chứa giá tri được gọi là Giá trị kiểm tra tính toàn vẹn. ICV là một mã xác thực thông báo hoặc là một phần đã được cắt của mã được tạo bởi thuật toán MAC.

Các sơ đồ xác thực bao gồm:

* HMAC-MD5-96
* HM AC-SHA-1 -96

Cả hai sơ đồ này dùng thuật toán HMAC, sơ đồ đầu tiên dùng hàm hash MD5 và sơ đồ thứ hai dùng hàm hash SHA-1.

Trong cả hai trường hợp, giá trị HMAC được tính nhưng bị cắt bằng việc chỉ dùng 96 bits đầu tiên, nó là giá tri ngầm dinh của trường dữ liệu xác thực.

MAC được tính trên các thành phần sau:

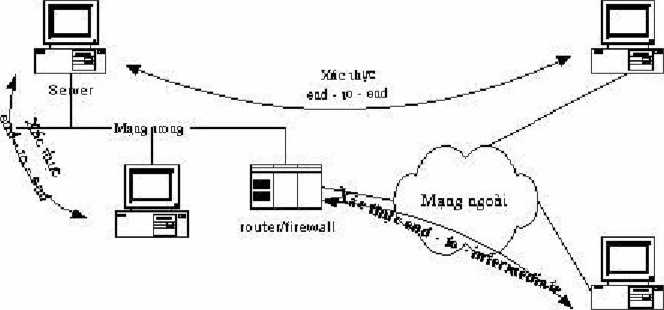
* Các trường IP header hoặc không thay đổi trong khi tmyền hoặc có thể đoán trước khi đến điểm đầu cuối của AH SA. Các trường có thể thay đổi trong khi truyền và các giá trị không thể đoán trước đực gán giá trị 0 để tính tại nguồn và đích.
* AH header trừ trường dữ liệu xác thực. Trường dữ liệu xác thực được đặt bằng 0 để tính cho cả nguồn và đích.
* Toàn bộ dữ liệu giao thức mức trên được thừa nhận là không thay đổi trong khi truyền (phân đoạn tầng vận tải hoặc gói IP bên trong trong phương thức tunnel).

Với IPv4, ví dụ về các trường không thay đổi là độ dài IP header và địa chỉ nguồn. Ví dụ về trường thay đổi được nhận giá trị 0 trước khi tính MAC là trường Time to Live và trường Header checksum.

Chú ý rằng cả địa chi nguồn và địa chỉ đích được bảo vệ, nên việc tấn công giả địa chi (address spoofing) bị chặn.

1. - Cấc phương thức Transport và Tunneỉ:

Hình 2.9 chỉ ra hai cách thức trong đó dịch vụ xác thực IPSec có thể được dùng.



Hình 2.6: Xác thực end-to-end và end-to-intermediate

Trong trường hợp thứ nhất, xác thực được cung cấp trực tiếp giữa một server và một chent, Client có thể trên cùng một mạng hoặc trên một mạng mở rộng. Khỉ clỉent và server chia sè một khoá bí mật, quá trình xác thực là an toàn. Trường hợp này dùng ưansport mode SA. Trong trường hợp khác, một ưạm từ xa xác thực nó với Fửewall, hoặc để truy nhập tới toàn mạng ữong hoặc bởi vì server được đòi hỏi không hỗ ượ tính năng xác thực. Trường hợp này dùng tunnel mode SA.

❖ Tunnel mode và giao thức AH (Tunnel mode AH)

Đỉipc tẳc ỉhưr 5[ừ-ic IIUÍÍC3 c5 ihií ihay đái. iraiiặ New IP ft&ddi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| New  |p headei | AH | IP  header | TCPAJDP  header | Dư liêu, ứug du Lg |

Hình 2.7 : Cấu trúc gói IP với tunnel mode AH

Với Tunnel mode AH, toàn bộ gói IP ban đầu được xác thực và AH được chèn vào giữa IP header ban đầu và IP header mới. Trong IP header mới chứa

địa chỉ IP cả các gateway an toàn.Toàn bộ gói IP ban đầu được bảo vệ bồi AH. Riêng IP header mối được bảo vệ ngoại trừ những trường có thể thay đổi Transport mode với AH (Transport mode AH)

3

4

Điídc xác t tiLCc trừ cấc trưòtig có thể thay dỏi ►

|  |  |  |  |
| --- | --- | --- | --- |
| IP  header | AH | TCP^UDP  header | Dữ liều, ứag dung |

Hình 2.8 : Cấu trúc gói ỈP với Transport mode AH

Với Transport mode AH thì AH được chèn sau IP header và trước phân đoạn tầng vận tải. Xác thực được thực hiện đối với toàn bộ gói, ngoại trừ những trường có thể thayđổi trong IP header (được gán giá trị 0 khi tính mã xác thực thông báo).

2.1.4- Khối an toàn tóm lược (Encapsulating Security Payload).

Khối an toàn tóm lược cung cấp các dịch vụ bí mật, bao gồm bí mật nội dung thông báo và bí mật luồng traffic. Với đặc trưng lựa chọn ESP có thể cung cấp các dịch vụ xác thực giống như AH.

2.1.4.ỉ -DạngESP.

Đít: 0

|  |  |  |  |
| --- | --- | --- | --- |
| Security Parameters Index (SRI) | | | |
| Sequence Number | | | |
|  | Payload Data | | |
|  | Padding (P-255 byte?) | | |
|  |  | Pad Length | Next Headei |
| Authentication D3ta | | | |

18

24

31

Xác ttiLíc Bí rítìt

ir \*

Hình 2.9 : Cấu trúc của IPSec ESP

* Chỉ số tham số an toàn (Security Paramater Index - 32 bits) :đinh danh liên kết an toàn
* Số liên tiếp (Security Number - 32 bits): một giá trị bộ đếm tăng đều đặn, giúp cho việc chống lại tấn cổng replay
* Payload data (thay đổi):là một phân đọc tầng vận tải (phương thức transport), hoặc gối ỈP (phương thức tunnel) được bảo vệ bằng mật mã
* Padding (0-255 bytes): phục vụ cho các kỹ thuật mã khối
* Pad length (8 bits):chỉ ra số byte của trường Padding.
* Next Header (8bỉts): Xác định kiểu của dữ liệu được chứa ưong trường Payload data (8 bỉts): xác đỉnh kiểu của dữ liệu được chứa ừong trường

Payload data bằng việc xác đinh header đầu tiên ừong payload này(ví

dụ như TCP header)

• Dữ liệu xác thực (Authentication Data) (Thay đổi):Một trường độ dài

thay đổi (phải là số nguyên của từ 32 bits) chứa “Giá trị kiểm tra toàn

vẹn” được tính trên gói ESP trừ trường dữ liệu xác thực.

2.1.42 - Thuật toán mã và xác thực.

Payload data, Padding, Pad Length và next header được mã bỏi dịch vụ ESP. Nếu thuật toán được dùng để mã payload đòi hỏi dữ liệu đổng bộ mã như vec tơ khởi tạo IV, thì dữ liệu này được chứa ngay đầu tiên của trường Payload data. Nếu có IV đi cùng thì nó thường không được mã mặc dù nó thường được coi như một phần của bản mã.

Các kỹ thuật mã khối được gán đinh danh trong tài liệu DOI là:

* Three-key triple DES
* RC5
* IDEA
* Three-key triple IDEA
* CAST
* Blowfish

Cũng như với AH, ESP cho phép dùng MAC vói độ dài ngầm đinh 96 bits. Cũng như với AH, các sơ đồ xác thực là HMAC-MD5-96, HMAC-SHA- 1-96.

2.1.43 - Padding:

Trường Padding phục vụ cho các mục đích sau:

* Nếu thuật toán mã hoá đòi hỏi văn bản rõ là bội của một vài bytes (nghĩa là bội của một khối đơn cho mã khối), trường Padding được dùng để mở rộng văn bản rõ (bao gồm các trường Payload Data, Padding, Pad length, Next header) tói độ dài được đòi hỏi.
* Dạng ESP đòi hỏi rằng các trường Pad Length và Next Header phải nằm bên phải của từ 32 bits
* Padding bổ xung có thể thêm để cho phép giữ bí mật luồng traffic bằng việc giấu độ dài thực sự của payload

2.1.4.4 - Quá trình xử lý gói tin trong ESP.

Quá trình sử dụng ESP (Encapsulate Security Payload) để mã hóa vùng payload của IP packet nguyên thủy, thực hiện việc xác thực gói tin IP thông qua hàm HMAC và đúng gói lại (theo cơ chế IP-in-IP). Quy trình được mô tả qua các bước sau:

Bl. Khai báo dữ liệu gói tin ESP.

B2. Khai báo dữ liệu cho gói tin AH.

B3. Khai báo cấu trúc cho việc thực hiện hàm HMAC.

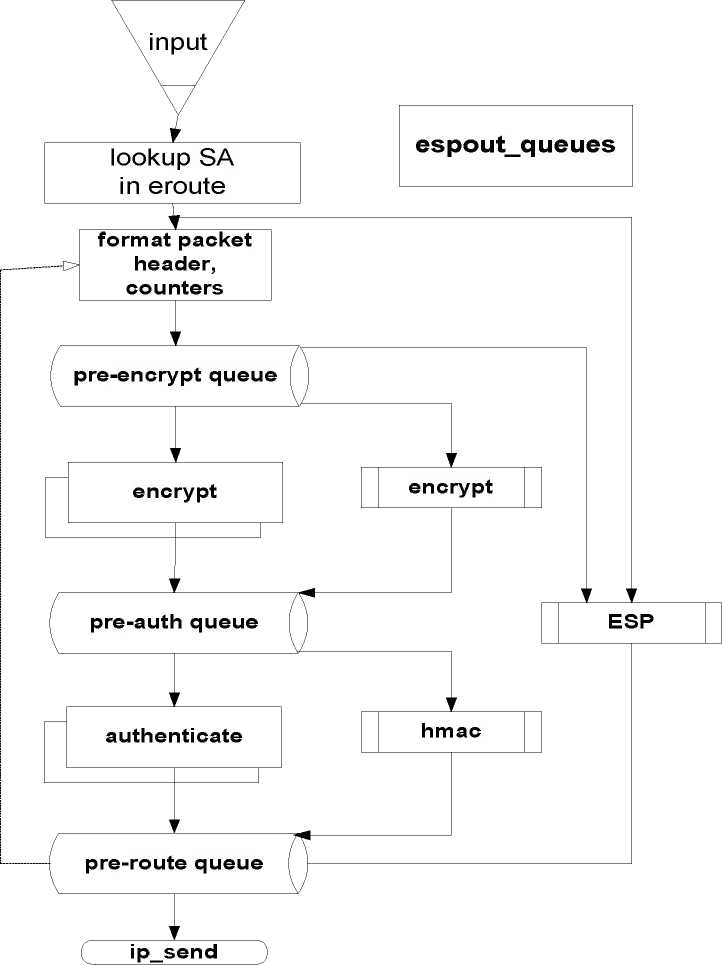
B4. Tính headroom và tailroom cho gói tin AH.

B5.Chọn thuật toán mã hoá.

B6. Xác định thuật toán HMAC và tính AH.

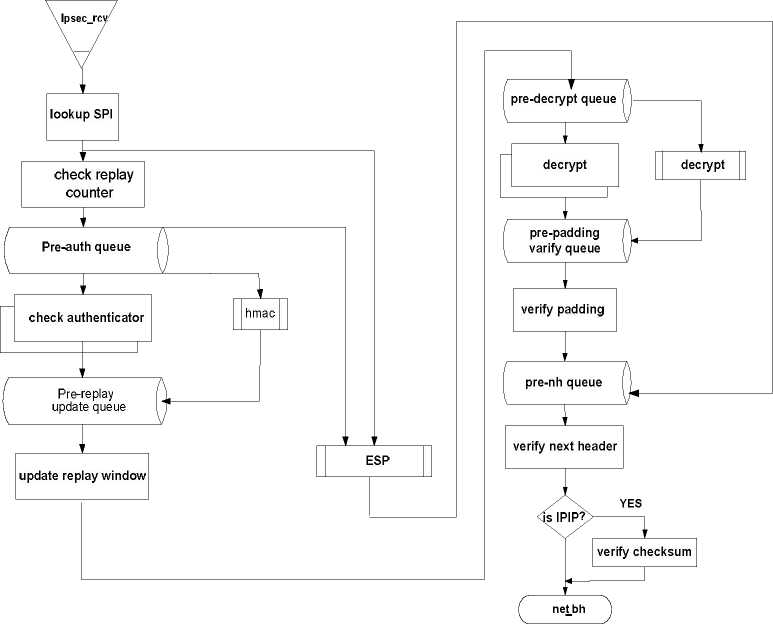
B7. Chuẩn bị head room và tail room cho gói HMAC và ESP. B8. Thực hiện phép toán ESP.

B9. Đúng gói lại gói tin ESP trong IPsec driver.



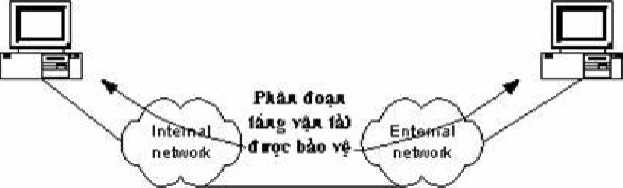
Hình 2.10 : Xử lý gói tin đi của giao thức ESP

Quá trình nhận được gói tin ESP bên nhận sẽ xử lý ngược lại: Xác thực tính toàn vẹn của gói tin, giải mã gói tin bằng hàm mã khối, loại bỏ phần IP Header (IP-in-IP) ngoài để cho ta gói tin nguyên thủy ban đầu, quá trình này được thể hiện bằng sơ đồ sau:



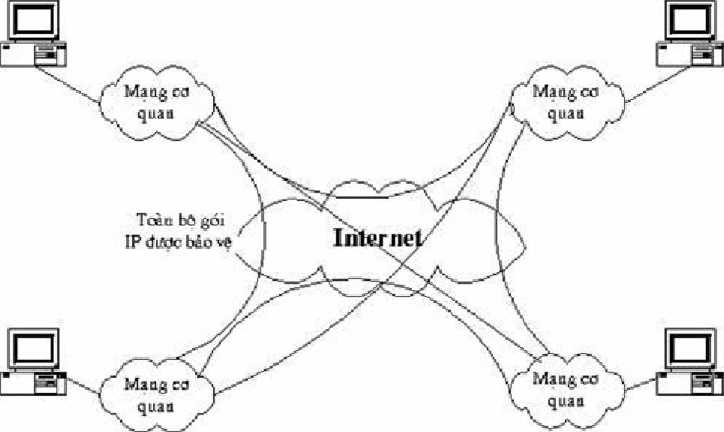
Hình 2.11 : xử lý gói tin đến của giao thức ESP

1. - Các phương thức Transport và Tunnel.



Hình 2.12 : An toàn với phương thức transport

Hình 2.15, hình 2.16 chỉ ra hai cách trong đó dịch vụ IPSec ESP có thể dùng. Trong hình 2.15, mã hóa ( xác thực lựa chọn) được cung cấp trực tiếp giữa các host. Hình 2.16 chỉ ra việc dùng tunnel mode để tạo ra mạng riêng ảo. Trong ví dụ này một tổ chức có 4 mạng riêng kết nối vởi nhau qua Internet.



Hình 2.13 : An toàn với phương thức tunnel

Các hosts trên mạng ưong dùng Internet để truyền dữ liệu nhưng khổng giao tiếp với các máy khác trên Intemet.Bằng việc ngắt các tunnel tại Gateway an toàn của mỗi mạng ưong, cấu hình cho phép các hosts không phải cài đặt các chức năng an toàn.

Kỹ thuật trước được hỗ trợ bởi việc dùng transport mode SA, trong khi kỹ thuật sau dùng tunnel mode SA.

