DESIGN AND IMPLEMENTATION OF ROUTER REMOTE CONTROL SYSTEM SUPPORTING STATIC ROUTE SUMMARIZATION

Ali Hussein Ali

Alialnooh71@gmail.com

Mosul University/Computer Science Department/2008

Abstract

In order to improve and upgrade the performance of devices used in data communications, such devices are particularly Routers, which are used to link computer networks among themselves, and for identification and selection of the shortest ways to access data, this research is designed and programmed for remote controlling and managing a router that links two networks by managing the Route table through controlling Route table contents. This is done by applying Route summarization algorithm for all routes that are defined statically, which aims to reduce the number of table's routes hence, increasing search operation and arise all burdens from the network administrator from route summarization. Also this research gives the ability to the network administrator to administrate and manipulate the router device without using the Telnet service which is always under attacks by the hackers. In this research, Windows Server 2003 Enterprise Edition has been used to convert a computer to a router. The research has been tested on network computers that composed of three computers, one of them is a router used to link two networks, the second computer is supplied by the Router Administration System. The other computer acts as another network. Programming is accomplished using C# programming language, one of the Visual Studio .NET Environment languages.

الخلاصة

من أجل تحسين ورفع اداء الاجهزة التي تُستخدم في تناقل البيانات ومنها بالخصوص اجهزة الموجهات (Routers) التي تستخدم في ربط شبكات الحاسبات فيما بينها من خلال تحديد واختيار اقرب طرق وصول البيانات، تم في هذا البحث تصميم برنامج لغرض التحكم عن بعد وإدارة أحد الموجهات التي تربط بين شبكتين وذلك بالسيطرة على جدول المسارات (Route table) الخاص بالموجه من خلال التحكم بمحتويات جدول المسارات وبتطبيق خوارزمية الـ (Route summarization) لجميع المسارات المعرفة بطريقة الـ (Static Routes) حيث تهدف هذه الخوارزمية إلى إختزال اكبر عدد من المسارات الموجودة بالجدول لغرض تسريع عملية البحث فيه وبالتالي رفع العبأ عن مدير الشبكة في اختزال المسارات. وكذلك وفر البرنامج سهولة لمدير الشبكة في الإدارة والتعامل مع الموجه وبدون استخدام المنافذ المشهورة (مثل منفذ الـ Telne) التي تكون معرضة وبشكل دائم للهجوم من قبل القراصنة. تم استخدام نظام Windows Server 2003 Enterprise Edition من أجل تحويل الحاسبة الى موجه شبكة. كما تم اختبار البحث على شبكة حاسبات مؤلفة من ثلاث حاسبات ، احدى الحاسبات تعمل كموجه يربط بين شبكتين، وحاسبة اخرى تعمل لاستقبال نظام إدارة الموجه، والحاسبة الأخيرة تعمل كشبكة اخرى قائمة بحد ذاتها. استخدمت لغة C# البرمجية، إحدى اللغات التابعة لـ Visual Studio .NET Environment.

1.1 Introduction

As we know, networking devices especially routers or gateways are very important devices that needs a professional staff to work on it.

This is because it contains so many information that guide the traffic of the packets between Local Area Networks (LAN), or Wide Area Networks (WAN) and Internet.

All information are programmed by network administrators and saved inside the router. One of the most important information is the route table.

To guarantee the access of packets between all hosts inside the network and other hosts of other networks, route table most be defined in an accurate way. This can be done using **Static routing** and **Dynamic Routing**. [1]

For Static routing the network administrator is responsible for defining the routes manually of other networks in the route table using special commands.

While in Dynamic routing the network administrator is not responsible for manual definition of networks routes, but is responsible for the activation of some dynamic routing protocols like Routing Information Protocol (RIP), Open Shortest Path First (OSPF), … etc that supports the building of the route table using dynamic routing.

Routing information is so critical to networks behaviors because any error occurred inside will decrease the performance and network utilization, and will not pass the traffic in a good manner. [2]

1.2 Relative Works

The most relative work that supports router administration is "WinAgents RouterTweak", which is a specialized terminal shell making it easier to control the configuration of routers, switches and firewalls from the Cisco Systems company. This system does not support static route summarization and works on Telnet port (port 21).

1.3 Aim of Research

This research aims to improve computers networks performance through increasing the speed of packets direction (routing) inside the router.

This is done by using route summarization algorithm. Hence, the number of routes of router’s route table would decrease and consequently, the comparisons performed for the route table when receiving a packet from any computer connected to a network, would decrease too.

Since route summarization is supported automatically in dynamic routing like RIP and OSPF and not supported for static routing. So if the administrator deals with static routing , let us to say hundreds of routes inside the route table, the need to accomplish that is to design a system for this purpose that accomplish route summariztion on static routes without the need to the administrator, that was the first objective of the system.

The other objective gained from this work is to remote control the router from another computer, and changing the route table through applying different route instructions, such as addition, deletion, and changing, host filtering besides to route summarization process. As a result the research has simplified the administrator tasks and offered him/her the ability to remote control and program the router without the need to use Telnet service wich is always attacked by the hackers.

