IP Monitoring and Filtering

By
Gnanambal Chithambaram
Sandeep Dubey
Smrithi Barrenkula
Subraja Krishnamurthy
Sucheta P Kodali

Abstract

In our project IP Monitoring and filtering we developed a Java application to extract source IP address from TCP packet header. The IP address is tracked along with geographical location and registration data which is aggregated from ARIN WHOIS database and stored locally. The monitoring capabilities are enabled via an administrator website which allows the administrator to monitor and filter IP addresses accessing the website. The application is implemented based on a combination of libraries like Jpcap, WinPcap to listen, log and process the network traffic. Active Server Pages were used to build the administrator website

1. Introduction

As websites gain popularity so does traffic from unwanted visitors, it's the responsibility of the site administrator to monitor and filter malicious IP addresses. The project provides tools to the administrator to filter malicious IP addresses.

In this project we have developed an application that tracks, logs and audits website user IP addresses, domain and geographical location. We have develoedp a simple web site that is hosted on IIS(web server) which provides an option for our java utility application to track users who are accessing the website. The webserver hosts two websites one "Project Home" which is accessible to all users and the second one is a restricted access "Admin" website. Security features have been built into the "Project Home" to redirect requests to appropriate pages based on request IP address.The administrator can use these details to block IP address range and perform an application level filtering.

2. Software components used

Programming Languages:	Java (JDK 1.5), HTML, ASP 2.0
Libraries	Jpcap version 0.5, WinPcap version 3.1
Web server:	MYSQL 5.0
Database:	MYSQL 5
IDE	Eclipse 3.1

Table 1.1 Software Components

3. Hardware Components Used:

Number of Computers	2 personal computers or professional grade servers
Processor	Pentium 4 2.8 GHz or greater
Memory	512 MB of RAM or greater
Disk space	100 GB of available hard-disk
Display	Super VGA (800 × 600) or higher-resolution monitor
Operating system	Microsoft Windows® 2000 with Service Pack 3 (SP3) or later, or Windows XP,Professional or Windows 2003
Internet Connection	Internet connectivity with a static IP is required for users to access the website from different computers
Networking	2 computers were connected via cat 5 cables to a hub/switch so that they could be on their private network