Trong phần này chúng ta xem xét phạm vi cùa ESP cho cả hai phương

thức.

❖ Tunnel mode với giao thức ESP (Tunnel mode ESP)

Đưac xác TtLUE

EXracằm toá

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nu»  IP header | ESP  hrd | IP  headei | TCP/UDP  header | Dir hâu ứmg dụag | SSP  trk | ESP  ỈUttl |

-“Gòi IP baa diu

Hình 2.14 : Cấu trúc gối IP trong tunnel mode ESP

Tunnel mode ESP được dùng để mã toàn bộ gói IP. Trong mode này một ESP header được thêm vào phía trước gói và sau đó toàn bộ gói cùng vởi ESP trailer được mã. Phương pháp này được dùng để chống lại việc phân tích traffic.

Bởi vì IP header chứa địa chỉ đích và các thông tin đường dẫn khác nên không thể đơn giản truyền gói IP đã mã cùng vcd ESP header. Các router trung gian sẽ không thể sử lý các gói như vậy cần thêm vào toàn bộ khối (bao gồm ESP header, gói IP đã mã, Dữ liệu xác thực) một IP header mớỉ chứa các thông tín phù hợp cho việc đỉnh tuyến nhưng không giúp cho việc phân tích ữaffic.

Xem xét trường hợp một host bên ngoài muốn truyền thông với một host trên mạng trong được bảo vệ một firewall và trong đó ESP được cài đặt tại host ngoài và Firewall.

Các bước sau xẩy ra khỉ truyền một phân đoạn tầng vận tải từ host bên ngoài tới host bên ương:

* Máy nguồn chuẩn bị một gói IP bên trong (innerlP packet) với địa chỉ đích là máy bên trong muốn đến. Gói này được nối một ESP header về phía trước, sau đó gói và ESP trailer được mã và dữ liệu xác thực (Authentication Data) có thể được thêm vào. Khối thu được được tốm lược (encapsulated) với một IP header mối, ưong đố địa chỉ đích là Fkewall để tạo thành gói IP bên ngoài (outer IP packet)
* Gói IP bên ngoài được đinh tuyến tới firewall đích. Mỗi router trung gian cần kiểm tra và xử lý IP header bên ngoài nhưng không cần kiểm ưa phần mã.
* Firewall đích kiểm ưa và xử lý IP header bên ngoài. Sau đó trên cơ sở của SF1 ương ESP header, nứt đích giải mã phần còn lại của gối để khôi phục phần rõ của gói IP bên ương. Gói này sau đó được truyền ương mạng bến ưong.
* Gói bên trong được chuyển qua các router (nếu có) của mạng ương để tới máy đích.
* Transport mode với giao thức ESP (Transport mode ESP)

-Đưoc xãc líiute-

-Difldema hũá

|  |  |  |
| --- | --- | --- |
|  | ESP | ESP |
|  | Wr | ÍLIỆh |

Hình 2.15 : Cấu trúc gói IP trong tranport mode ESP

|  |  |  |  |
| --- | --- | --- | --- |
| IP | ESP | TCPAJDP |  |
|  |  | hè^aỉMi |  |

Dữ liều ứa g dung

Transport mode ESP được dùng để mã và xác thực dữ liệu lựa chọn dữ liệu gói IP (bao gồm TCP/UDP header và dữ liệu ứng dụng). Một ESP header được chèn vào phía trước header tầng vận tải (TCP/UDP header) và một ESP ưaỉler (Padding,Pad Length ,Next header) được dặt sau góỉ IP,nếu xác thực

được lựa chọn, trường dữ liệu xác thực ESP (ESP auth) được thêm vào sau ESP trailer.

Toàn bộ phân đoạn tầng vận tải cộng ESP trailer được mã.Xác thực được thực hiên trên toàn bộ phần được mã và ESP header.

Phương thức Transport có thể tóm tắt như sau:

* Tại nguồn, khối dữ liệu bao gồm ESP trailer cộng toàn bộ phân đoạn tầng vận tải được mã và phần rõ của khối này được thay thế bỏi phần mã của nó để tạo gói IP truyền đi. Xác thực được thêm nếu được lựa chọn.
* Sau đó gói được định tuyến đến đích. Mỗi router trung gian cần kiểm tea và xử lý IP header nhưng không kiểm tra phần mã.
* Đích kiểm tra và xử lý IP header. Sau đó dựa trên SPI trong ESP header, nút đích giải mã phần còn lại của gói để khôi phục nội dung rõ của phân đoạn tầng vận tải

Phương thức Transport cho phép giữ bí mật dữ liệu của các ứng dụng bất kỳ. Phương thức này có ưu điểm là tăng ít tổng độ dài của gói IP. Điểm yếu của phương thức này là có thể phân tích traffic trên các gói truyền.

1. - Quản lý khóa.

Việc quản lý khóa cuả IPSec bao gồm việc xác định và phân phối các khóa bí mật. Một đòi hỏi đặc trưng là 4 khóa cho truyền thông giữa hai ứng dụng: truyền và nhận các cặp cho cả AH và ESP. Tài liệu kiến trúc IPSec cho phép hỗ trợ hai kiểu quản lý khóa:

Bằng tay (Mannual): Người quản trị hệ thoonhg cấu hình bằng tay mỗi hệ thống với các khóa của các hệ thống truyền thông khác. Điều này là phù hợp với môi trường nhỏ, tĩnh.

Tự động (Automated): Một hệ thống tự động cho phép tạo theo yêu cầu các khóa cho các SA và các phương tiện dùng khóa trong hệ thống phân tán lớn với cấu hình đòi hỏi.

* Giao thức phân phối khóa tự động ngầm đinh cho IPSec là

ISAKMP/Oakley và chứa các thành phần sau: Oakley Key

Determination Protocol: Oakley là một giao thức trao đổi khóa dựa trên thuật tóan Diffie-Heilman nhưng an toàn hơn. Oakley là tổng quát, nó không bắt buộc các dạng riêng biệt.

* Internet Security Association and Key Management Protocol (ISAKMP): ISAKMP cung cấp một khuôn mẫu (framework) cho quản lý khóa và hỗ trợ các giao thức cụ thể, bao gồm cả định dạng cho việc thỏa thuận các thuộc tính an toàn.

ISAKMP không bắt buộc một thuật toán trao đổi khóa riêng biệt, nó bao gồm một tập các kiểu thông báo cho phép dung một vài thuật tóan trao đổi khóa. Oakley là thuật tóan trao đổi khóa được dùng với phiên bản ban đầu của ISAKMP.

Chúng ta bắt đầu với tổng quan của Oakley và sau đó là ISAKMP

2.1.5.1- Giao thức xác định khóa Oakley (Oakley Key Determination Protocol)

Oakley là sự tinh lọc của thuật tóan hao đổi khóa Diffie-Hellman Trong giao thức Diffie-Hellman, hai người dùng A và B thỏa thuận vói nhau từ trước hai tham số chung:

* q: số nguyên tố lớn
* a : phần tử nguyên thủy của q.

A chọn một số nguyên ngẫu nhiên Xa làm khóa bí mật và truyền tói B khóa công khai của nó là YA=a Xa.

B chọn một số nguyên ngãu nhiên XB làm khóa bí mật và truyền tới A khóa công khai của nó là YB=a xb.

Mỗi bên tính khóa bí mật:

K= (Yb) Xa mod q = (YA)xb mod q = (X XaXb mod q Thuật toán Diffie-Hellman có hai đặc trưng chính sau:

* Các khóa bí mật chỉ được tạo khi cần thiết. Không cần phải chứa các khóa bí mật trong một khoảng thời gian dài
* Việc thỏa thuận dựa trên các tham số chung

Tuy nhiên Thuật toán Diffie-Hellman có một số điểm yếu sau:

* Nó không cung cấp thông tin bất kỳ về các đinh danh của các bên

• Nó dễ bị tấn công kiểu “man -in-the-middle - Người đàn ông ở giữa”, trong đó một người c thứ ba mạo danh là B trong khi truyền thông với A và mạo danh A trong khi truyền thông vói B. Cả A và B đều thỏa thuận khóa với c, sau đó c có thể nghe các thông tin được trao đổi giữa A và B.

Kiểu tấn công “Người đàn ông ở giữa” tiến hành như sau:

* B gửi khóa công khai YB trong một thông báo tói A
* Một kẻ E chặn thông báo này. E giữ lại khóa công khai của B và gửi một thông báo tới A với định danh người dùng là B nhưng chứa khóa công khai YE của E. Thông báo này được gửi trong cách thức như nó được gửi từ hệ thống của B. A nhận thông báo của E và chứa khóa công khai của E với định danh người dùng của B.

Tương tự E gửi một thông báo tói B vói khóa công khai của E, nhưng với định danh người dùng là A.

* B tính một khóa bí mật KI dựa trên khóa bí mật của B và YE. A tính một khóa bí mật K2 dựa trên khoa bí mật của A và YE. E tính KI dùng khóa bí mật XE của E và YB và tính khóa K2 dùng XE và YA.
* Từ bây gìơ E có thể chuyển tiếp các thông báo từ A tói B và từ B tói A, thay đổi một cách phù hợp nội dung bản mã bằng cách dùng các khóa chung Kl, K2. Trong khi cả A và B đều không biết rằng chúng đang truyền thông với E.
* Nó đòi hỏi nhiều phép tính tóan. Dễ bị tổn thương do tấn công clogging, trong đó đối thủ đòi hỏi một số lớn các khóa. Nạn nhân dành số lượng đáng kể tài nguyên tính toán cho việc làm vô ích là tính lũy thừa mô

đun.

Okley được thiết kế để giữ lại những ưu điểm của Thuật toán Diffïe- Hellman và khắc phục những điểm yếu của nó.

a - Các đặc trưng của Oakley :

Thuật toán Oakley được mô tả bởi năm đặc trưng sau:

* Nó dùng cơ chế cookies để chống lại tấn công clogging
* Nó cho phép hai bên thỏa thuận một nhoam, trong đó xác đinh các tham số chung của trao đổi khóa Diffie-Hellman
* Nó dùng nonces để chống lại tấn công replay
* Nó cho phép trao đổi các giá trị khóa chung của Diffie-Hellman
* Nó xác thực trao đổi Diffie-Hellman để chống lại kiểu tấn công “Ngưòi đàn ông ở giữa”

Chúng ta đã thảo luận về Diffie-Hellman. Bây giờ chúng ta xem xét thêm về một số tấn công đối với thuật toán này.

Đầu tiên là vấn đề tấn công “clogging”. Trong tấn công này, một kẻ thù giả mạo địa chỉ nguồn của người dung họp pháp và gửi khóa công khai Diffie- Hellman tới nạn nhân. Nạn nhân thực hiện phép lũy thừa mô đun để tính khóa bí mật.Các thông báo được lặp lại của kiểu này có thể làm cản trở hệ thống của nạn nhân với những công việc vô ích.

Trao đổi cookie đòi hỏi rằng các mỗi bên gửi một số giả ngẫu nhiên, gọi là cookie

Trong thông báo khởi tạo và bên kia sẽ báo nhận. Việc báo nhận này phải được lặp lại trong thông báo đầu tiên của trao đổi khóa Diffie- Hellman.Nếu địa chỉ nguồn bị giả thì kẻ thù không nhận được câu trả lời.Như vậy một kẻ thù chỉ có thể ép một người dùng sinh ra thông báo báo nhận nhưng không tiên hành tính giá trị lũy thừa mô đun trong Diffie-Heilman. ISAKMP yêu cầu cookie phải thỏa ba yêu cầu sau:

* Cookie phải phụ thuộc vào các bên cụ thể. Điều này chặn kẻ tấn công lấy được cookie bằng cách dùng địa chỉ IP thật và cổng UDP và sau đó dùng nó để làm lụt nạn nhân vói các đòi hỏi từ các địa chỉ IP và các cổng được chọn ngẫu nhiên.
* Không một ai khác ngoài người công bố cookie có thể sinh ra cookie được chấp nhận . Điều này ngụ ý rằng bên phát hành sẽ dùng thông tin bí mật cục bộ trong việc sinh và bên kia thẩm tra coookie.Mờu chốt của đòi hỏi này bên phát hành không cần sao chép các cookie của nó nhưng có thể xác nhận một báo nhận cookie khi cần
* Phương pháp sinh và xác thực cookie phải nhanh để chống lại tấn công phá hủy tài nguyên vi xử lý.

Phương pháp được đề nghị để tạo cookie là dùng một hàm hash nhanh (như MD5) lên các địa chi IP nguồn và đích, các cổng UDP nguồn và đích và các giá trị bí mật được sinh cục bộ.

Oakley cung cấp các nhóm khác nhau cho too đổi khóa Diffie- Hellman. Mỗi nhóm bao gồm việc đinh nghĩa khai tham số chung và định danh của thuật toán.

Các tham số chung bao gồm số nguyên tố và phần tử nguyên thủy.

Oakley dùng nonce để chống lại tấn công replay. Mỗi nonce là một số giả ngẫu nhiên được sinh cục bộ. Các nonce xuất hiện trong các đáp ứng và được mã.

Ba phương pháp xác thực khác nhau có thể dùng vói Oakley:

* Các chữ ký số: việc trao đổi được xác thực bằng việc ký một giá trị hash mã hai bên có thể tạo ra.
* Mã khóa công khai: Việc trao đổi được xác thực bằng việc mã các tham số như các định danh và các nonce vói khóa bí mật của người gửi.
* Mã khóa đối xứng: một khóa được giao bỏi một vài cơ chế dựa trên một kênh an toàn khác có thể được dùng để xác thực việc trao đổi bằng mật mã khóa đối xứng các tham số trao đổi.

b - Ví dụ trao đổi Oakley:

Trong hình 2.19 chỉ ra giao thức trao đổi khóa “aggressive”

Trong bước đầu tiên, bên khởi tạo Initiator I truyền một cookie, một nhokm được dùng và khóa Diffie-Hellman công cộng của I để phục vụ cho việc trao đổi khóa.

I cũng chỉ ra mã khóa công khai, hàm hash và thuật tóan xác thực được đề nghị để dùng trong việc trao đổi này.Thông báo cũng bao gồm các đinh danh của I và bên đáp ứng responder R, giá trị nonce của I cho việc hao đổi.

Cuối cùng I nối chữ ký dùng khóa bí mật của I ký hai định danh, giá trị nonce, nhóm, khóa công khai Diffie-Hellman và các thuật tóan đươc đề nghị.

Khi R nhận được thông báo, R xác nhận chữ ký số dùng khóa ký công khai của I, R báo nhận cho I bằng cách gửi lại cookie, đinh danh và nonce của I, cũng như thông tin về nhóm. R cũng gửi kèm trong thôngbáo một cookie, khóa công khai Diffie-Hellman của R,các thuật tóan được chọn (nó phải trong các thuật tóan đề nghị), định danh của R, nonce của R cho phiên hao đổi này. Cuối cùng R nối chữ ký dùng khóa bí mật của R để ký hai định danh, hai nonce, thông tin nhóm, hai khóa công khai Diffie-Hellman và các thuật tóan được lựa chọn.

Khi I nhận thông báo thứ hai, I xác nhận chữ ký dùng khóa công khai của R.Các giá trị nonce trong thông báo đảm bảo rằng không có việc truyền lại (replay) một thông báo cũ. Để hoàn thành phiên trao đổi, I phải gửi một thông báo trở lại R để xác nhận rằng I đã nhận khóa công khai của R.