2.1 Sockets

A socket is an object consists of two parts, IP address and port number. Socket represents a low-level access point to the IP stack. This socket can be open or closed or one of a set number of intermediate states. A socket can send up and receive data down this connection. Data is generally sent in blocks of a few kilobytes at a time for efficiency – depending on the network technology – each of these blocks is called a packet.

All packets that travel on the Internet must use the Internet protocol. This means that the source IP address, destination address must be included in the packet. Most packets also contain a port number. A port is simply a number between 1 and 65,535 that is used to differentiate higher protocols, such as email or FTP. Ports are important when it comes to programming network applications because no two processes can use the same port. [3]

Sockets programming are two or more applications communicate with each other. The applications are typically on different computers but they can be on the same computer. For the two applications to talk to each other, one application is generally a server that keeps listening to the incoming requests (listening on a specified port number) and the other application acts as a client or makes the connection to the server application.

The server application can either accept or reject the connection. If the server accepts the connection, a dialog can begin between the client and the server. Once the client is done with whatever it needs to do, it can close the connection with the server. During the time that the client has an active connection it can send the data to the server and/or receive the data. [4].

2.2 Routing

Routing is the process of selecting paths in computer networks to send data traffic. Routing is performed in many kinds of networks, including the telephone network, the Internet, … etc.

Routing directs forwarding, the passing of logically addressed packets from their source toward their ultimate destination through intermediary nodes. The routing process usually directs forwarding on the basis of routing tables, which maintain a record of the routes to various network destinations. Thus constructing routing tables, which are held in the routers' memory, becomes very important for efficient routing. [5]

2.2.1 Static Routing and Dynamic Routing

IP routing can be partitioned into two broad categories: static routing and dynamic routing. Routers are called static if they do not change. Thus, a static routing table is loaded with values when the system starts, and the routes do not change unless an error is detected. In contrast, dynamic routing refers to a system that can change routing table information over time. Dynamic routing begins exactly like static routing by loading an initial set of routes into a routing table when the system boots. **[**5].

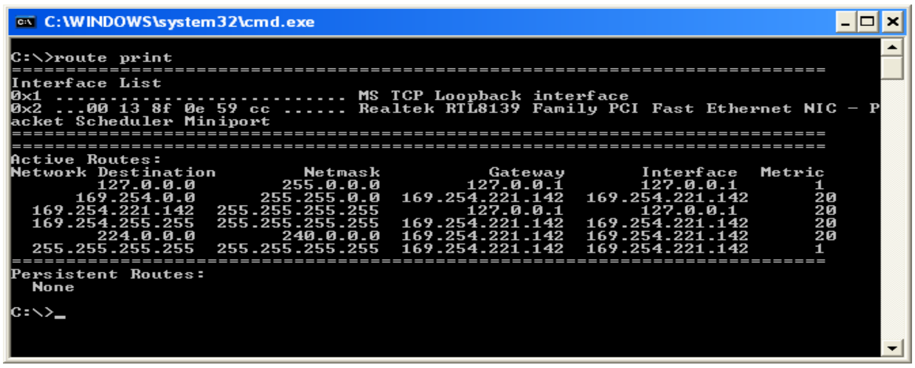
Static routing contains information entered manually. The administrator enters the route for each destination into the table. When a table is created, it cannot update automatically when there is a change in the internet. The table must be manually altered by the administrator. Static routing can be used in LAN or WAN that does not change very often [6]

2.2.2 Routing Table

Routing table is organized in a hierarchical scheme with direct-delivery entries first, host-specific delivery entries next, network-specific entries third, and the default delivery entry last.

The routing table usually has these fields: mask, destination address, next-hop address (gateway), interface and metric.

- ❖ **Mask:** This field defines the mask applied to the destination IP address of the packet to find the network or sub-network address of the destination.
- ❖ **Destination address:** This field defines either the destination host address (host specific address) or the destination network address (network-specific) address.
- ❖ **Next-hop address:** This field defines the address of the next-hop router to which the packet is delivered.
- ❖ **Interface:** This field shows the name of the interface that is responsible for routing the packets.
- ❖ **Metric:** This field contains a value that is used in dynamic routing when two or more paths are existed in the table, usually the lowest one is chosen. Figure (1) shows a view of route table:



```
C:\>route print
===========================================================================
Interface List
0x1 ........................... MS TCP Loopback interface
0x2 ...00 13 8f 0e 59 cc ...... Realtek RTL8139 Family PCI Fast Ethernet NIC - P
acket Scheduler Miniport
===========================================================================
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
        127.0.0.0        255.0.0.0        127.0.0.1       127.0.0.1       1
      169.254.0.0      255.255.0.0  169.254.221.142 169.254.221.142      20
  169.254.221.142  255.255.255.255        127.0.0.1       127.0.0.1      20
  169.254.255.255  255.255.255.255  169.254.221.142 169.254.221.142      20
        224.0.0.0        240.0.0.0  169.254.221.142 169.254.221.142      20
  255.255.255.255  255.255.255.255  169.254.221.142 169.254.221.142       1
===========================================================================
Persistent Routes:
  None

C:\>_
```

Figure(1)Aview of route table.