4. System layout diagram

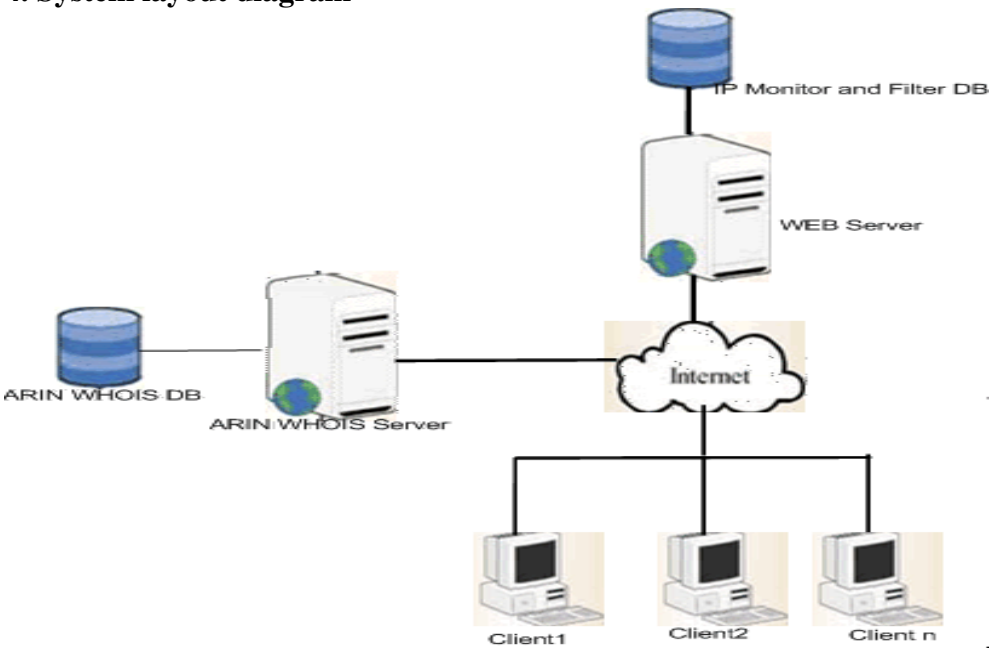


Figure 1.1 System Layout diagram

5. Project Architecture diagrams

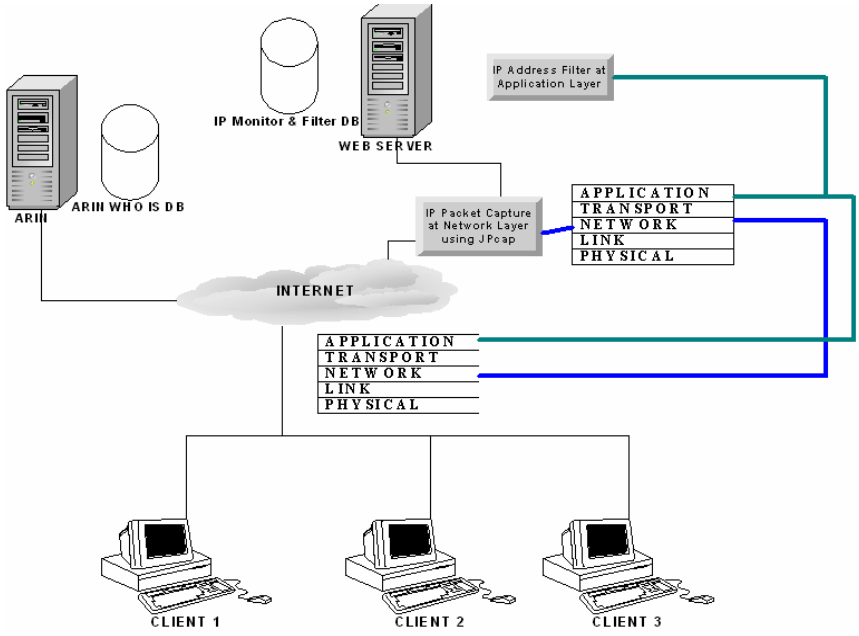


Figure 1.2 Project Architecture diagram

6. Project Description

6.1 Capturing the Packet header data

Our IP monitoring application is a Java application that captures the IP address from the TCP header of all incoming packets. We used Java class Jpcap to capture the IP address.

Jpcap is a Java class package that allows Java applications to capture and/or send packets to the network. Jpcap is based on libcap/wincap and Raw Socket API. Therefore, Jpcap is supposed to work on any OS on which libcap/wincap has been implemented. Jpcap supports the following types of packets: Ethernet, IPv4, IPv6, ARP/RARP, TCP, UDP, and ICMPv4. In our project we used winpcap to make Jpcap work.

The program on initialization acquires information regarding all connected devices. Advanced information like source address, Destination address, Mac address, Device description and Data link are obtained. All the devices are stored in an array. Each device (adaptor) in the array is opened and a timeout is set for each device. The packets in the device are captured. Filter is set to filter the traffic i.e. to get only the TCP packets.

By interpreting the packets, the details of the packets are obtained. The contents are printed onto the screen. Below is a sample output:

1152761003:90970 /67.15.230.17 → /192.168.0.102 protocol(6) priority(0) hop(48) offset(0) ident (42237) TCP 80 > 1139 seq (3436737781) win(7224) ack 1625370999

Once the source port, destination port, Source IP Address and Destination IP Address are captured from the header, the device is closed. The above steps are repeated for each device and the capturing is a continuous process.