I R :O[Ylr0K\_KEYx.GRP;/m\0;MDPiDIJD?.:NIẬa[ID1 II nx Il N, Il GEP II g[[2]](#footnote-2) [[3]](#footnote-3) IIEHAO]

R -» I :CKY,mVOK\_KI\T.œPg-m\S^WJD-JDIK.J\]i5v,[E)?ll n>. Il N-II rç II ŒPII gĩ' Il g II EHANS]

I -> R ;CKYl5CKY3l5OK\_EEYXaGRP:gIIEHAS^nDPín>IJDHJNAẰEI[IĐ1 II nx.ll rçil N\* IIGRPII g%’ÏIEHANS]

Các kv Mên:

I=ĩiỉĩtiaror :BẾn khởi tao

R-Responder :Bíiiđap ứng

CKYj-CKY-:Cookié của bèn kha. tạo va bèn đáp ứng

OK\_KEYX:kiéu thong báo tao dõi khóa

GRPTẽn của nhổm Diữle-Hẩlỉran cho phÌẾe tao dổi

gr. ữ\*: Kbổa cỏũữ khai cùa bên khởi tao vã btiidápứim;

alĩ = khóa pMéxỉnhận đưạ

EHAOjEHAS:các hầm xác thục. hầm ha ảỊ hám mã dưọc dc nglìị và lm chọn MDPichi ra. lầng mã hóa không đooc dừng cho phin còn lá cùa thòng báo náV.

IDj-JD\*: đinh danh cho khởi tao vã đáp ứng.

NjNr: ậa tá noncẽ ngầu. rửìiỀn đưổC cuna cấp bôi bén khởi tao vầbỀn d áp ứng cho lâng trao đôi này. S^p^-Sy,{X]: Chữ kí' tỉén XdihìE khóa kv (khóabí mât) của ben kiwi tao. tón dáp ứng.

Hình 2.16 : Ví dụ về trao đổi khoá Aggressive Oakley

2.1.52 - ISAKMPịInternet Securỉty Association and Key Management Protocol).

ISAKMP định nghĩa các thủ tục và các dạng góỉ để thiết lập, thỏa thuận,thay đổi và xốa các liên kết an toàn. Là một phần của thiết lập SA, ISAKMP định nghĩa các khối (payload) cho việc trao đổi để sinh khóa và dữ liệu xác thực.Các dạng payload này cung cấp một khuôn mẫu (framework) độc lập với giao thức trao đổi khóa, thuật tóan mã hóa và cơ chế xác thực.

1. - Dạng header của ISAKMP:

Một thông báo ISAKMP bao gồm ISAKMP header được theo sau bỏd một hoặc nhiều payload. Tất cả thông báo này được chuyển trong giao thức vận tải. Một yêu cầu cho cài đặt là phải dùng ƯDP cho giao thức tầng vận tải.

Ctofcis fchài tai (ĩttỉtiaiQrGQQkìì)

Cookie dap u'Dig (Res ponder Cookie)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nexl .Pdjrload | BAJVei | MnVei | ẼKchainge tppe | Flags |

M«s»ge ID

Lénasri

M\*xt Pjij(l04d

(a) ISAKMP header

RESERVED

Pjylojd Lf.-r.9iN

(b) Generic Payload header

Hình 2.17 :Cấc dangISAKMP

Hình 2.20 chỉ ra dạng header của một thông báo ISAKMP. Nố bao gổm các trường sau:

* Intiator Cookie (64bits):Cookie của bên khởi tạo thiết lập SA, khai báo SA và xóa SA.
* Responder Cookie (64 bits):Cookie của bên đáp ứng, null ưong thông báo đầu tiên từ bên khởi tạo.
* Next payload (8bỉts):Chỉ ra kiểu của payload đầu tiên trong thông báo
* Major Version (4bits):chỉ ra version chính của ISAKMP đang dùng
* Minor Version (4 bits):Chỉ ra version phụ đang dùng
* Exchange type (8 bits):ơủ ra kiểu của trao đổi
* Flags (8 bits):Chỉ ra một tập các lựa chọn cụ thể cho lận ưao đổi ISAKMP này. Hai bit đã được định nghĩa: bit mã (encryption) được xác lạp nếu tất cả các payload theo sau header được mã dùng thuật toán mã cho SA. Bit cam kết (commit) được dùng để đảm bảo rằng tài liệu mã không được nhận trước khỉ hoàn thành việc thiết lập SA.
* Message ID (32 bits): đỉnh danh duy nhất của thông báo này.
* Length (32 bỉts):độ dài của thông báo tổng (header cộng tất cả các payload) trong octest.

1. - Cắc kiểu ISAKMPpayload:

Tất cả các ISAKMP payload bắt đầu vớỉ cùng payload header chung được chỉ ra trong hình 14b.

Trường Next payload có giá trị 0 nếu nó là trường cuối cùng ừong thông báo, ngược lại giá tậ của nó là kiểu của payload tiếp theo. Trường độ dài payload (Payload length) chỉ ra độ dài trong octets cùa payload này, bao gồm payload header chung.

SA payload (SA):

Được dùng để bắt đầu thiết lập một SA. Trong payload này. Trong payload này tham số DOI (Domain of Interpretation) chỉ ra DOI mà thỏa thuận sử dụng. IPSec DOI là một ví dụ nhưng ISAKMP có thể được dùng trong phạm vi khác. Tham số Situation đinh nghĩa chính sách an toàn cho thỏa thuận này, thực chất là các mức an toàn được đòi hỏi cho mã hóa và độ bí mật được xác đinh.

Payload đề nghị (proposal payload - P)

Chứa thông tin được dùng trong thỏa thuận SA. Payload này bao gồm giao thức cho SA này (ESP hoặc AH) , các dịch vụ và các cơ chế được thỏa thuận cho nó. Payload cũng bao gồm giá trị SPI của bên gửi và số các transform. Mỗi transform được chứa ừong một transform payload. Việc dùng nhiều transfrom cho phép bên khởi tạo đề nghị một vài khả năng, bên đáp ứng phải chọn một trong các khẳ năng đó hoặc không chấp nhận đề nghị. Transform payload (T):

Định nghĩa một transform an toàn được dùng để an toàn kênh truyền thông cho giao thức được gán. Tham số transform # dduwowvc dùng để định danh payload đặc trưng mà bên đáp ứng có thể chỉ ra sự chấp nhận transform này. Các trường Transform-ID và Attributes xác đinh một transform cụ thể (ví dụ 3DES for ESP, HMA-SHA-1-96 for AH) vói các thuộc tính liên kết như độ dài hash.

Key Exchange payload (payload trao đổi khóa- KE)

Có thể được dùng cho một vài kỹ thuật trao đổi khóa bao gồm trao đổi khóa Oakley, DIFFIE-HELLMAN , RSA-based được dùng bởi PGP.Trường dữ liệu trao đổi khóa (Key Exchange data ) chứa dữ liệu được đòi hỏi để sinh khóa phiên và phụ thuộc vào thuật tóan trao đổi khóa được dùng.

Identification payload (ID):

Được dùng để xác định định danh của các bên truyền thông và có thể được dùng cho xác thực thông tin. Trường ID data còn chứa một địa chỉ Ipv4.

Certificate payload (CERT):

Chuyển giao một public-key certificate. Trường certificate encoding chỉ ra kiểu certificate hoặc thông tin có quan hệ vói certificate. Nó có thể bao gồm:

* PGP certificate
* DNS signed key
* X.509 certificate-signature
* X.509 certificate-key exchange
* Kerboros tokens
* Certificate Revocation List (CRL)
* Authority Revocation List (ARL)
* SPKI certificate

Điểm mấu chốt trong trao đổi ISAKMP là người gửi có thể bao gồm certificate Request payload để đòi hỏi một certificate của bên truyền thông. Payload có thể liệt kê nhiều kiểu certificate có thể được chấp nhận.

Hash payload (HASH):

Chứa dữ liệu được sinh bởi một hàm HASH được tính ữên một vài phần của thông báo hoặc trạng thái ISAKMP . Payload này có thể được dùng để xác nhận tính toàn vẹn của dữ liêu trong thôngbáo hoặc xác thực các bên thỏa thuận.

Signature payload (SIG):

Chứa dữ liệu được sinh ra bởi hàm chữ ký số tính trên một vài phần của thông báo và trạng thái ISAKMP. Payload này có thể được dùng để xác nhận tính toàn vẹn của dữ liệu trong một thông báo và có thể dùng cho các dịch vụ không từ chối (nonrepudiation services)

Nonce payload : Chứa dữ liêu ngẫu nhiên được dùng để chống lại kiểu tấn công replay.

Notification payload: Chứa lỗi hoặc thông tin trạng thái được liên

kết với SA này hoặc thỏa thuận SA này.

Một số thông báo lỗi như:

* Invalid payload type
* DOI not supported
* Situation not supported
* Invalid Cookie
* Invalid SPI
* Invalid trasform IP
* No Proposal Chosen

Chỉ có một thông báo trạng thái ISAKMP được định nghĩa là Connected, cùng với sự thông báo ISAKMP này, các thông báo DOI cũng được dùng. Với IPSec , các thông báo trạng thái sau được đinh nghĩa:

Responder-lifetime: Thời gian sống của SA được chọn bỏi bên đáp ứng.

Replay-Status: Được dùng cho việc khẳng đinh của bên đáp ứng là có tiến hành phát hiện tấn công replay hay không.

Intial-Contact: Báo cho bên kia biết rằng đây là SA đầu tiên được thiết lập với hệ thống từ xa. Người nhận thống báo này có thể xóa các SA bất kỳ đang tồn tại.

Delete payload: Chỉ ra một hoặc nhiều SA mà người gửi đã xóa từ cơ sở dữ liệu của nó.

1. - Các trao đổi ISAKMP.

ISAKMP cung cấp một khuôn mẫu (framework) cho việc trao đổi thông báo với các kiểu payload. Có 5 kiểu trao đổi ngầm đinh được cung cấp, chúng được tóm tắt trong bảng 3 dưới đây.

Trong bảng này, SA tham khảo tói một SA payload vói các payload protocol và transform.

a - Trao đổi cơ sở (Base exchange):

Cho phép tài liệu trao đổi khóa và xác thực được truyền với nhau. Nó tối thiểu số ứao đổi, không hỗ ứợ việc bảo vệ đinh danh. Hai thông báo đầu tiên cung cấp các cookie và thiết lập một SA với các transform và giao thức đã thỏa thuận.Cả hai bên dùng một số nonce để chống lại tấn công replay. Hai thông báo cuối cùng trao đổi tài liệu khóa, đinh danh ngưòi dùng vói payload xác thực Auth được dùng để xác thực các khóa, các đinh danh, và các nonce từ hai thông báo đầu tiên.

b - Identity Protection Exchange :

Mở rộng “Trao đổi cơ sở” để bảo vệ các đinh danh của người dùng.Hai thông báo đầu tiên thiết lập SA. Hai thông báo tiếp theo tiến hành trao đổi khóa, với các nonce để chống lại tấn công replay.Khi khóa phiên đã được tính, hai bên trao đổi các thông báo được mã chứa thông tin xác thực như các chữ ký số và các certificate xác nhận khóa công khai.

c - Authentication Only Exchange:

Được dùng để tiến hành xác thực bằng tay không cần trao đổi khóa. Hai thông báo đầu tiên thiết lập SA. Người đáp ứng dùng thông báo thứ hai để chuyển ID của nó và dùng xác thực để bảo vệ thông báo. Người khởi tạo gửi thông báo thứ 3 để chuyển ID được xác thực của nó.

d - Aggressive Exchange :

Tối thiểu số trao đổi bằng cách không bảo vệ định danh. Trong thông báo đầu tiên người khởi tạo đưa ra một SA vói giao thức và các lựa chọn transform đã được đề nghị. Người khởi tạo cũng bắt đầu việc trao đổi khóa và cung cấp định danh của nó. Trong thông báo thứ hai người đáp ứng chỉ ra sự chấp nhận của nó đối với SA bao gồm giao thức và transform đặc trưng, hoamn thành việc trao đổi khóa và xác thực các thông tin được truyền. Trong thông báo thứ 3 người khởi tạo truyền kết quả xác thực tính trên thông tin trước đó được mã dùng khóa bí mật chia sẻ.

Information Exchange được dùng cho việc chuyển giao thông tin quản lý SA.

|  |  |
| --- | --- |
| (a) Base Exchange | |
| 1. 1 —» R :SA; nonce 2. R -> I : SA; NONCE 3. I —» R :KE; ID^ AUTH 4. R->I :KE;IDR;AUTH | Bắt đầu thỏa thuận ISAKMP SA SA cơ sở được thỏa thuận Khóa được sinh. Định danh người khỏi tạo được xác nhận bỏi người đáp ứng Định danh người đáp ứng được xác nhận bởi người khỏi tạo. Khóa được sinh; SA được thiết lập |
| (b)Idem | tify Protection Exchange |
| 1. 1 —» R :SA 2. R->I:SA 3. I —» R :KE;nonce 4. R—»I :KE;nonce 5. \* I R TDịỉAUTH 6. R ->• I : IDR;AUTH | Bắt đầu thỏa thuận ISAKMP SA SA cơ sở được thỏa thuận Khóa được sinh.  Khóa được sinh  Định danh người khởi tạo được xác nhận bởi người đáp ứng  Định danh người đáp ứng được xác nhận bởi người khởi tạo; SA được thiết lập |
| (c) Auứientication Only Exchange | |
| 1. 1 —> R :SA; nonce 2. R -» I : SA; NONCE IDr;AUTH   (3)IR :IDj; AUTH | Bắt đầu thỏa thuận ISAKMP SA SA cơ sở được thỏa thuận Khóa được sinh. Định danh người đáp ứngđược xác nhận bởi người khởi tạo Định danh người khỏi tạo được xác nhận bải người đáp ứng. |
| (d) Aggressive Exchange | |
| 1. 1 R :SA;KE; nonce; IDj 2. R I : SA;KE;NONCE IDr;AUTH   (3)\* I —» R : AUTH | Bắt đầu thỏa thuận ISAKMP SA và trao đổi khóa  Định danh người khỏi tạo được xác nhận bởi người đáp ứng. Khóa được sinh.SA cơ sở được thỏa thuận Định danh người đáp ứng được xác nhận bởi người khởi tạo. SA được thiết lập. |
| (e) Informational Exchange | |
| (1)\* I -> R :N/D | Thông báo lỗi, trạng thái hoặc xóa |

Bảng 2.3 : Các kiểu trao đổi ISAKMP

Các ký hiệu:

I=Người khởi tạo R=Người đáp ứng \*=payload mã sau ISAKMP

1. - IKE (Internet Key Exchange).
2. — Giới thiệu về IKE.

ISAKMP không xác định một khóa trao đổi. Nó là để cho các giao thức khác. Cho IPSec cần xác định khóa trao đổi là IKE - Trao đổi khóa trên mạng công cộng. IKE sử dụng ngôn ngữ của ISAKMP để xác định khóa trao đổi và cách để điều đình các dịch vụ an ninh. IKE thực sự xác định số lượng các trao đổi và các tùy chọn có thể được áp dụng cho trao đổi. Ket quả của trao đổi IKE là xác thực được khóa và đồng ý - nhờ dịch vụ an ninh - nói một cách khác, một hiệp hội an ninh IPSec.

1. - Sự trao đổi IKE (IKE Exchanges).

Hai chế độ trao đổi exchanges—main và Aggressive có cùng một điều : Thành lập sử bảo mật và xác thực của một kênh thông tin liên lạc (IKE SA) và xác thực khóa (key) sử dụng để cung cấp sự bảo mật, tích họp thông báo, và xác thực nguồn thông báo tới thông tin liên lạc IKE giữa hai sự tương đương. Tất cả những sự trao đổi khác được định nghĩa trong IKE có một sự xác thực IKE SA như là một điều kiện tiên quyết. Bởi vậy nó là một bước của sự trao đổi, hoặc chế độ chính (main) hoặc chế độ Aggressive, cần phải được thực hiện trước khi có bất kỳ một trao đổi khác.

1. - Chế độ trao đổi chính của IKE (Main Mode Exchange).

Chế độ Main sử dụng sáu thông báo, trong ba vòng truyền đi, để thiết lập các IKE SA. Ba bước dưới đây là trao đổi SA, một trao đổi Diffie-Hellman và một trao đổi của Nonces, và sự xác thực của tương đương.