2.2.3 Routing Table Search Algorithms

The algorithms that search the routing table must also be changed to make classless routing more efficient.

- Searching in Classful Addressing:

In classfull addressing, the routing table is organized as a list. However, to make searching easier, the routing table can be divided into three buckets (areas), one for each class. When the packet arrives, the router applies the default mask (which is inherent in the address itself) to find the corresponding bucket (A, B, or C). The buckets then search the corresponding buckets instead of the whole table.

- Searching in Classless Addressing:

In classless addressing, we can also use buckets; specifically, 32 buckets, one for each prefix length. However, the problem is that there is no self-contained information in the destination address to help the router decide which bucket to search. The simplest, but not the most efficient, method is called **longest match**. The router first tries to use the longest prefix (/32). If the destination address is found in the bucket, the search is complete (this bucket is for host-specific routing). If the address is not found, the bucket for the next prefix (/31) is searched. And so on. It is obvious that this type of search takes along time; on average, 16 bucket must be searched. [6].

2.2.4 Route Summarization

In large internetworks, hundreds, or even thousands, of network addresses could exist. It is often problematic for routers to maintain this volume of routes in their routing tables. Route summarization can reduce the number of routes that a router must maintain, because it is a method of representing a series of network numbers in a single summary address.

The number of subnets and network addresses contained in routing table is rapidly increasing based on expansions in the networks. This growth has had a negative impact on CPU resources, bandwidth, and memory used to maintain the routing tables.

Therefore, route summarization was introduced as a way to reduce the size of network routing tables.

If configured properly, route summarization can reduce the latency associated with router hop, since the average speed for routing table lookup will be increased due to the reduced number of entries.

The overhead for routing protocols can also be reduced since fewer routing entries are being advertised.

Another advantage of using route summarization in large, complex networks is that it can isolate topology changes from other routers. This can aid in improving the stability of the network by limiting the propagation of routing traffic after a network link goes down.

For example, if a router only advertises a summary route to the next router hop, then it will not advertise any changes to specific subnets within the summarized range. This can significantly reduce any unnecessary routing updates following a topology change. Hence, increasing the speed of convergence and allowing for a more stable environment. The following explanation discusses how route summarization is implemented:

In table (1), Router A has the following networks in its routing table:

Address	First Byte	Second Byte	Third Byte	Fourth Byte
192.168.98.0	11000000	10101000	01100010	00000000
192.168.99.0	11000000	10101000	01100011	00000000
192.168.100.0	11000000	10101000	01100100	00000000
192.168.101.0	11000000	10101000	01100101	00000000
192.168.102.0	11000000	10101000	01100110	00000000
192.168.105.0	11000000	10101000	01101001	00000000

Table (1): An example for route summarization

First of all, the address must be converted to binary format and aligned them in a list as shown in the table.

Summarized route will be:

Net ID is :192.168.96.0

192.168.96.0	11000000	10101000	01100000	00000000

Mask is: 255.255.240.0

255.255.240.0	11111111	11111111	11110000	00000000

Therefore, the previous route table with six entries will be summarized to one route entry only. [5]

For dynamic routing, the system check the metric field for the similar routes, and if the metric is equal to each other, summarization will be applied, otherwise no summarization is done.

2.2.5 Route Commands

In Microsoft software routers, there are commands deal directly with the route table.

These commands are:

- **route add :** This instruction will add a route to the route table.
- **route change :** This instruction will change a route to a new gateway.
- **route delete :** This instruction will delete a route from the route table.
- **route print :** This instruction will print the route table.
- **route –f :** This instruction will delete all the route table.

The table below shows these commands.

#	Command	Syntax
1	route add	route add *destination* mask w.x.y.z *gateway*
2	route change	route change *destination* mask w.x.y.z *gateway*
3	Route delete	route delete *destination*
4	Route print	route print
5	route -f	Route -f

Table (2): Route Commands

3.1 System Design

The design phase of the proposed system consists of two parts, hardware building and software design.

All physical requirements needed to run the proposed system will be explained in paragraph 3.2. This paragraph will represent the infrastructure of the system. Software design and all algorithms are explained in paragraph 3.3 and so on.

3.2 Hardware Building

The proposed system was implemented on a hypothetical network that consists of three computers as shown in figure (2). This infrastructure needs the following requirements:

- ❖ Three PCs type P4. Client to run the router administration program. The other will work as a router between the two clients, with four network interface cards (NIC), one for each client and two for the router.

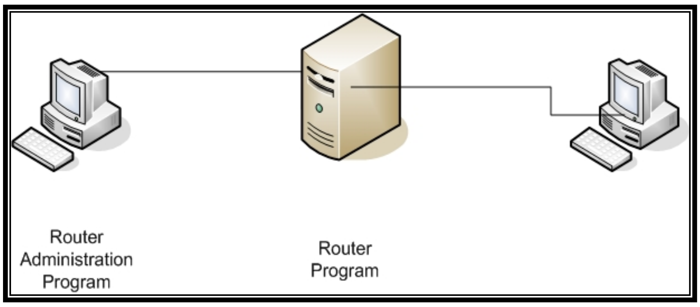


Figure (2): The hypothetical network used by the system.

3.3 Software Design

The system needs the following software for the three computers.

- ❖ Windows Server 2003 Enterprise Edition for the router computer.
- ❖ Windows XP for the clients computers.
- ❖ .NET Framework environment for the router and administrator computer.

The block diagram of the proposed system, will be shown in figure (3).

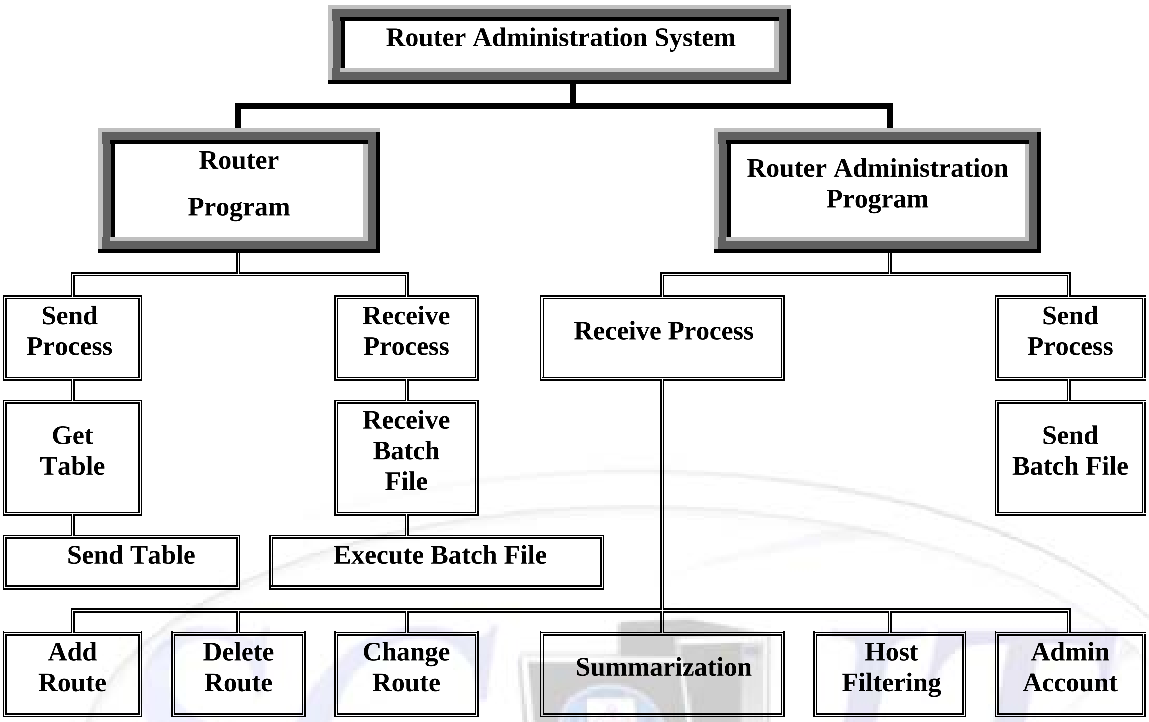


Figure (3): System block diagram.

3.3.1 Creating C# System Programs

C# language was used to program the system. Two C# programs have been created, one for administration computer and the other for the router.

The first program is called **Router Program** and the other is called **Router Administration Program**. Figure (4) shows the flowchart of the first program, while figure (5) shows the flowchart of the second program.

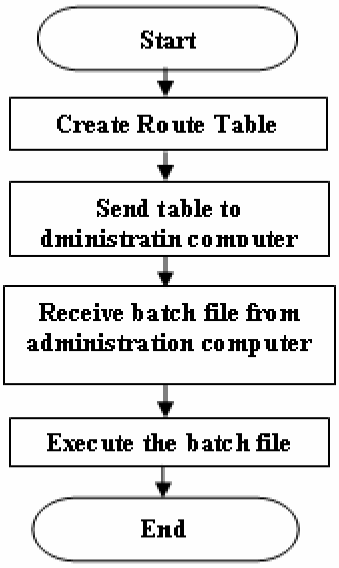

Start
Create Route Table
Send table to dministratin comouter
Receive batch file from administration computer
Execute the batch file
End