1. — Chế độ trao đổi nỗ lực (Aggressive Mode Exchange).

Mục đích của chế độ trao đổi Aggressive là giống như chế độ trao đổi main - việc thành lập một hiệp hội an ninh xác thực, và những khóa (key), mà IKE có thể được sử dụng để thiết lập các hiệp hội an ninh bảo vệ an ninh cho các giao thức. Sự khác biệt chính là chế độ Aggressive chỉ dùng bằng một nửa số thông báo như ở chế độ main. Bằng cách hạn chế số lượng thông báo, chế độ Aggressive cũng giới hạn quyền trao đổi của nó và cũng không cung cấp bảo vệ danh tính (identity).

1. — Chế độ trao đổi nhanh ị Quick Mode Exchange).

Sau khi một IKE SA được thành lập, thông qua chế độ Main hoặc Aggressive, nó có thể được sử dụng để tạo ra SAs cho an ninh các giao thức khác như IPSec. Các SAs được thành lập thông qua chế độ trao đổi nhanh.

Chế độ trao đổi nhanh được thực hiện dưói sự bảo về của một IKE SA được thành lập trước đó. Nhiều chế độ nhanh có thể được thực hiện cùng với một chế độ main hoặc Aggressive. Trong thực tế, việc thực hiện nhiều chế độ nhanh có thể được thực hiện dưới sự bảo vệ của một IKE SA.

2.2.ố - Chế độ trao đổi khác (Other IKE Exchanges).

Tất cả những trao đổi được mô tả trong IKE cho đến nay đều tạo ra SAs - Đó là một trong những giai đoạn tạo ra IKE SA và hai giai đoạn tạo ra IPSec SAs. Hai định nghĩa IKE ứao đổi khác cung cấp cho sự bảo trợ của IKE SA và cho sự điều đình của các nhóm riêng tư Diffie-Hellman.

Chương 3 : ứng dụng của IPSec trong bảo mật thông tin trên

mạng.

1. - Tìm hiểu về bộ phần mềm OpenS/wan.
2. — OpenS/wan.

OpenS/wan là mã nguồn mở dùng để triển khai IPSec trên hệ điều hành Linux. Nó là một nhánh mở của dự án FreeS/WAN, bắt đầu bằng vài người phát trien thất bại với việc chính trị bao quanh dự án đó. OpenS/wan là, không có câu hỏi, dễ dàng với tất cả các giải pháp để VPN hoạt động trên Linux; nhưng không phải nói nhiều về vấn đề đó, các giải pháp khác có thể là sai lầm.

OpenS/wan là một mã nguồn mở nên bất cứ ai cũng có thể trở thành người phát triển. Họ có thể vá các lỗi, sửa đổi và bổ sung chức năng mới.

OpenS/wan hỗ trợ cho hầu hết các mở rộng (RFC + Bản thảo IETF) có liên quan đến IPSec, bao gồm cả kỹ thuật chứng nhận số X.509, NAT Traversal, và rất nhiều thứ khác.

1. - Chức năng của OpenSIwan.

Thiết lập các hệ thống mạng an toàn (mạng riêng ảo) trên nền mạng công cộng theo công nghệ IPSec.

1. - Cấu trúc của OpenSIwan.

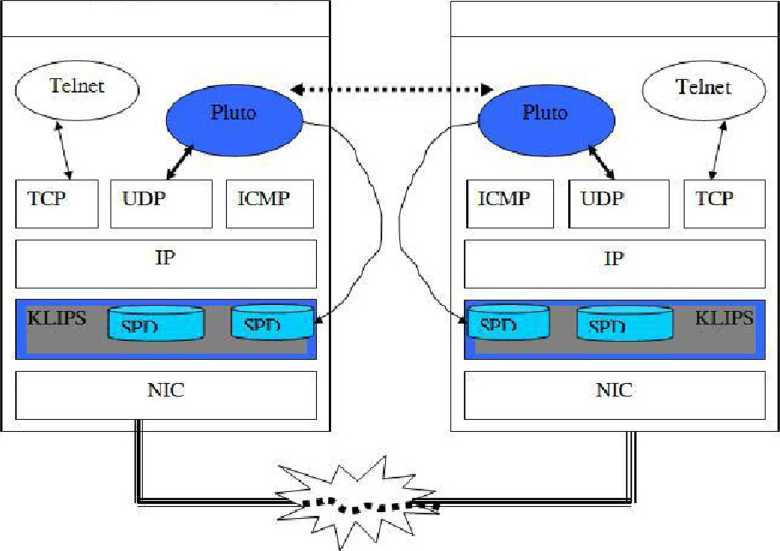
OpenS/wan bao gồm hai phần chính cấu thành là KLIP và PLUTO. PLUTO có chức năng : Trao đổi, thiết lập kênh an toàn (IKE, ISAKMP,...)

KLIP có chức năng : Thực hiện việc mã húa các gói IP theo công nghệ

IPSec.

1. —PLUTO

Cơ chế hoạt động của PLUTO trong OpenS/wan.



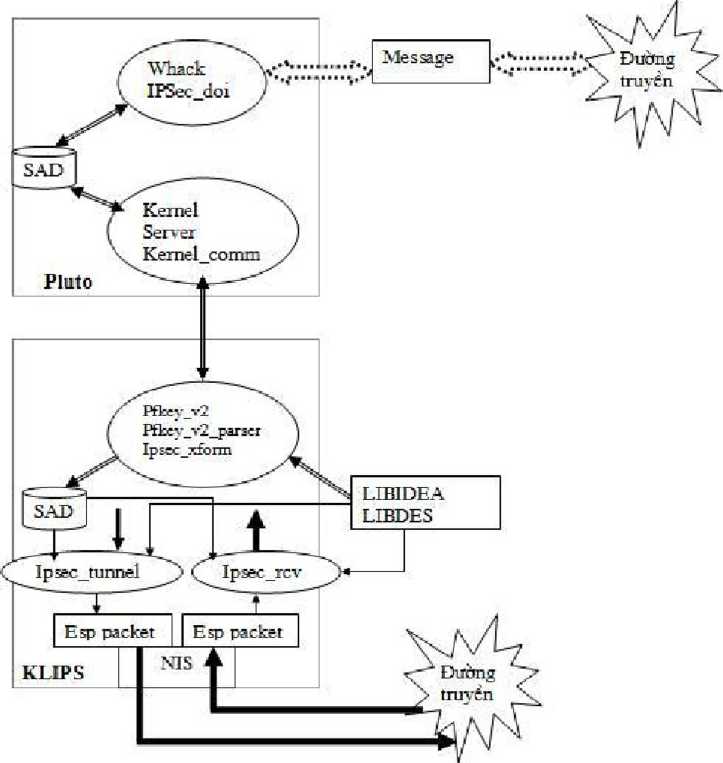
Hình 3.1 : Mô hình hoạt động của PLUTO Các module chương trình chính ưong PLUTO và KLIP

|  |  |
| --- | --- |
| Tên module | Chức năng |
| Spdc.h  spdc.c | Xây dựng các cơ sở của chính sách an toàn ( ví dụ : các cấu trúc của các transform, thuật toán xác thực, mã hoá, nhóm của Diffie - Heilman ...) |
| Ipsec\_doi.il  Ipsec\_doi.c | Thực hiện việc trao đổi các thông tin vái các gateway khác để thực hiện các quá trình thoả thuận các SA an toàn, trong module này, nó cũng thực hiện việc xác thực các gateway mã hoá bằng việc sử dụng khoá chian sẻ trước hay hệ mật khoá công khai. |
| Preshared.h Preshared, c | Thực hiện việc lấy các giá trị khoá của RSA đặt trong /etc/ipsec.conf và /etc/ipsec.secrets để phục vụ cho việc mã hoá, giải mã và xác thực |
| Crypto.h  Crypto.c | Mã và giải mã các message của pluto ừong quả trình thoả thuận các SA an toàn. Thuật toán mã hoá ở đây sử dụng DES hoặc IDEA ( giá trị default là IDEA) |
| Cookie.h  Cookie.c | Sinh ra các cookie |
| Connection.h  Connection.c | Sử lý các thông tin về các kết nối |
| KemeLh  Kemel.c | Thực hiện việc trao đổi các thông tin giữa pluto và klips |

|  |  |
| --- | --- |
| Server.h Server, c  Kemel\_comm.h  Kemel\_comm.c |  |
| Whack.h  Whack.c  Kerkel\_comm.h  Kerkel\_comm.c | Giao diện để thực hiện các lệnh của pluto(whack) và thực hiên các tiến trình của whack. |
| Constants.h | Lưu và định nghĩa các hằng, bao gồm cả số Diffie - Heilman. |
| Md5.h  Md5.c | Tạo các giá trị tóm lược của hàm băm theo giải thuật md5 |
| shal.h  shal.c | Tạo các giá trị tóm lược của hàm băm theo giải thuật shal |
| Config.in | Lưu giữ các thông tin cấu hình của IPSec để add vào kernel |
| Defconfig | Đặt các giá trị ngầm đinh của IPSec vào kernel |
| Ipsec\_md5.h  Ipsec\_md5.c  Ipsec\_shal.h  Ipsec\_shal.c | Thực hiện các hàm băm md5 và shal |
| pfkey\_v2.h pfkey\_v2.c pfkey v2 parse r.h  pfkey\_v2\_parse  r.c | Thực hiện việc trao đổi thông tin giữa Pluto và Klips |
| ipsec\_xform.h  ipsec\_xform.c | Kết hợp với các thư viện DES, IDEA và các thông tin đã thoả thuận được của Pluto chuyển xuống để tạo ra các khoá an toàn cho phiên liên lạc. |
| ipsec\_rcv.h  ipsec\_rcv.c | Thực hiện việc giải mã các gói tin từ ngoài tói. |
| ipsec\_tunnel.h  ipsec\_tunnel.c | Mã và đóng gói các gói tin rồi gửi ra đường mạng. |

Bảng 3.1 : Các module chính trong PLUTO và KL1P

Sơ đồ hoạt động cùa các module :



Hình 3,2 : Sơ đồ hoạt động của cắc module.

1. KUP

KLIP là lèn đầu tiên có trong IPSec Stack cho Linux. Phiên bản đầu tiên chạy trên Linux 2.0, và phiên bản mới nhất chạy trên bất cứ cái nào giữa 2.2 và 2.6. Nó được sử dụng duy nhất cho Linux IPSec Stack trong hơn một năm qua; nhưng trong thực tế, nố đố được sử dụng hơn năm năm. Nố cố uy tín cao trong cộng đồng IPSec, và nó được xem như là nền tảng của de facto interop bởi nhiều nhà cung cấp thương mại. Neu KLIP không sử dụng được với phần cứng IPSec của chúng ta, thì chúng ta đó làm sai cái gì đó.

KLIP được viết lại giữa phiên bản Free »s/wAN 1.99 và phiên bản 2.X. Một số chức năng của nó đó được phát triển mạnh đến nay và được re­factored. Các mã 2.X cũng giới thiệu hệ thống kiểm tra regression. Mỗi một

5

6

tính năng của KLIP đều được kiểm tra trong các bài kiểm tra nightly regression. Các phiên bản 2.X còn là cơ sở cho OpenS/wan 2.X.

1. - ipsecXInterfaces

Ke từ khi KLIP cú sẵn các mã netfilter trong nhân Linux, nó đá tìm ra một cách mới để móc nối vào các nhân và các mạng lưới Stack. Giải pháp tạo ra các thiết bị ảo, các thiết bị ipsecX, và áp dụng một mánh khóe định tuyến để gửi những gói tin Packet vào các thiết bị ảo. Lợi thế là lưu lượng gói tin Packet rất rõ ràng. Một mã hóa gói đi kèm trong thiết bị ethx. Nó được nhận diện đây là một gói IPSec, và nó được gửi đến các mã KLIP để xử lý. KLIP giải mã gói tin Packet, và đặt các gói tin giải mã trên thiết bị ipsecX. Như vậy các gói sẽ đi ngang qua mọi Linux iptables (hoặc ipfwadm /ipchains trên nhân cũ) một lần trên giao diện Interface, cho phép tồn tại các luật tường lủa riêng biệt cho mã hóa và giải mã gói tin. Điều này giúp viết các luật tường lửa một cách dễ dàng và được coi là một trong những tính năng lớn của KLIP.

1. - Gói Caching đầu tiên.

Một tính năng quan trọng là bộ nhớ đệm (Caching) của các gói tin cho mạng mà nó được hiểu là một đường hầm IPSec cần được tạo ra. Bởi vì trong bộ nhớ đệm, đường hầm có thể dễ dàng đưa lên và xuống mà không có bất kỳ mất mát về gói nào. Bây giờ mất một gói tin nói chung không phải là một vấn đề lớn, nhưng nếu mất vài gói tin thì sẽ mang đến sự trì hoãn đáng kể.

1. — Nghiên cứu dẫn đường MTU.

Một tính năng của KLIP là nó hỗ trợ đày đủ Path MTU discovery (RFC 1191). Path MTU discovery mô tả một phương pháp để xác định đơn vị truyền cực đại của một gói tin Packet.

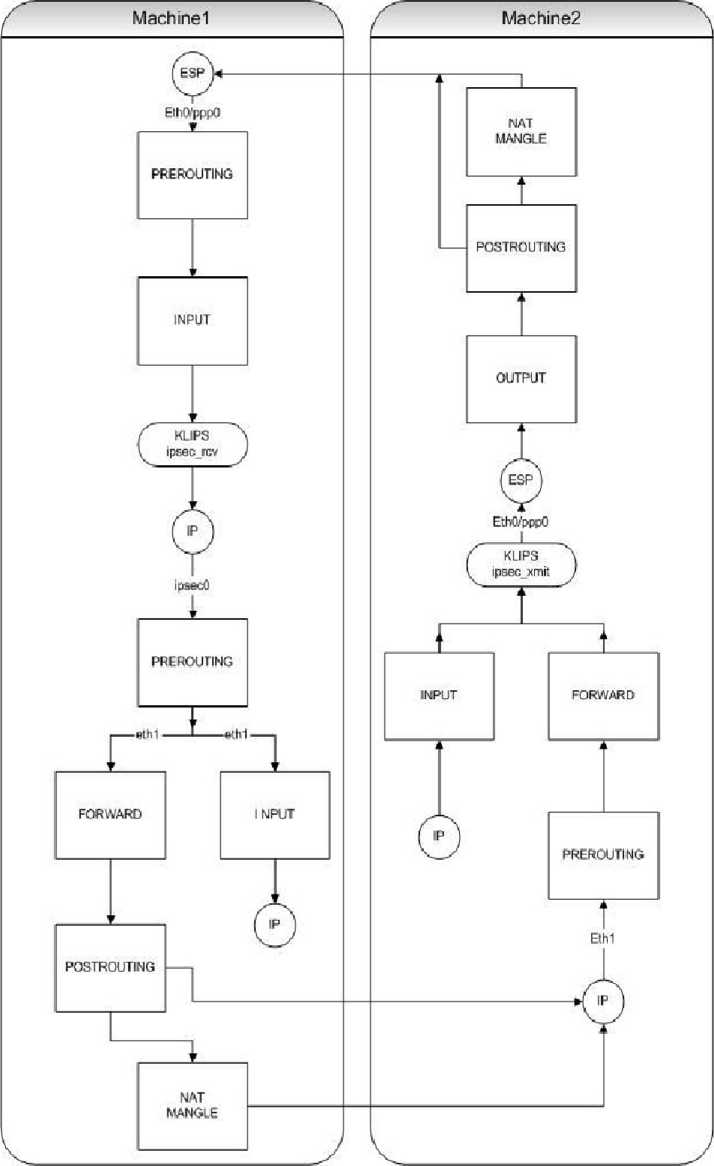
Path MTU discovery tìm ra kích thước lớn nhất các gói tin có thể được xử lý bởi các định tuyến trung gian giữa hai máy tính. Các máy tính sẽ khởi xướng gửi các gói tin nhỏ, nhưng khi nó nhận chính xác về mọi mặt của gói tin thì nó sẽ tăng kích thước gói tin lớn dần. Tại một điểm nào đó, hoặc là điểm kết thúc, hoặc một máy tính ở giữa mà đang đặt trên gói tin, sẽ có gói tin không thể gửi xa hơn được nữa vì nó quá lớn. Nó sẽ drop gói tin và gửi lại một thông báo cho các máy chủ. Đáy là một gói tin ICMP “Destination Unreachable (không thể tới được đích)“ trong đó sẽ có một thông báo nói rang "Datagram Too Big (Gúi dữ liệu quỏ lớn)". Máy tính gửi sẽ nhận được gúi tin ICMP đó, đọc giá trị 'Next-Hop MTƯ, và sử dụng gói Packet nhỏ hơn. Neu đường giữa hai máy tính thay đổi, và một hop trong chuỗi cú thể không nhận được kích thước của gói tin, cùng một quá trình sẽ được bắt đầu từ máy chủ. Và chỉ một lần trong một khoảng thời gian, máy tính sẽ tăng kích thước gói tin gửi đi và chỉ giới hạn trong trường hợp không có máy tính giới hạn ở giữa.