Figure (4): Router program flowchart

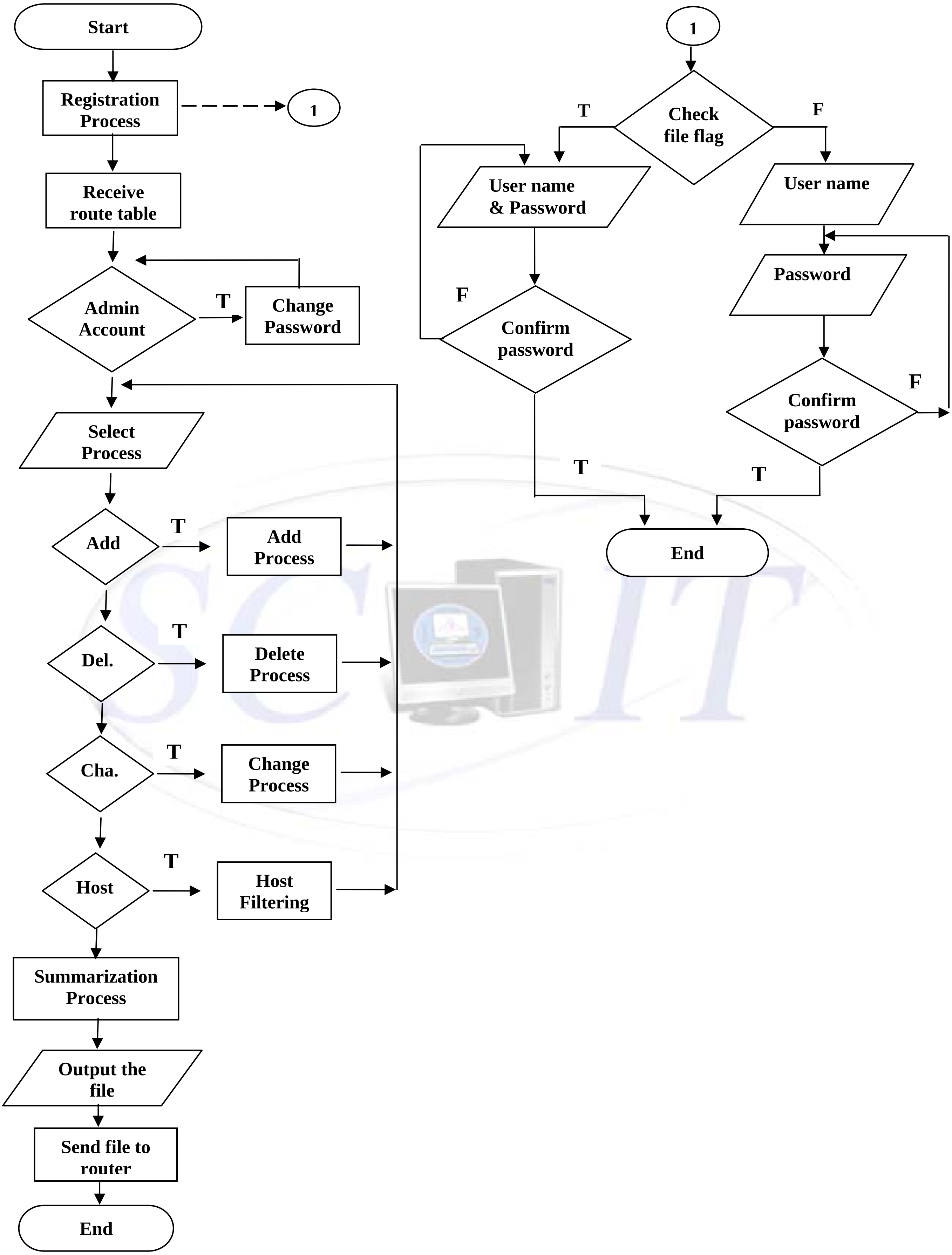


Figure (5): Router administration program flowchart

3.3.2 Application Interfaces

The first program (Router Program) could be implemented in two modes, the first one is manual-mode. Within this mode a form is designed, which contains several controls, such controls are Text Box, Buttons, and a Label. The Text Box is used to enter the administration computer IP (which runs the router administration system). Also, three Buttons are used in this design, the first one for getting the route table, the second is for sending it, while the third is to execute a batch file in order to create the new route table.

The second mode is an auto-mode. Within such mode a Picture Box is shown to notify the administrator that the program is running (this mode depends on a predefined IP for the computer that runs the router administration system).

For the second program, i.e. router administration system, many forms have been designed, each for a specific task, to implement the software procedures.

The first form represents the Administrator Account form, which utilizes the registration and software protection.

In the other forms, each tab Page control handles one of the router instructions explained previously.

Each Tab Page is supplied with a Check Box for route print instruction that gives the route table records throughout the software execution.

3.3.3 Transfer Files

The TCP server is multithreaded. In this case, three threads are used: the main thread maintains the user interface, a second thread listens for connections (listenerThread), and a third thread handles the connections (handlerThread).

One socket is required for each connection and will remain loaded in memory until the connection is closed. These sockets need to be stored in a dynamic array rather than a standard array because it is impossible to predict how many connections will be received.

A determined method called Form1_Load is used to start the thread that will wait for incoming connections, i.e., listenerThread. If this thread were to be called directly, the program would become unresponsive and appear to hang, while the socket waited on incoming connections. This effect is avoided by executing the listenerThread method in a separate thread of execution, which can block without adversely affecting the user interface.

The listenerThread method's function is to wait inefinitely for TCP connections on port 7002 (a random port number above 1024) and then to redelegate the work of handling these requests to the handlerThread method. This function also reports the source of the connections.

This time, the reason for redelegating work to a thread is not to maintain the responsiveness of the user interface, but rather to ensure that the application will continue to listen for new connections while it is handling a previous client. The new thread will be required to have access to the socket that is dealing with the current client. Otherwise, there would be no means of returning data.

The listenerThread will block on the call to AcceptSocket. Execution will not continue until an incoming connection has been detected; when it has, a new socket is created and the handler thread is invoked.

The remainder of the work is carried out in the handlerThread method. This function finds the last used socket and then retrieves the stream from this socket. Once the stream is fully received, its contents are copied into this array, and once the connection closes, the data is written to a specified file type. It is important to have the lock( ) keyword around the lines of code associated with file access; otherwise, if two concurrent connections try to access the same file, the program will crash. The socket is then set to null to remove it from memory. [4].

3.3.4 Router Administration Program Design

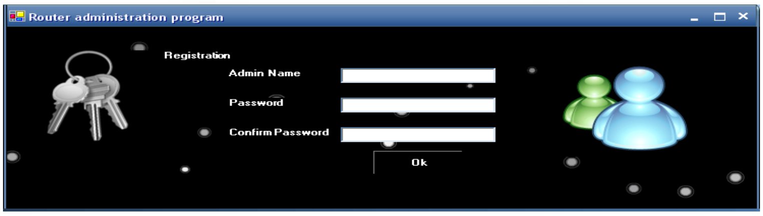
This research is formed using multi-forms and various controls to satisfy its job. Also, different methods are written for file processing and router instruction handling. The following points give the explanation for program design.

3.3.4.1 Administrator Account Form

The application has to be protected from any un-authorized access, therefore; a form is designed, which depicts the following steps, as shown in figure (6) and (7):

Step1: Registration: running the application for the first time requires a form of administrator registration. This implies entering the administrator name and password and confirms password. At the end of this process the administrator count is saved as a database for the application.

Step2: Logon: Subsequent software running -after registration- involves the application to ask again for the name and password. Comparing these two values with the earlier stored values, and starting further application's functions, or else, a message is displayed to the tell administrator that an error has occurred.



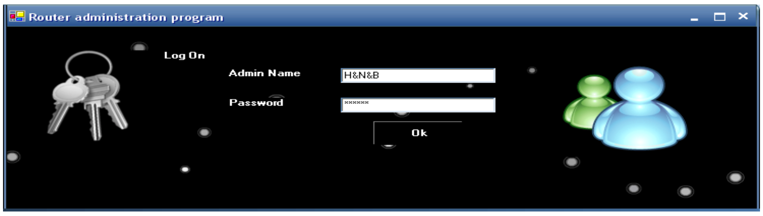
Registration

Admin Name

Password

Confirm Password

Figure (6): Router Administration Program/Registration.



Log On

Admin Name H&N&B

Password *******

Figure (7): Router Administration Program/Logon.

3.3.4.2 Receive Form

This form is designed for programming the receiving process for the routing table.

Step1: The read function of NetworkStream object copies the information of the received route table to an array of bytes, as the latter has to be written into a new text file using Stream class.

Step2: Most of the information found in the route table are significant for further operations. These information are: Net ID, Mask, and Gateway. As a result file process is started to store these records in another text file.

3.3.4.3 File Processing

The route table is stored in a text file. The information within this file must be processed to get the most significant information needed for other functions.

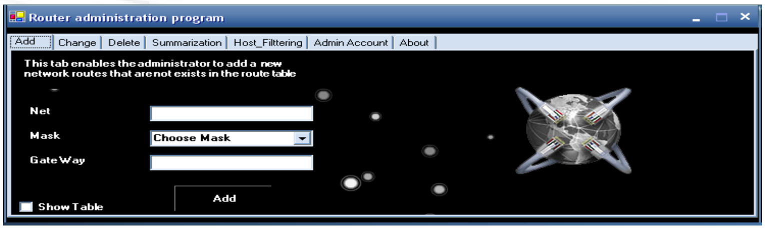
Each line in this file is divided into three strings and stored as an array of structure, which has three fields of string class. These fields are: Net ID, Net Mask, and Gateway. Filling this array is repeated before a determined operation is applied at this file These operations might be route addition, changing, deletion, IP filtering, or summarization.

3.3.4.4 Add Route

This process is applied to the table in order to add a new route. The following are the addition steps, as shown in figure (8):

Step1: Enter the route to add, with its three fields, Net ID, Net Mask, and Gateway.

Step2: Compare this route with the existing routes, if it matches any other route, a message is displayed to notify that this route already exists; otherwise the route is added to the table.



This tab enables the administrator to add a new network routes that are not exists in the route table

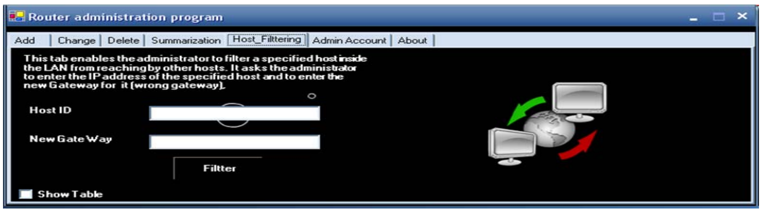
Figure (8): Add New Network Routes Tab.