1. - KLIPS ’ Downside.

KLIPS không được bao gồm bong nhân Linux chính thức. Như yậy, nhiều người sẽ không thích hoặc không tin vào KLIPS. Thông thường, các quản bị hệ thống đó thử những cái gì mới, IPSec, và không muốn thực hiện điều gì khó hơn những cái họ đã có, ví dụ bằng cách thêm mã riêng và xây dựng một nhân riêng của họ. Biên soạn KILPS vào nhân của chúng ta được thực hiện một cách dễ dàng, do đó không được loại bỏ nó như một lựa chọn. Nhớ rang KLIPS đó tồn tại được hơn 10 năm và nó đã chứng minh được sự ổn định của nó. Mặt khác NETKEY là một thương hiệu mới. Nó có vẻ rất hứa hẹn, nhưng không thấy được một triển khai quy mô lớn nào được nêu ra.

1. — Klips-packetflow.

OperỉSWBTl-2 KL'P$



(C)opyright Xeleranca 2006

*Hình 3.3* : *Sơ đồ hoạt động của KLỈP*

1. - Cách cài đặt OpenS/wan

Tất cả các bản vá quan trọng, bao gồm cả X.509 và NAT Traversal. Neu muốn cài đặt từ đầu chúng ta có thể tải nó từ <http://www.OpenS/wan.org/code>, và làm theo các hướng dẫn cài đặt bao gồm trong các gói.

Chỳng ta cú hai lựa chọn cài đặt IPSec stack mà ta muốn cài đặt trong nhân. Chúng ta có thể sử dụng OpenS/wan của IPSec stack (Klips), hoặc sử dụng ữong xây dựng IPSec stack ữong nhân 2.6 (26sec). Neu ta chạy trên nhân 2.4, chỉ có tùy chọn là Klips, cần phải có bản vá NAT Traversal hỗ trợ trong nhân (Neu ta muốn sử dụng nó) , và xây dựng các module ipsec.o . Neu sử dụng nhân 2.6 hoặc nhân 2.4 với hỗ trợ backported 26sec (chẳng hạn như nhân của Debían cung cấp), ta không cần đụng chạm đến nhân - vùng ở tất cả - chúng ta có thể chỉ cần cài đặt các OpenS/wan người dùng - vùng tiện ích và đi. Lưu ý rằng nó không như là một tùy chọn để sử dụng Klips trên nhân 2.6; nó là trên người phát triển OpenS/wan tới-liệt kê, nhưng thực tế không phải là ưu tiên cao.

Chúng ta sẽ cần người dùng-vùng tiện ích. Neu cài đặt từ mã nguồn, ‘thực hiện các chương trình; thực hiện cài đặt’ sẽ giúp chúng ta có được những gì cần. Neu chạy Debían thử nghiệm hoặc không ổn định, chỉ cần chạy ‘apt-get install OpenS/wan’ để cung cấp mức tiện ích người dùng. Atrpms cung cấp một OpenS/wan trọn gói cho các phiên bản gần đây của RedHat và Fedora Core.

Sau khi đó lựa chọn và thiết lập IPSec stack và cài đặt các chương trình sử dụng cho người dùng-vùng (user-land). Chúng ta sẽ cấu hình OpenS/wan.

a - Cài đặt chứng chi trên Gateway.

Cài đặt các tập tin trong những vùng thích hợp ( Neu cài đặt vào một máy tính từ xa, cần phải chắc chắn để sao chép các tập tin một cách an toàn).

$ cp /var/sslca/host.example.com.keỵ /etc/ipsec.d/private $ cp /var/sslca/host.example.com.pem /etc/ipsec.d/certs $ cp /var/sslca/demoCA/cacert.pem /etc/ipsec.d/cacerts $ cp /var/sslca/crl.pem /etc/ipsec.d/crls/crl.pem

b — Cẩu hình OpenSIwan trên máy Gateway.

Bước 1 : Cấu hình ipsec.secrets :

/etc/ipsec.secrets should contain the following:

: RSA host.example.com.key "password"

Mật khẩu ở trên càn phải là cụm từ mật khẩu PEM mà ta đã nhập vào khi tạo ra chứng chỉ SSL.

6

0

Bước 2 : cấu hình ipsec.conf:

/etc/ipsec.conf nên cấu hình giống như cấu hình ở dưới (lưu ý rằng các indentation là rất quan trọng, không có nó, thì OpenS/wan sẽ lỗi.

version 2.0

config setup

interfaces=%defaultroute

nat\_traversal=yes

virtual\_private=%v4:10.0.0.0/8,%v4:172.16.0.0/12,%v4:192.168.

0**.**0/16

conn %default

keyingtries=l

compress=yes

disablearrivalcheck=no

authby=rsasig

leftrsasigkey=%cert

rightrsasigkey=%cert

conn roadwarrior-net

leftsubnet=(your\_subnet)/(your\_netmask) also=roadwarrior

conn roadwarrior

left=%defaultroute leftcert=host.example.com.pern right=%any

rightsubnet=vhost:%no,%priv

auto=add

pfs=yes

conn block

auto=ignore

conn private

auto=ignore

conn private-or-clear auto=ignore

conn clear-or-private auto=ignore

conn clear

auto=ignore

conn packetdefault auto=ignore

Cấu hình này sẽ thiết lập cho bất kỳ ai có được chứng chỉ hợp lệ có chữ ký của CA sẽ có thể kết nối được đến máy chủ của chúng ta. Có hai hồ sơ kết

nối : một cho kết nối trực tiếp tới Gateway, và một cho khách hàng kết nối đến các mạng lưới phía sau gateway, cấu hình này cũng bao gồm cấu hình cho NAT Traversal sẽ cho phép bất kỳ ai ở một host đằng sau NAT gateway sử dụng các địa chỉ riêng RFC1918 (được định nghĩa trong hàng ‘virtual\_private’) để kết nối. Tất cả các mục 'auto=ignore' sẽ được sử dụng để vô hiệu hóa Opportunistic Encryption (OE), vì nó có thể gây ra vấn đề nếu không được cấu hình đúng.

Neu kết nối từ windows tới máy chủ sử dụng L2TP trờn IPSec, sẽ cần những kết nối sau đây, ở noi nào đó trên định nghĩa 'roadwarrior' :

conn roadwarrior-12tp pfs=no

leftprotoport=17/0 rightprotoport=l7/1701 also=roadwarrior

conn roadwarrior-12tp-updatedwin pfs=no

leftprotoport=17/1701 rightprotoport=l7/1701 also=roadwarrior

Ngoài ra, nếu muốn tất cả lưu lượng truy cập của khách hàng đều thông

> rr >

qua giao thức IPSec, thì cân phải có một kêt nôi cho phép điêu đó.

conn roadwarrior-all

leftsubnet=0.0.0.0/0 also=roadwarrior

c - Client Setup: OpenS/wan.

Bước 1 : Thực hiện theo các bước dưới “Tạo một chứng chỉ (Generating a Certificate)” để tạo ra một chứng chỉ mới cho máy tính của khách hàng, sửa đổi tên tập tin và khi cần thiết, (ví dụ tên tập tin cho khách hàng này là ‘clienthost.example.com’).

Bước 2 : Sao chép các tập tin sau (trong một thời gian an toan) trờn máy khách hàng:

host. example. com.pern (gateway cấp chứng chỉ của tập tin)

clienthost.example.com.key clienthost.example.com.pern cacert.pern crl.pern

Bước 3 : Sao chép các tệp tin vào đúng vị trí:

$ cp clienthost.example.com.key /etc/ipsec.d/private $ cp clienthost.example.com.pem /etc/ipsec.d/certs $ cp host.example.com.pem /etc/ipsec.d/certs $ cp crl.pem /etc/ipsec.d/crls

$ cp cacert.pem /etc/ipsec.d/cacerts/cacert.pern

Bước 4 : cấu hình IPSec :

ipsec.secrets:

: RSA clienthost.example.com.key "password" ipsec.conf: version 2 config setup

interfaces=%defaultroute

nat\_traversal=yes

conn %default

keyingtries=l

compress=yes

authby=rsasig

leftrsasigkey=%cert

rightrsasigkey=%cert

conn roadwarrior-net

leftsubnet=(your\_subnet)/(your\_netmask) also=roadwarrior

conn roadwarrior

left=(ip.of.host) leftcert=host.example.com.pern right=%defaultroute

rightcert=clienthost.example.com.pern

auto=add

pfs=yes

conn block

auto=ignore

conn private

auto=ignore

conn private-or-clear  
auto=ignore

conn clear-or-private  
auto=ignore

conn clear

auto=ignore

conn packetdefault

auto=ignore

Bước 5 : Bắt đầu các liên kết VPN, và chắc chắn rằng mọi thứ hoạt

động:

# /etc/init.d/ipsec restart $ ipsec auto --up roadwarrior $ ipsec auto --up roadwarrior-net

Bước 6 : Nếu muốn có các kết nối bắt đầu tự động, thay đổi ‘auto=add’ thành ‘auto=start\

d - Client Setup: Windows 2000/XP.

Chỳ ý : Neu như trước đó đã cài đặt SSH Sentinel, và muốn sử dụng xây dựng IPSec stack trong windows, cần phải bỏ cài đặt (hoặc vụ hiệu hóa) SSH Sentinel, và kích hoạt dịch vụ ‘ipsec’. Điều này cũng áp dụng đối với khách hàng nào khác có thể đã được cài đặt IPSec - \*cần\* chắc chắn rằng đã vô hiệu hóa trước khi sử dụng các dịch vụ xây dựng trong IPSec.

Bước 1 : Tạo các chứng chỉ, làm theo các bước ‘Tạo một chứng chỉ (Generating a Certificate)’. Giả sử gọi chứng chỉ của Windows 2000 là 'winhost.example.com'. cần phải làm theo các hướng dẫn để tạo ra file .pl2. Ngoài ra, tạo cho nó một ghi chú :

$ openssl x509 -in demoCA/cacert.pem -noout -subject

Điều này cần cho cấu hình VPN.

Bước 2 : Sao chép các tập tin này về máy tính Windows trong secure fashion, ví dụ như 'scp' hoặc một đĩa mềm. Không sử dụng FTP.

Bước 3 : Tải tiện ích Marcus Miller của IPSec từ

<http://vpn.ebootis.deand> và giải nén nó vào trong một thư mục Windows (ví dụ C:\IPSec)

Bước 4 : Tạo IPSEC + Certificates MMC.

* Vào Start/Run, gừ MMC
* Trong File (hoặc Console) - Add/Remove Snap-in
* Click vào ‘Add’
* Click vào 'Certificates', rồi chọn ‘Add’
* Chọn 'Computer Account' và nhấn ‘Next’
* Chọn 'Local computer' và nhấn 'Finish'
* Click trên 'IP Security Policy Management' và nhấn ‘Add’
* Chọn 'Local computer' và nhấn 'Finish'
* Click ‘Close’ sau đó nhấn ‘OK’

Bước 5 : Thêm chứng chỉ.

* Click mũi tên cộng của 'Certificates (Local Computer)'
* Click chuột phải vào 'Personal' và click 'All Tasks' sau đó chọn 'Import'
* Nhấn ‘Next’
* Gõ đường dẫn đến file .pl2 (hoặc nhấn ‘browse’ và chọn đường dẫn đến file) và nhấn ‘Next’
* Gõ Password nhập vào, rồi nhấn ‘Next’
* Click vào 'Automatically select the certificate store based on the type of certificate' và nhấn ‘Next’
* Nhấn 'Finish', và chọn ‘Yes’ với bất kỳ một sự nhắc nhở nào được hiện lên.
* Thoát khỏi MMC, và lưu lại nó như một tập tin, do đó không cần phải thêm vào Snap Ins một lần nữa.

Bước 6 : Thiết lập tiện ích IPSec

Cài đặt ipsecpol.exe (Windows 2000) hoặc ipseccmd.exe (Windows XP) như được mô tả trong các tài liệu hướng dẫn cho các tiện ích IPSec. Lưu ý rằng cho windows XP SP2, cần một phiên bản mới của ipseccmd.exe.

Hiệu chỉnh ipsec.conf (trêờn cửa sổ máy tính), thay thế cho "RightCA" với đầu ra của 'openssl x509 -in demoCA/cacert.pem -noout -subject'; Định khuôn lại như ở dưới (Cần phải thay đổi dấu / , và thay đổi tên của một vài File).

conn roadwarrior left=%any

right=(ip\_of\_remote\_system)

rightca="C=US,s=state,L=Citỵ,0=ExampleCo,CN=CA,Email=host@ex

ample.com"

network=auto

auto=start

pfs=yes

conn roadwarrior-net left=%any

right=(ip\_of\_remote\_system) rightsubnet=(your\_subnet)/(ỵour\_netmask)

rightca="C=US,s=state,L=Citỵ,0=ExampleCo,CN=CA,Email=host@exa mple.com"

network=auto auto=start pfs=yes

Nếu muốn mã hóa tất cả dữ liệu đi qua đường hầm, cần phải làm :

conn roadwarrior-all left=%any

right=(ip\_of\_remote\_system) rightsubnet=[[4]](#footnote-4)

rightca="C=US,s=state,L=city,0=ExampleCo,CN=CA,Email=host@exa mple.com"

network=auto auto=start pfs=yes

Bước 7 : Bắt đầu liên kết.

Chạy lệnh 'ipsec.exe'. Dưới đây là một ví dụ đưa ra.

c: UpseOipsec

IPSec Version 2.2.4 (c) 2001,2002 Marcus Mueller Getting running Config ...

Microsoft's Windows XP identified Host name is: (local\_hostname)

No RAS connections found.

LAN IP address: (local\_ip\_address)

Setting up IPSec ...

Deactivating old policy...

Removing old policy...

Connection roadwarrior:

MyTunnel : (local\_ip\_address)

MyNet : (local\_ip\_address)/255.255.255.255

PartnerTunnel: (ip\_of\_remote\_system)

PartnerNet : (ip\_of\_remote\_system)/255.255.255.255

CA (ID) : c=us,s=state,L=City,0=ExampleCo,...

PFS : y Auto : start Auth.Mode : MD5 Rekeying : 3600S/50000K Activating policy...

Connection roadwarrior-net:

MyTunnel : (local\_ip\_address)

MyNet : (local\_ip\_address)/255.255.255.255

PartnerTunnel: (ip\_of\_remote\_system)

PartnerNet : (remote\_subnet)/(remote\_netmask)

CA (ID) : c=us,s=state,L=City,0=ExampleCo,...

PFS : y Auto : start Auth.Mode : MD5 Rekeying : 3600S/50000K Activating policy...

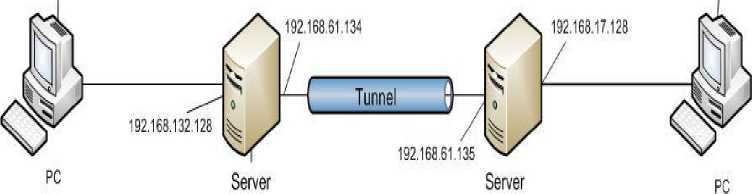
c:\ipsec>

Bây giờ, ping tới Gateway của máy, nó sẽ hiện 'Negotiating IP Security' trong một vài lần, và sau đó cung cấp một ping hồi đáp. Lưu ý rằng có thể mất một vài lần cố gắng kết nối.

1. - ứng dụng của OpenS/wan để thiết lập mạng an toàn.

3.2.1 - ứng dụng trong bảo mật mạng ỉan-to-ỉan

192.1C8.132.130 192.163.17.130



Hình 3.4 : Mô hình mạng lan-to-lan.

Site 1

* Dùng một mạng riêng có IP thuộc khoảng 192.168.132.0/24
* Có một thiết bị VPN/firewall làm default gateway với địa chỉ IP là

192.168.61.134

Site 2

* Dùng một mạng riêng có IP thuộc khoảng 192.168.17.0/24
* Có một thiết bị VPN/firewall làm default gateway với địa chỉ IP là

192.168.61.135

3.2.1.1 -Cài đặt OpenSIwan a - Cài đặt OpenS/wan từRPM

Chương trình OpenS/wan có thể được cài đặt từ mã nguồn hay tập tin RPM mà ta có thể tải về từ trang web <http://www.QpenS/wan.org>. Sau đây là các bước cài đặt:

**[root@VPNl tmp]#** rpm -ivh 0penS/wan-2**.**6**.**21.rpm

Lưu ý: nếu dựng ipsec stack có sẵn của kernel thể bạn càn phải có gói ipsec-tools được cài đặt sẵn trên máy.

b- Cài đặt OpenSIwan từ mã nguồn.

[root@VPNl tmp]# tar **-xvzf 0penS/wan-2.6.21.tar.gz** [root@VPNl tmp]# cd **0penS/wan-2**.**6**.**21**.**tar**. **gz** [root@VPNl OpenS/wan-2.6.21]# •

[rootỄVPNl 0penS/wan-2.6.21]# **make programs install**

* Cài userland và module ipsec.o:

**[rootỄVPNl 0penS/wan-2.6.21]#** make KERNELSRC=/usr/src/linux-2.6.9 programs module

**[rootỄVPNl OpenS/wan-2.6.21]#** make KERNELSRC=/usr/src/linux-2.6.9 install minstall

* Cài KLIPS (nếu chưa có) :

**[root@VPNl OpenS/wan-2.6.21]#** make KERNELSRC=/lib/modules/linux2.6.9/build module install

• Kiểm fra OpenS/wan:

[root@VPNl Opens/wan-2.6.21] # **service ipsec restart** [root@VPNl OpenS/wan-2.6.21] # **ipsec verify**

* Kiểm fra card mạng ảo:

[root@VPNl OpenS/wan-2.6.21]# **ifconfig -a**

c - Cài đặt OpenS/wan bằng lệnh yum

* Tìm các gói cài đặt của OpenS/wan

[root@VPNl ~]# yum search OpenS/wan Loaded plugins: refresh-packagekit ============Matched: Opens/wan ============

OpenS/wan.Í386 : OpenS/wan IPSEC implementation OpenS/wan-doc.Í386 : OpenS/wan IPSEC full documentation tinyca2.noarch : Simple graphical userinterface to manage a small CA

xl2tpd.i386 : Layer 2 Tunnelling Protocol Daemon (RFC 2661)

* Cài đặt các gói cài đặt của OpenS/wan:

[rootỄVPNl ~]# yum install Opens/wan Loaded plugins: refresh-packagekit Setting up Install Process Parsing package install arguments Resolving Dependencies --> Running transaction check

> Package OpenS/wan.i386 0:2.6.19-1.**fcio** set to be updated

--> Finished Dependency Resolution

Dependencies Resolved

|  |  |  |
| --- | --- | --- |
| Package | Arch | Version |
| Repository | Size |  |
| Installing : Opens/wan 1.2 M | Í386 | 2.6.19-l.fclO |
| Transaction | Summary |  |
| Install | 1 Package(s) |  |
| Update | 0 Package(s) |  |
| Remove | 0 Package(s) |  |

Total download size: 1.2 M Is this ok [y/N]: y Downloading Packages :

OpenS/wan-2.6.19-1.fcio.Í386.rpm I 1.2 MB

00:06

warning: rpmts\_HdrFromFdno: Header V3 DSA signature: NOKEY, key ID 4ebfc273

updates/gpgkey I 2.3 kB

00:00

Importing GPG key 0x4EBFC273 "Fedora (10)

<[fedora@fedoraproject.org](mailto:fedora@fedoraproject.org)>" from /etc/pki/rpm-gpg/RPM-GPG-KEY-

fedora-i38 6

Is this ok [y/N]: y

Running rpm\_check\_debug

Running Transaction Test

Finished Transaction Test

Transaction Test Succeeded

Running Transaction

Installing : OpenS/wan

1/1

Installed:

OpenS/wan.Í386 0 : 2.6.19-1.fcio Complete !

3.2.1.2 - Chạy OpenS/wan.

• Chúng ta có thể khởi chạy OpenS/wan ngay khi khởi động hệ thống bằng câu lệnh chkconf ig

[rootsVPNl ~]# cđ **/tmp/**

**[rootỄVPNl tmp]#** chkconfig ipsec on

• Bạn có thể tắt, mở, chạy lại OpenS/wan sau khi khởi động máy bằng cách chạy các script từ những câu lệnh sau:

**[rootSVPNl tmp]#** service ipsec start

SV: Thái Trung Thắng 6

9

**[rootỄVPNl tmp]#** service ipsec stop **[rootỄVPNl tmp]#** service ipsec restart

• Luôn nhớ chạy lại tiến trình Ipsec mỗi khi có sự thay đổi trong tập tin ipsec. conf để những thay đổi đó được áp dụng cho tiến trình đang chạy.

1. - Kiểm tra trạng thái cài đặt OpenSIwan.

Ngay sau khi cài đặt OpenS/wan bạn nên chạy câu lệnh ipsec verify để kiểm tra trạng thái hoạt động của chương trình.

[root@VPNl ~]# **ipsec verify**

Checking your system to see if IPsec got installed and started correctly

Version check and ipsec on-

ath [OK]

Linux OpenS/wan 2.4.6 (klips)

Checking for IPsec support in kernel [OK]

Checking for RSA private key (/etc/ipsec.secrets) [OK]

Checking that pluto is

running [OK]

Two or more interfaces found, checking IP forwarding [OK]

Checking NAT and MASQUERADEing [N/A] Checking for 'ip' command [OK]

Checking for 'iptables' command [OK]

[root0VPNl tmp]#

1. - Sửa một số lỗi trạng thái phổ biến.

Việc kiểm tea hạng thái cú thể cung cấp một số lỗi, phổ biến nhất là IP forwarding và opportunistic encryption.

• IP forwarding

Mỗi thiết bị VPN cần phải được bật chế độ routing và IP forwarding. Đe bật các chức năng này ta cần sửa nội dung tập tin

/etc/sysctl.conf.

#

* File: /etc/sysctl.conf

#

#

* Enable routing (IP forwarding)
* net.ipv4.ip\_forward = 1

Sau khi hoàn thành, dòng lệnh " s y s c 11 - p " để kích hoạt chức năng vừa thiết lập.

[root@VPNl tmp]# **sysctl -p**

net.ipv4.ip\_forward = 1 [root@VPNl tmp]#

• Opportunistic Encryption DNS Checks:

Đây là một tính năng của OpenS/wan cho phép gateway có thể mã hóa dữ liệu truyền tải, ngay cả khi những quản trị viên của hai mạng chưa liên hệ trước với nhau và mỗi hệ thống hiện chưa có thông tin về những hệ thống còn lại trong VPN. Lý do cơ bản của việc này là làm cho mọi kết nối đến server đằng sau thiết bị VPN đều được tự động mã hóa bởi Ipsec giống với cách mà mọi truy xuất HTTP được mã hóa để trở thành H iTPS.

Tính năng này nên được tắt một cách mặc định trong tập tin cấu hình và thông báo lỗi về việc này cú thể không cần quan tâm đến nữa.

32.1.5 — Các bước cấu hình VPN (Sử dụng khóa RSA)

Một trong những cách thiết lập đường hàm VPN an toàn là mã hóa dữ liệu sử dụng chứng chỉ dựa trên các khóa. Có nhiều bước mà theo đó ta có thể tạo các chứng chỉ.

Còn rất nhiều tham số khác của VPN nhưng OpenS/wan rất dễ bỏ qua khi thiết lập đường hầm. Nó sẽ tự động duyệt qua nhiều kết hợp khác nhau của IPSec và IKE thiết lập với VPN ở xa cho đến khi tìm được sự trùng khớp.

Bạn không cần phải cấu hình hầu hết các thông số này một cách tường minh giống như phải làm với các thiết bị router và fỉiewall/VPN.

Tập tin /etc/ipsec.conf

Những thông số trong tập tin cấu hình /etc/ipsec.conf

|  |  |
| --- | --- |
| Tham số | Mô tả |
| Left | Địa chỉ internet của thiết bị VPN bên trái |
| Leftsubnet | Mạng được bảo vệ bởi thiết bị VPN bên trái. |
| Leftid | Tên DNS của thiết bị VPN bên trái, đứng ngay sau kí tự Neu DNS chưa được thiết lập cho địa chỉ IP, bạn có thể bỏ qua thông số này, nếu cho giá trị không đúng sẽ làm cho quá trình khải tạo VPN bị thất bại. |
| Leftrsasigkey | Khóa RSA công khai của thiết bị VPN bên trái. Thông số này có thể xem bằng câu lệnh "ipsec showhostkey - -left". |
| Leftnexthop | Địa chỉ IP của router tiếp theo của thiết bị VPN bên trái khi nó muốn kết nối đến thiết bị VPN bên phải. Ta nên sử dụng giá trị được tự động tạo ra là “%def aultroute”, nó sẽ họp lệ trong hầu hết các trường họp, hoặc địa chỉ IP của router tiếp theo trong trường họp nó không phải là router mặc định. |
| Right | Địa chỉ internet của thiết bị VPN bên phải |
| Rightsubnet | Mạng được bảo vệ bởi thiết bị VPN bên phải. |
| Rightid | Tên DNS của thiết bị VPN bên phải, đứng ngay sau kí tự Nếu DNS chưa được thiết lập cho địa chỉ IP, bạn có thể bỏ qua thông số này, nếu cho giỏ trị không đúng sẽ làm cho quá trình khởi tạo VPN bị thất bại. |
| Rightrsasigkey | Khóa RSA công khai của thiết bị VPN bên phải. Thông số này có thể xem bằng câu lệnh "ipsec showhostkey - -left". |
| Rightnexthop | Địa chỉ IP của router tiếp theo của thiết bị VPN bên phải khi nó muốn kết nối đến thiết bị VPN bên trái. Ta nên sử dụng giá trị được tự động tạo ra là “%def aultroute”, nó sẽ họp lệ trong hầu hết các trường họp, hoặc địa chỉ IP của router tiếp theo trong trường họp nó không phải là router mặc định. |

Bảng 3.2 : Các thông số trong file cấu hình letclipsec.conf

Khi đó có đầy đủ thông tin, bạn phải nhập vào tập tin cấu hình /etc/ipsec. conf. Các bước thực hiện như sau:

Bước 1: Chọn khóa RSA

Theo yêu cầu của tập tin cấu hình, ta cần lấy khóa công khai của cả hai thiết bị VPN bên hái và bên phải. Trong trường hợp chưa có khóa, ta cần tạo khóa và đưa nó vào tập tin /etc/ipsec.conf.

1. Tạo khóa RSA

Khi cài đặt OpenS/wan, chương trình sẽ tự động tạo khóa, nhưng nếu bạn muốn thay đổi khóa thì có thể thực hiện bằng các lệnh sau.

**[rootSVPNl tmp]#** ipsec rsasigkey—verbose 2048 > privatekey.tmp **[root@VPNl tmp]#**

Bạn có thể sửa tập tin /etc/ipsec.secrets bằng cách thay thế nội dung của đoạn "RSA: {" đến hết "} " bằng nội dung của tập tin keys.tmp được tạo ra ở trên.

Một cách đơn giản hơn là xóa tập tin /etc/ipsec.secrets, sau đó thực hiện câu lệnh sau:

**[root@VPN2 tmp]#** ipsec newhostkey—output /etc/ipsec.secrets **[root0VPN2 tmp]#**

1. Lấy khóa công khai bên trái

Trên thiết bị VPN bên trái ta thực hiện câu lệnh sau để đưa nội dung khóa công khai ra tập tin /tmp/left.pub.

**[rootỄVPNl tmp]#** ipsec showhostkey—left > /tmp/left.pub **[root@vpnl tmp]#**

1. Lấy khóa công khai bên phải

Trên thiết bị VPN bên phải ta thực hiện câu lệnh sau để đưa nội dung khóa công khai ra tập tin /tmp/right.pub.

**[root@VPNl tmp]#** ipsec showhostkey—right > /tmp/ríght.pub **[rootỄVPNl tmp]#**

Bước 2: Chỉnh sửa tập tin cấu hình /etc/ipsec.conf

Mỗi kết nối VPN trong tập tin cấu hình có thông tin riêng của nó. Ở đây ta tạo một kết nối có tên “lan-to-lan” và phải cung cấp tất cả câc thông số cần thiết.

#

* File: /etc/ipsec.conf

#

conn lan-to-lan

left=192.168.61.134 leftsubnet=192.168.132.0/24 leftnexthop=l92.168.61.135

leftrsasigkey=OsAQOBUgWIaiE8SVWmH6wUwCYsOcVMPUUrhMKkqaLUkLeVlu 9zJHA+wxU/H/dVOgDTWjMivSnCg/Prr964Tgp+LMlJlLNWdSvxlX2Q8ỵG9LfrA uq+bh2hVrqn+jG7PxlC7qmlwnOTGWjVt/ỵ4z6vrt6i5+ZEJx7rgsijXmjLQWlR KFpeIF6ONzvNx3qeVeHdkjDCbrG5brDgtG2eJx30hGlLYfeBQK5MACmGB+4+ev ZVd4NfUQgstK2tdVER70xei2fWCtsl+o402t6BlCGSJp3S09z58kdgMxEỵarsJ VuAHYEUlYXxfQNBRLVF/jJhpfIwscDRSM9CQveOYuSN9Qut2rMNG+P69trOB4F F0zhsqdX6naX

right=192.168.61.135 rightsubnet=192.168.17.0/24 rightnexthop=192.168.61.134

rightrsasigkey=OsAQOKt2f8THh6Bb4wDCH0ZF+A/BJqNDZQtwbfmxỵ59g3K0 iyGuJLo3krIRUsWLmnRHXcMeVLR3PxdSqdc0/4BKloZY+it5WFM9Kp9pZ9dUlM RvulsFxgulybb7bH+Wa5dP8mf5Kwj 9Z0v2AiIGUWnVngju9Nf+tFxbỵw3c0i9F HyBwtCsudJ:VJSN4GHiwlx20M8nbDlKNjYvMntw6BjulB4NPh01faMJ7UWf59c 8AZ6nHb6X8sMYsLYc8HqDvc5+gmVD3ChoXhgaJEwkUzza4f+5tN+S8ZmSgxD50 TaWj5EevhJmFD+DcRO7odm/QXSMtLwx0hoXJLiC5YOlUXBỵYilb/DXU+tvL0FP 2N5eItE+Md4mh

auto=add

Một số lưu ý quan trọng về tập tin /etc/ipsec.conf

1. Cần phải thực hiện lại đầu dòng đúng như quy định dưới đây Đúng

conn lan-to-lan left=x.X.X.X leftsubnet=y.y.y.y/2 4

Sai

conn lan-to-lan

left=x.X.X.X

leftsubnet=y.y.y.y/2 4

1. Thông tin về kết nối "lan-to-lan" phải giống nhau ừong tập tin /etc/ipsec.conf ở cả hai thiết bị VPN bên trái và bên phải.
2. Không được có dòng trống giữa các thông số của kết nối “lan-to- lan”. Dòng ghi chú chỉ được bắt đầu bằng ký tự # được chấp nhận ở đây.