3.3.4.5 Host Filtering

Host filtering is one of the best ways to implement security to a specified host or a specified subnet inside the network. Filtering is done through giving a specified Net IP or Host IP a wrong gateway inside the route table, in order to prevent connections to the desired destination (host or subnet). This type of filtering is used especially to isolate a specified network from reaching a special server that supports a very important service inside the network like DHCP server for example. The mask value for this IP is always 255.255.255.255. Host filtering is shown in figure (9).

Step1: Enter the IP with its wrong gateway to add.

Step2: Compare this IP with the existing IPs, if there is another identical IP, a message, declares that this route already exists, is shown; otherwise the new IP is added to the route table.



This tab enables the administrator to filter a specified host inside the LAN from reaching by other hosts. It asks the administrator to enter the IP address of the specified host and to enter the new Gateway for it (wrong gateway).

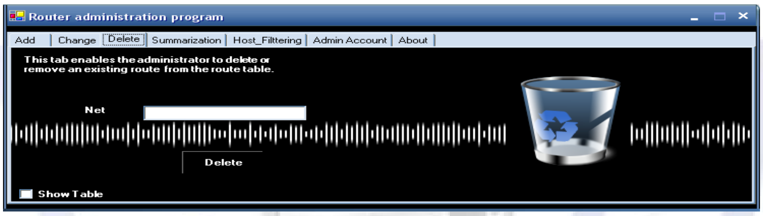
Figure (9): Host Route Filtering Tab.

3.3.4.6 Delete Route

Deleting a specified route is done through the following steps, as shown in figure (10):

Step1: Enter the net ID to delete.

Step2: Search the table for this ID. If this ID is found, then the route is deleted, otherwise an information message is displayed to inform the user.



This tab enables the administrator to delete or remove an existing route from the route table.

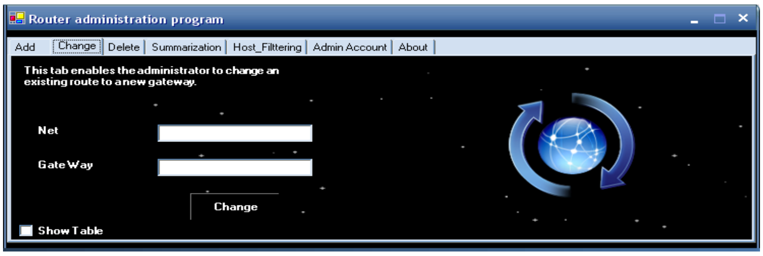
Figure (10): Delete Route Tab.

3.3.4.7 Change Route

Changing a route implies that modifying its gateway. Hence, net ID shouldn't be changed. Change process takes the following steps, as shown in figure (11):

Step1: Enter the net ID and its new Gateway to change.

Step2: Compare the net ID with the table contents, if the ID doesn't exist, a message must be displayed for information. If the ID is found, its gateway is compared with the new gateway, and if matching is occurred, then another message is shown to notify the user that this route is replicated again. Other case, route alteration is performed.



This tab enables the administrator to change an existing route to a new gateway.

Figure (11): Change a Route Tab.

3.3.4.8 Route Summarization

As mentioned before, the main system objectives is to achive route summarization in static routing, this is accomplished on the router's table records having the same gateway. This process is applied depending on a certain mechanism discussed through the following steps:

Step1: A new array of structure is allocated depending on the number of Net IDs stored in the table.

Step2: A batch file is created to save the new information. Some records that are contained within the previous file are excluded from summarization process. These are the default gateway, the router IPs, the loop-back IP, and the host route.

The host route records should be saved in the newly created file, as they wouldn’t be taken in the consideration of summarization process.

Step3: Sorting array elements in ascending order depending on the gateway. This procedure would facilitate the search process for equivalent gateways.

Step4: Comparing the gateway of each record with the one of the next record. If they are similar, they stored in the same route group.

Step5: the resultant groups are then converted to a new array of structure with the same original three elements, but of byte type. This operation is necessary because of the summarization process that needs to deal with bits.

At the beginning of route summarization each group count is tested. If this count equals to one then there is no summarization applied to this group; differently, if the count is greater than one, then the most significant bit of the byte for all groups is compared together.

The value of each tested bit is calculated for both, Net ID and Mask. All bits’ values are compared and the operation of byte right-shift is repeated until a distinct bit is occurred.

Reaching different bits implies setting the ultimate values for both ID and its Mask. The remainder byte values, if any, are set to zero value. And, of course, the gateway values would stay as they are.

After completion of this process, the array of byte groups would be filled with the summarized route for each group.

Step6: Finally, the above array is stored as route addition instructions in the batch file created at the beginning of summarization process.

Figure (12) shows the route summarization tab.