Khởi động lại IPSec để chạy lại tập tin cấu hình không cần phải khởi động lại đường hầm. Neu bạn thiết lập “auto=add”, đường hầm sẽ chỉ được khởi động một cách thủ công bằng câu lệnh “ipsec auto — up lan-to-lan”.

Bước 3: Khởi động lại OpenS/wan

Việc này cần phải thực hiện ở cả hai đầu VPN để những thiết lập mới trong tập tin cấu hình /etc/ipsec.conf có hiệu lực.

[root@VPNl ~]# service ipsec restart ipsec\_setup: Stopping OpenS/wan IPsec...

ipsec\_setup: stop ordered, but IPsec appears to be already stopped!

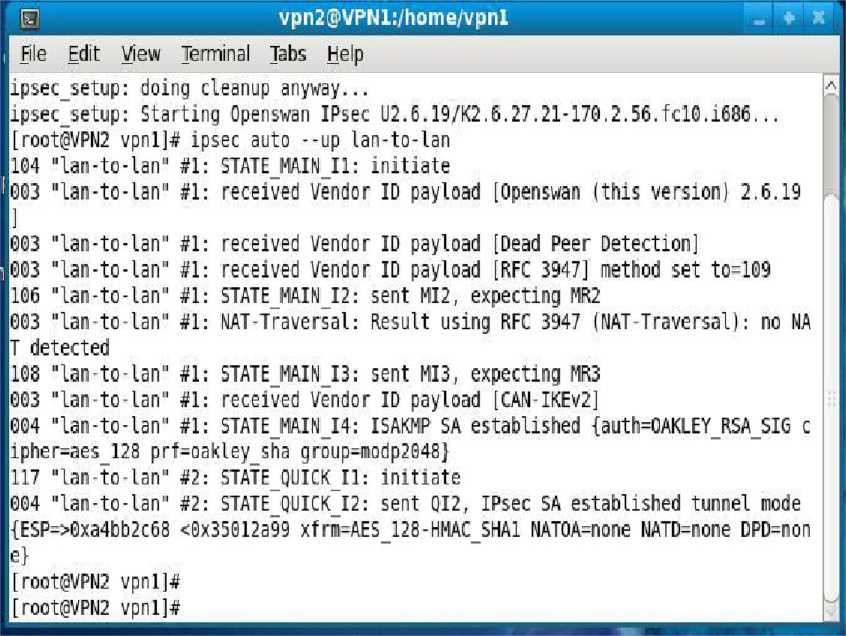
ipsec\_setup: doing cleanup anyway...

ipsec\_setup: Starting OpenS/wan IPsec U2.6.19/K2.6.27.21- 170.2.56.fcl0.i686...

[root@VPNl -]#

Bước 4: Khởi tạo một đường hàm mới

Bạn có thể sử dụng câu lệnh ipsec để khởi động đường hầm "lan-to-lan" đó được tạo trong tập tin /etc/ipsec.conf



Hình 3.5 : Khởi tạo một đường hầm mới

Bước 5. Thay đổi các rules của IP tables NAT/Masquerade

Nêu bạn đang sử dụng ỉptables với chế độ NAT/masquerading trên thiết bị VPN thì cần phải có một số điều chỉnh trong các rules của Iptaples.

• Thiết bị VPN bên ừái

Cũ

iptables -t nat -A POSTROUTING -o ethO -s 192.168.132.0/24 ~j MASQUERADE

Mới

ỉptables -t nat -A POSTROUTING -o ethO -s 192.168.132.0/24 -d \! 192.168.17.0/24 -j MASQUERADE

Thiết bị VPN bên phải Cũ

iptables -t nat -A POSTROUTING -o ethũ -s 192.168.17.0/24 -j MASQUERADE

Mái

iptables -t nat -A POSTROUTING -o ethO -s 192.168.17.0/24 -d \ ! 192.168.132.0/24 -j MASQUERADE

Bước 6 : Gỡ lỗi OpenS/wan

Việc gỡ lỗi luôn là một vấn đề quan trọng khi thiết lập VPN vì có nhiều thứ có thể gây ra lỗi. Sau đây là một số kiểm tra nhanh để bạn có thể chắc chắn hệ thống hoạt động chính xác và ổn định.

1. Kiểm tra trạng thái của đường hầm

Câu lệnh “ipsec auto — status” sẽ trả về trạng thái hoạt động của OpenS/wan trên thiết bị VPN. Ket quả gồm những phần sau:

IKE Section: Khai báo những thuật toán trao đổi khóa đã mã hóa và những tham số của chúng. Phần này cần ít nhất một sự trụng khớp giữa thiết bị VPN hai bên về thuật toán. Đồng thời phần này cũng cho biết một vài thông số của tiến trình trao đổi khóa mức thứ nhất, là bước đầu tiên xảy ra khi thiết lập một VPN.

ESP Section: Khai báo những thuật toán mã hóa khác nhau và những tham số của chúng. Phần này cũng cần ít nhất một sự trụng khớp giữa hai thiết bị VPN ở hai đầu. Đồng thời phần này chứa những thông tin về các tham số dựng trong quá trình trao đổi khóa ở mức thứ hai, là bước thứ hai và cũng là bước cuối cùng trong thiết lập VPN.

VPN Section: Phần này thường được bắt đầu bằng tên của đường hầm VPN, trong trường hợp này là "lan-to-lan" .

* Nếu không có dòng nào ở đây thì chưa có VPN nào được thiết lập.
* Nếu có nhiều thông tin nhưng thiếu dòng " STATE\_QUICK\_R2 (IPsec SA established)" thì khi đó IPSec đã được cấu hình nhưng đường hầm vẫn chưa được thiết lập. Điều này xảy ra là do đường hầm được kích hoạt khi SA được tạo ra nhưng sau đó lại bị đứt giữa chừng do một khoảng thời gian quá hạn. Cách tốt nhất để khắc phục vấn để này là cho dữ liệu được truyền tải qua đường hầm khi nó được thiết lập. Đơn giản nhất là ta sử dụng lệnh ping để gởi những gói ICMP nhằm kiểm tra trạng thái hoạt động của đường hầm.

|  |  |  |  |
| --- | --- | --- | --- |
| [root@VPN2 ~]# ipsec auto --status | | |  |
| 000 | using kernel interface: netkey | |  |
| 000 | interface lo/lo ::1 |  |  |
| 000 | interface lo/lo 127. | o  o  I-1 |  |
| 000 | interface lo/lo 127. | o  o  I-1 |  |
| 000 | interface ethO/ethO | 192.168.61. | 135 |
| 000 | interface ethO/ethO | 192.168.61. | 135 |
| 000 | interface ethl/ethl | 192.168.17. | 128 |
| 000 | interface ethl/ethl | 192.168.17 . | 128 |
| 000 | %myid = (none) |  |  |
| 000 | debug none |  |  |
| 000 |  |  |  |
| 000 | virtual private (%priv): | |  |
| 000  192. | - allowed 3 subnets: .168.132.0/24 | : 192.168.61 | .0/24, |
| 000 | - disallowed 0 subnets: | |  |

192.168.17.0/24,

000 WARNING: Either virtual\_private= was not specified, or there was a syntax

000 error in that line. 'left/rightsubnet=%priv'

will not work!

000

000 algorithm ESP encrypt: id=2, name=ESP\_DES, ivlen=8, keysizemin=64, keysizemax=64

000 algorithm ESP encrypt: id=3, name=ESP\_3DES, ivlen=8, keysizemin=192, keysizemax=192

000 algorithm ESP encrypt: id=7, name=ESP\_BLOWFISH, ivlen=8, keysizemin=40, keysizemax=448

000 algorithm ESP encrypt: id=ll, name=ESP\_NULL, ivlen=0, keysizemin=0, keysizemax=0 000 algorithm ESP encrypt: id=12, name=ESP\_AES, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=13, name=ESP\_AES\_CTR, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=14, name=ESP\_AES\_CCM\_A, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=15, name=ESP\_AES\_CCM\_B, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=16, name=ESP\_AES\_CCM\_C, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=18, name=ESP\_AES\_GCM\_A, ivlen=8, keysizemin=128, keysizemax=256

7

8

000 algorithm ESP encrypt: id=19, name=ESP\_AES\_GCM\_B, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=20, name=ESP\_AES\_GCM\_C, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=22, name=ESP\_CAMELLIA, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=252, name=ESP\_SERPENT, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP encrypt: id=253, name=ESP\_TWOFISH, ivlen=8, keysizemin=128, keysizemax=256

000 algorithm ESP auth attr: id=l,

name=AUTH\_ALGORITHM\_HMAC\_MD5, keysizemin=128, keysizemax=128 000 algorithm ESP auth attr: id=2,

name=AUTH\_ALGORITHM\_HMAC\_SHA1, keysizemin=l60, keysizemax=160 000 algorithm ESP auth attr: id=5,

name=AUTH\_ALGORITHM\_HMAC\_SHA2\_25 6, keysizemin=2 5 6, keys i z emax=2 5 6 000 algorithm ESP auth attr: id=8, name=AUTH\_ALGORITHM\_HMAC\_RIPEMD, keysizemin=160, keysizemax=160

000 algorithm ESP auth attr: id=9,

name=AUTH\_ALGORITHM\_AES\_CBC, keysizemin=128, keysizemax=128

000 algorithm ESP auth attr: id=251, name=(null), keysizemin=0, keysizemax=0

000

000 algorithm IKE encrypt: keydeflen=131

|  |  |  |
| --- | --- | --- |
| o  II  tỊ  ■H | name=(null), | blocksize=16, |
| id=3, | name=OAKLEY | BLOWFISH\_CBC, |
| id=5, | name=OAKLEY | 3DES\_CBC, |
| id=7, | name=OAKLEY | AES CBC, |

000 algorithm IKE encrypt: blocksize=8, keydeflen=128

000 algorithm IKE encrypt: blocksize=8, keydeflen=l92

000 algorithm IKE encrypt: blocksize=16, keydeflen=12

**IKE encrypt: id=65004, name=OAKLEY\_SERPENT\_CBC, keydeflen=128**

**IKE encrypt: id=65005, name =OAKLEY\_TWOFISH\_CBC, keydeflen=128**

**IKE encrypt: id=65289,**

**000 algorithm blocksize=16,**

**000 algorithm blocksize=16,**

**000 algorithm name=OAKLEY\_TWOFISH\_CBC\_SSH,**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 000 | algorithm | IKE | hash: | id=l |
| 000 | algorithm | IKE | hash: | id=2 |
| 000 | algorithm | IKE | hash: | id=4 |
| hashsize=32 | |  |  |  |
| 000 | algorithm | IKE | hash: | id=6 |

**hashsize=64**

**blocksize=16, keydeflen=128 name=OAKLEY\_MD5, hashsize=16 name=OAKLEY\_SHAl, hashsize=20 name=OAKLEY\_SHA2\_2 5 6,**

**name=OAKLEY\_SHA2\_512,**

000 algorithm IKE dh group: id=2, name=OAKLEY\_GROUP\_MODP1024, bits=1024

000 algorithm IKE dh group: id=5, name=OAKLEY\_GROUP\_MODP1536, bits=1536

000 algorithm IKE dh group: id=14, n ame =0AKLE Y\_GROUP\_MODP2048, bits=2048

000 algorithm IKE dh group: id=15, n ame =0AKLE Y\_GROU P\_MO DP3072, bits=3072

000 algorithm IKE dh group: id=16, n ame =0AKLE Y\_GROU P\_MO DP4096, bits=4096

000 algorithm IKE dh group: id=17, name =0AKLEY\_GROUP\_MODP6144, bits=6144

000 algorithm IKE dh group: id=18, name =0AKLEY\_GROUP\_MODP8192, bits=8192

000 **000 stats db\_ops: (curr\_cnt, total\_cnt, maxsz}**

:context={0,0,0} trans={0,0,0} attrs={0,0,0}

000

000 "lan-to-lan":

192.168.17.0/24===192.168.61.135<192.168.61.135>[+S=C]...192. 168.61.134<192.168.61.134>[+S=C]===192.168.132.0/24; erouted; eroute owner: #6

000 "lan-to-lan": myip=unset; hisip=unset;

000 "lan-to-lan": ike\_life: 3600s; ipsec\_life: 28800s;

rekey\_margin: 540s; rekey\_fuzz: 100%; keyingtries: 0

000 "lan-to-lan": policy:

RSASIG+ENCRYPT+TUNNEL+PFS+UP+IKEv2ALLOW; prio: 24,24; interface: ethO;

000 "lan-to-lan": newest ISAKMP SA: #8; newest IPsec SA:

#**6**;

000 "lan-to-lan": IKE algorithm newest: AES\_CBC\_128-SHA1-

MODP2048 000

000 #7: "lan-to-lan":500 STATE\_MAIN\_I4 (ISAKMP SA established); EVENT\_SA\_REPLACE in 2321s; lastdpd=-ls(seq in:0 out:0); idle; import:admin initiate

000 #2: "lan-to-lan":500 STATE\_QUICK\_I2 (sent QI2, IPsec SA established); EVENT\_SA\_REPLACE in 24058s; isakmp#l; idle; import:admin initiate

000 #2: "lan-to-lan" [esp.a4bb2c68@192.168.61.134](mailto:esp.a4bb2c68@192.168.61.134) [esp.35012a99@192.168.61.135](mailto:esp.35012a99@192.168.61.135) tun.00192.168.61.134 tun.00192.168.61.135 ref=0 refhim=4294901761

000 #8: "lan-to-lan":500 STATE\_MAIN\_R3 (sent MR3, ISAKMP SA established); EVENT\_SA\_REPLACE in 2813s; newest ISAKMP; lastdpd=-ls(seq in:0 out:0); idle; import:not set

8

0

000 #6: "lan-to-lan":500 STATE\_QUICK\_I2 (sent QI2, IPsec SA established); EVENT\_SA\_REPLACE in 24025s; newest IPSEC; eroute owner; isakmp#3; idle; import:admin initiate

000 #6: "lan-to-lan" [esp.aae2decf@192.168.61.134](mailto:esp.aae2decf@192.168.61.134) [esp.e8c20633@192.168.61.135](mailto:esp.e8c20633@192.168.61.135) tun.00192.168.61.134 tun.00192.168.61.135 ref=0 refhim=4294901761

000 #5: "lan-to-lan":500 STATE\_QUICK\_R2 (IPsec SA established); EVENT\_SA\_REPLACE in 24665s; isakmp#3; idle; import:not set

000 #5: "lan-to-lan" [esp.ac766d85@192.168.61.134](mailto:esp.ac766d85@192.168.61.134) [esp.7cl63dde@192.168.61.135](mailto:esp.7cl63dde@192.168.61.135) tun.00192.168.61.134 tun.00192.168.61.135 ref=0 refhim=4294901761

000 #4: "lan-to-lan":500 STATE\_QUICK\_R2 (IPsec SA established); EVENT\_SA\_REPLACE in 24665s; isakmp#3; idle; import:not set

000 #4: "lan-to-lan" [esp.80a7c272@192.168.61.134](mailto:esp.80a7c272@192.168.61.134) [esp.22aelllf@192.168.61.135](mailto:esp.22aelllf@192.168.61.135) tun.00192.168.61.134 tun.00192.168.61.135 ref=0 refhim=4294901761

000

[root@VPN2 -]#

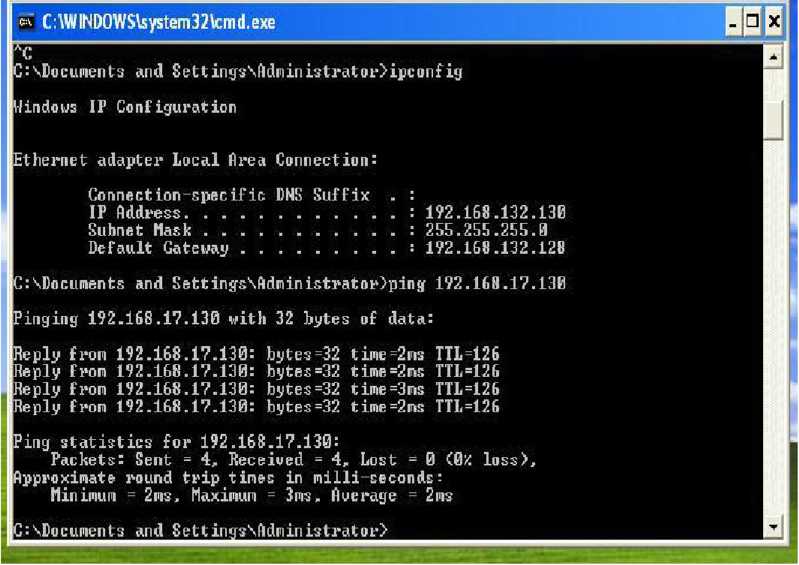
1. Kiểm tra kết nổi VPN

Ta có thể kiểm tra kết nối VPN bằng cách gửi lệnh “ping” từ mạng riêng này sang mạng bên kia. Trong mô hình ở trên, ta có thể gửi lệnh ping từ client 192.168.10.2 sang server 192.168.20.2 hoặc ngược lại

Nếu đường hầm đó được kích hoạt nhưng lệnh ping không hoạt động thì bạn cần kiểm tra các phần sau:

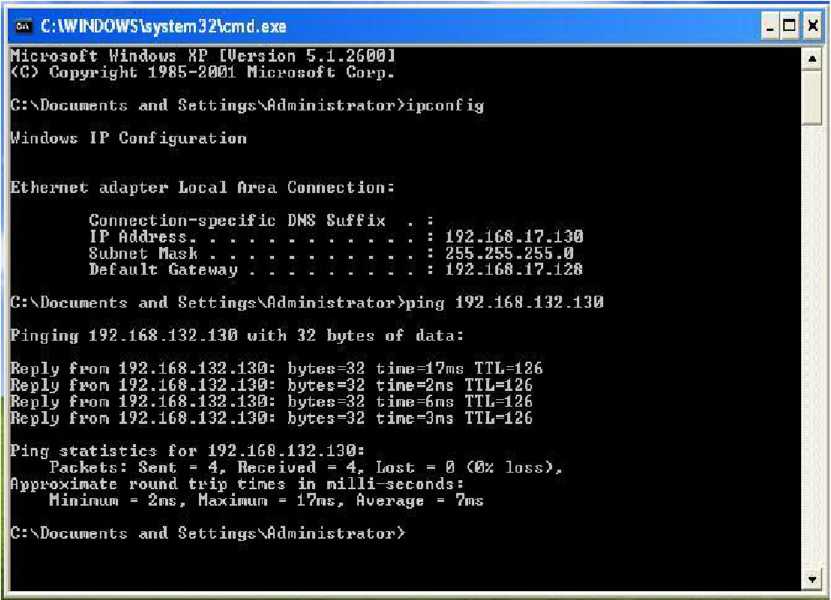
* Kiểm tra lại bảng route của các máy ở cả hai đầu, nhất là tại hai thiết bị VPN.
* Kiểm tra lại tường lửa và các quy luật đã thiết lập.

- Trên client 1 : (có IP là 192.168.132.130)



Hình 3.6 : Kiểm tra kết nối trên Client 1.

- Trên clinet 2 : (Có IP là 192.168.17.130)



*Hình 3.7 : Kiểm* tra kết nối trên *mấy* Client *2.*

1. Dùng TCP dump

Đe kiểm tra xem các đường hầm đã thông hay chưa hay chỉ đơn giản muốn xem gói tin đó được IPSec đóng gói lại như thế nào, chúng ta có thể dùng TCPdump, hoăc nếu muốn biết cụ thể hơn thì nên cài một IDS (Snort hoàn toàn miễn phí và nhiều tính năng).

Đây là kết quả của lệnh ping khi các gói ICMP chưa được mã hóa:

[root@VPN2 ~]# tcpdump -n -i ethl icmp

tcpdump: verbose output suppressed, use -V or -w for full protocol decode

listening on ethl, link-type EN10MB (Ethernet), capture size 96 bytes

00:18:39.346897 IP 192.168.17.130 > 192.168.132.130: ICMP echo request, id 512, seq 19712, length 40 00:18:39.347535 IP 192.168.132.130 > 192.168.17.130: ICMP echo reply, id 512, seq 19712, length 40 00:18:39.391381 IP 192.168.132.130 > 192.168.17.130: ICMP echo request, id 512, seq 20992, length 40 00:18:39.391847 IP 192.168.17.130 > 192.168.132.130: ICMP echo reply, id 512, seq 20992, length 40

00:18:40.346644 IP 192.168.17.130 > 192.168.132.130: ICMP echo request, id 512, seq 19968, length 40 00:18:40.349427 IP 192.168.132.130 > 192.168.17.130: ICMP echo reply, id 512, seq 19968, length 40

00:18:40.392314 IP 192.168.132.130 > 192.168.17.130: ICMP echo request, id 512, seq 21248, length 40 00:18:40.392992 IP 192.168.17.130 > 192.168.132.130: ICMP echo reply, id 512, seq 21248, length 40 Ac

8 packets captured 8 packets received by filter 0 packets dropped by kernel [root@VPN2 -]#

Và đây là kết quả khi các gói ICMP được đóng gói và gửi trong đường hầm giữa hai thiết bị VPN. Địa chỉ IP nguồn và đích thật sự đã được ẩn đi, thay vào đó là hai địa chỉ IP Internet ở đầu ngoài của thiết bị VPN.

[root0vpn2 tmp]# **tcpdump -n -i ethl** 02:08:23.637149 IP 192.168.61.134> 192.168.61.135: ESP(spi=0xf4909a7e,seq=0x73)

02:08:24.635302 IP 192.168.61.135> 192.168.61.134: ESP(spi=0x808e9a87,seq=0x74)

02:08:24.637988 IP 192.168.61.134> 192.168.61.135: ESP(spi=0xf4909a7e,seq=0x74)

02:08:25.638015 IP 192.168.61.135> 192.168.61.134:

ESP(spi=0x808e9a87,seq=0x75)

Ac [root@vpn2 tmp]#

1. Kiểm tra thông báo lỗi

Thường xuyên kiểm tra tập tin /var/log/messages để nhận biết kịp thời các thông báo lỗi và có hướng khắc phục phù họp.

Thông điệp "Invalid Key"

Nếu khóa công khai bên trái và bên phải được khai báo trong tập tin không phù hợp với khóa bí mật được được lưu trữ trong tập tin /etc/ipsec.secrets thì sẽ xuất hiện thông điệp báo khóa không khớp và mọi thông tin khác sẽ bị bỏ qua.

003 "net-to-net" #1: ignoring informational payload, type ***INVALID KEY INFORMATION***

003 "net-to-net" #1: received and ignored informational message

003 "lan-to-lan" #1: discarding duplicate packet; already

STATE\_MAIN\_I3

031 "lan-to-lan" #1: max number of retransmissions (2) reached STATE\_MAIN\_I3. Possible

authentication failure: no acceptable response to our first encrypted message

1. - ứng dụng trong bảo mật mạng Road Warrior
2. - Road Warrior Cần phải biết:

* IP tĩnh của gateway.
* Dải IP của các subnet phía sau gateway.
* Tên mà mỗi bên có thể xác định chính nó cho IPsec negotiations. Nó là một tên miền có đủ tiêu chuẩn được đi trước bởi @. Ví dụ :

@ road.example.com.

1. -Lấy leftrsasigkey.

Trên máy tính của chúng ta, lấy ra mã công khai của IPSec bằng câu

lệnh :

ipsec showhostkey —left

Sẽ cho ta ra kết quả sau :

# RSA 2192 bits road.example.com Sun Jun 9 02:45:02

2002

leftrsasigkey=0sAQPIPN9uI...

1. - Lấy rightrsasigkey. Console trên gateway :

ssh2 xy.example.com

Xem mã khóa công khai của gateway bằng câu lệnh :

ipsec showhostkey —right

NÓ sẽ cho ta kết quả về mã khóa công khai của gateway :

# RSA 2048 bits xy.example.com Fri Apr 26 15:01:41 2002 rightrsasigkey=OsAQOnwiBPt...

1. - Tùy chỉnh file letclipsec.conf.

Trên máy tính của chúng ta, sao chép phần dưới đây vào file /etc/ipsec.conf.

conn road

left=%defaultroute #

Picks up our dynamic IP Local information Remote information

authorizes but doesn't start connection at startup

leftnexthop=%defaultroute #

leftid=@road.example.com #

leftrsasigkey=0sAQPIPN9uI... #

right=192.0.2.10 #

rightsubnet=10.0.0.0/24 #

rightid=@xy.example.com #

rightrsasigkey=OsAQOnwiBPt... #

auto=add #

this

#

Khung mẫu cho các gateway khác, chú ý làm sao cho đảo ngược trái và phải, giữ trong cùng một quy ước là bên trái là local, bên phải là remote. Chắc chắn sự chuyển đổi trong rsasigkeys giữ đúng điều này.

ssh2 xy.example.com vi /etc/ipsec.conf

và thêm vào phần sau :

conn road

left=l92.0.2.2 leftid=0xy.example.com leftsubnet=l92.0.2.128/29 leftrsasigkey=OsAQOnwiBPt... rightnexthop=%defaultroute right=%any laptop's IP

rightid=@road.example.com rightrsasigkey=0sAQPIPN9uI... auto=add

this

* Gateway's information

#

#

#

* correct in many situations
* Wildcard: we don't know the

#

#

* authorizes but doesn't start
* connection at startup

1. — Bắt đầu kết nối.

Chúng ta cần phải bắt đầu kết nối từ cạnh Road Warrior. Trên máy trạm, gõ câu lệnh :

ipsec auto --start net-to-net

Chúng ta có thể nhìn thấy :

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 104 | "net-net" #223: STATE MAIN | | | | 11: initiate | |  |
| 106 | "road" | #301 | STATE | \_MAIN\_I2 : | sent MI2, | expecting | MR2 |
| 108 | "road" | #301 | state" | MAIN 13: | sent MI3, | expecting | MR 3 |
| 004 | "road" | #301 | state" | MAIN 14: | ISAKMP SA | established | |
| 112 | "road" | #302 | state" | \_QUICK\_I1 | : initiate |  |  |
| 004 | "road" | #302 | state’ | \_QUICK\_I2 | : sent QI2, | IPsec SA |  |

established

1. - MASQ hoặc NAT những gói packet trong đường hầm.

Nếu chúng ta sử dụng IP masquerade hoặc Network Address Translation (NAT) trên gateway vào, điều ta cần bây giờ là miễn những gói packet mà bạn muốn từ đường hầm từ nghiên cứu. Ví dụ, nếu chúng ta có luật sau :

iptables -t nat -A POSTROUTING -o ethO -s 10.0.0.0/24 -j MASQUERADE

Chúng ta có thể thay đổi một thành :

iptables -t nat -A POSTROUTING -o ethO -s 10.0.0.0/24 -d \! 192.0.2.128/29 -j MASQUERADE

1. - Kiểm tra kết nối.

Từ máy trạm, ping tới máy đằng sau của gateway:

ping ns.winston.example.com

Dùng lệnh tcpdump để bắt các gói tin :

tcpdump -i wlanO

Cài đặt thành công nếu nhận được thông tin như ở dưới:

19:16:32.046220 192.0.2.2 > 192.0.2.9:

ESP(spi=0x3be6c4dc,seq=0x3)

19:16:32.085630 192.0.2.9 > 192.0.2.2:

ESP(spi=0x5fddlcf8,seq=0x6)

KẾT LUẬN

Trong quá tành phát triển của nền công nghệ thông tin thì việc nghiên cứu và phát triển các gói phần mềm giúp tạo ra những kênh liên lạc riêng trên nền hệ điều hành Linux cần có một thòi gian dài để phát triển mà đảm bảo được những tiêu chí của người dùng đưa ra là phải đảm bảo dữ liệu trao đổi trên mạng không bị rò rỉ, mất mát.

Trong thời gian ba tháng với những phạm vi và yêu cầu của đổ án, dưói sự hướng dãn tận tình của TS. Lê Đình Thích và Ths. Nguyễn Thanh Sơn, và sự giúp đỡ của nhà trường đã giúp em hoàn thành tốt đồ án này.

Em xin chân thành cảm ơn tới thầy giáo TS. Lê Đình Thích và Ths. Nguyễn Thanh Sơn, các thầy cô trong trường và các bạn đã giúp em hoàn thành đồ án này.

Kết quả đã đạt được:

Trong thời gian trên, với yêu cầu đề ra của đồ án em đã hoàn thành được những phần sau :

Về phần lý thuyết em đã hoàn thành được các nội dung sau đây:

* Hiểu được về giao thức TCP/IP.
* Nắm vững được công nghệ IPSec.
* Hiểu được tầm quan trọng của IPSec trong thực tế.
* Tìm hiểu được về gói phần mềm OpenS/wan.
* Hiểu được chức năng, ứng dụng của gói phần mềm OpenS/wan trong việc tạo các kênh truyền riêng trên Internet.

Về phần thực hành :

* Thực hiện cài đặt thành công được gói phần mềm OpenS/wan trên hệ điều hành Linux (Trong đồ án này em đã cài trên hệ điều hành Fedora 10)
* Tạo được kênh liên lạc riêng giữa hai máy trạm thuộc hai subnet khác nhau thông qua OpenS/wan.

TÀI LIỆU THAM KHẢO

Tài liệu tiếng việt:

1. Trần Văn Thanh, Mô hình TCP IIP.
2. Nguyễn Thúc Hải, Mạng và hệ thống mở.

Tài liệu tiếng anh :

1. Naganand Doraswamy, Dan Harkins IPSec: The New Security Standard for the Internet, Intranets, and Virtual Private Networks, Second Edition (Pub Date : March 13, 2003), Published by : Prentice Hall PTR, ISBN: 013046189X.
2. Paul Wouters and Ken Bantoft Building and Integrating Virtual Private Networks with Opens/wan (Fkst published : February 2006), Published by Packt Publishing Ltd, ISBN 1904811256.
3. James s. Tiller , A Technical Guide to IPSec Virtual Private Networks (Pub Date : December 11, 2000), Published by Auerbach Publications, ISBN: 9780849308765.

Một số website :

1. <http://www.Qpenswan.org> Tìm hiểu vé gói phẩn mém OpenS/wan.
2. <http://www.freeswan.org/> Tìm hiểu về gói phần mềm FreeS/wan
3. <http://www.natecarlson.com/linux/ipsec-x509.php>
4. [http://techgurulive.com/2008/ll/12/how-to-configure-Qpenswan- Openswan-to-Qpenswan-connections-using-rsasig-or-psk/](http://techgurulive.com/2008/ll/12/how-to-configure-Qpenswan-Openswan-to-Qpenswan-connections-using-rsasig-or-psk/)

1. 0 [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)
3. [↑](#footnote-ref-3)
4. Chỉ cài userland: [↑](#footnote-ref-4)