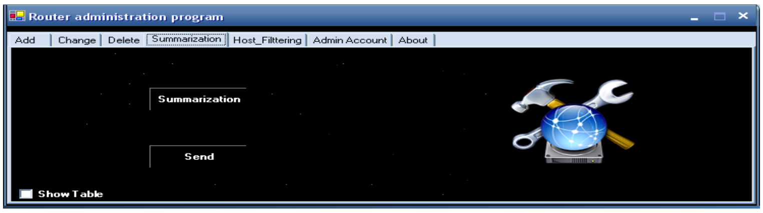


Figure (12): Summarization Tab.

3.3.4.9 Sending the File to Router

The sending process is start after saving the outcome instructions from summarization process in a batch file, i.e. after the administartor modify the route table. This file will be sended to the router to be executed to modify the route table.

Step1: Declare a new object of Stream in order to open the file in read mode, hence, to assign the information related to this file into the specified object.

Step2: Copy the information to an array of bytes using read function, in order to send this array through a socket.

Step3: Open a TCP/IP connection on port 7002 to initialize a connection with the destination and then to send data stored in the former buffer using the NetwokStream class.

3.3.5 Router Program

Router program is installed and stayed resident in the router. It sends the route table to the administration computer, as the latter handles this table using the router administration system. After finishing this process, the router administration system sends the outcome file, the one that router program receives. As mentioned previously, the router program could be implemented in two modes, manual as in figure (13) and auto mode as in figure (14). The steps below performed in auto-mode:

Step1: The program receives a message from the administration computer requesting the route table.

Step2: Receiving the message comprises the program to run a determined process, which creates a new text file and store the route table information in it. Next, the text file is sent to the administration computer to start several operations on it. The steps for sending the file are repeated again.

Step3: The program stays waiting for the network administrator to accomplish handling the table.

Step4: And the last but not the least step is to receive the batch file from the Router Administration System. This file will contain all the instructions needed to modify the route table according to the updates that have been done by the administrator. Now a new process is designed to execute this batch file in order to create the new route table.

In manual mode the only difference is that the administration computer IP is entered manually, in order to send the route table to the administration computer. Also, all other operations are done through pressing buttons to activate their suitable functions.

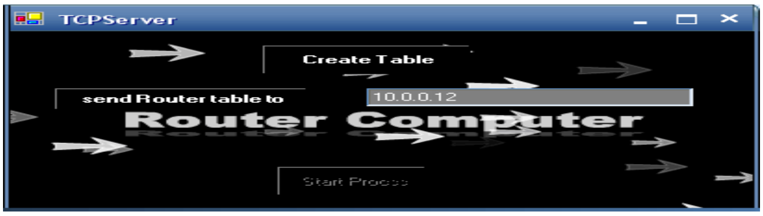


Figure (13): Manual Mode.

Figure (14): Auto Mode.

4.1 System Preperation

The preperation phase will show the main demands that are needed to guarantee the application of the system.

As a first step for implementing the system, we need to activate the routing services in the router computer. The activation will be done using the following path:

Start → Programs → Administrative tools → Routing and Remote Access Services → Right Click on Server → Enable.

5.1 Discussion

The system was designed and implemented on a hypothetical LAN consists of three computers, one of them as a router supported by the router program having Windows Server 2003 Enterprise edition installed on it. One of the clients works as a client communicating remotely with the router, supported by the router administration system. After that, the system was tested on a LAN consists of twelve computers, six of them were working as routers with multiple network routes.

Static routing commands were also used to change the routes of the router through opening a socket between the router computer and the client. The client will receive the route table and the network administrator will change the contents by adding, deleting, changing. Then the network administrator will run the route summarization algorithm to optimize the route table. After that the client will send the route table to the router and the router will depend on this new table instead of the old one.

5.2 Conclusions

After running this system, the network administrator will be aware of all burdens arise from network administration especially in static routing in a LAN or WAN.

Just as we know, dynamic routing like OSPF or RIP supports route summarization automatically without any burdens on the administrator. This system gives the power to the network administrator to achive route summarization in ststic routing just like dynamic routing.

Routers programming depens on Telnet service (port number 21) whcih is always attacked by the hackers, therefore using this system with a port number not known by the hackers will make the router in a safety side.

5.3 Suggestions

The research suggestions for the future works are as follows:

1- Building the system on a hardware router.
2- Support for IPv6.

References

1- Forouzan B. A., **"Data Communications and Networking"**, McGraw-Hill, 4th Edition, 2007.

2- Cisco Networking Academy Program, **"Cisco Certified Network Associate CCNA Semester 1"**, version 3.1.1, 2003.

3- Comer D. E., "**Computer networks and Internets with Internets applications"**, 4th Edition, Prentice-Hall, Inc., 2003.

4- Reid F., **"Network Programming in .NET With C# and Visual Basic .NET"**, Elsevier Digital Press, 2004.

5- Wikipedia web site, http://en.wikipedia.org/wiki/Routing_table

6- Forouzan B. A., **"TCP/IP Protocol Suite"**, McGraw-Hill , 2nd Edition., 2003