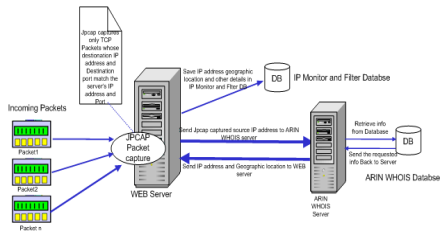


Figure 1.3 Packet Capturing Mechanism

6.2 Getting Geo-location and other relevant data from IP address:

The captured IP address is queried against the ARIN WHOIS database to get all information about the IP like, the organization's name, city, country etc and stored in a MySQL database. The database schema is provided in Appendix A. All requests to the web server are logged and stored into a database which is later analyzed by the administrator.

The webserver hosts two websites one "Project Home" which is accessible to all users and the second one is a restricted access "Admin" website. Security features have been built into the "Project Home" to redirect requests to appropriate pages based on request IP address. The rules for the redirecting the requests are defined by the site administrator using the "Admin" website.

Any request originating from an IP address which was previously blocked by the Administrator would be redirected to a blocked IP address webpage, requests from non blocked IP addresses are redirected to a allowed IP address webpage.

6.3 Application Modules

IP Monitoring application has following modules:

- Administrator Login Module
- View IP Logs
- View Blocked IP
- View Allowed IP
- View Filtered IP
- Manage IP
- Search IP
- Setup module
- Add user module
- Client Module

6.3.1. Administrator Login Module

- Administrator enters username and password in static login page running on the Web server.
- The login page then sends the user information to the login verifying page which checks the validity of the provided username and password.
- If any information is missing, it routes back to the login page with an error message with missing details.
- If both entries are made, the web server queries the database to check whether the user name and password are valid.
- If it is an authorized administrator, the admin is directed to the main page otherwise the web server routes back to the login page.

Once the admin logs in, he has the following options:

6.3.2 View IP Logs

This option allows administrator to view the IP logs.

- When this option is selected, the web server queries the database to generate a list of IPs of the clients that have visited the website. The report also contains the organization's name, time and date of the visit.
- This provides the following features:
  - view details of a particular IP
  - Block a particular IP

6.3.3 View Blocked IP

- When the administrator chooses the view blocked IP option, the web server queries the database to generate a list of blocked IP addresses.

6.3.4 View Allowed IP

- When the administrator chooses the view allowed IP's option the web server queries the database to generate a list all allowed IP address.

6.3.5 View Filtered IP

- When the administrator chooses the view filtered IP option, the web server queries the database to generate the list of all IPs which are not there in either allowed list or blocked list.
- These are the IPs for which the administrator has to make a decision whether to block or allow visiting the website. This happens during the "BLOCKED" mode.
- The administrator has an option to either add these IP's to blocked list or allowed list. Based on the administrator's decision the IP's are added in the database to appropriate lists.

6.3.6 Manage IP

This happens during the "BLOCKED" mode

- This option helps the administrator to add or delete IP from allowed list or blocked list.
- When the administrator chooses to add the IP to allow list, the web server queries the database to see if the IP exists in the block list.
- If it is not in blocked list, the IP is added to the allowed list in the database.
- If the IP exists in the blocked list, the administrator can either proceed and add the IP to the allowed list or deny the action.
- If the administrator chooses to proceed adding the IP to the allow list, the IP is removed from blocked list and then added to the allowed list in the database.
- If administrator chooses to deny his previous action, then the IP remains in blocked list and no IP is added to the allowed list.
- Similar set of actions take place when the administrator chooses to add a new IP to the block list.

6.3.7 Search IP

- When the admin chooses the search IP option, he is provided with an option to search all the details of a particular IP address.
- If the IP exists in the visited logs, the admin is provided with the list of all the details of the IP which can help him for future decisions for allowing or blocking.
- If the IP does not exist, he is simply provided with a statement that no one with that particular IP visited their website.

6.3.8 Setup Module

- Using this module the administrator can operate the "Project Home" website either in "BYPASS" or "BLOCKED" mode.
- The "BLOCKED" mode provides an additional feature to block new IP addresses that are visiting the site. These IPs are granted access to the website only after the administrator authorizes them.

6.3.9 Add User Module:

- This allows administrators to create new users to access "Admin" website, that is the administrator side of the application.

6.3.10 Logout Module

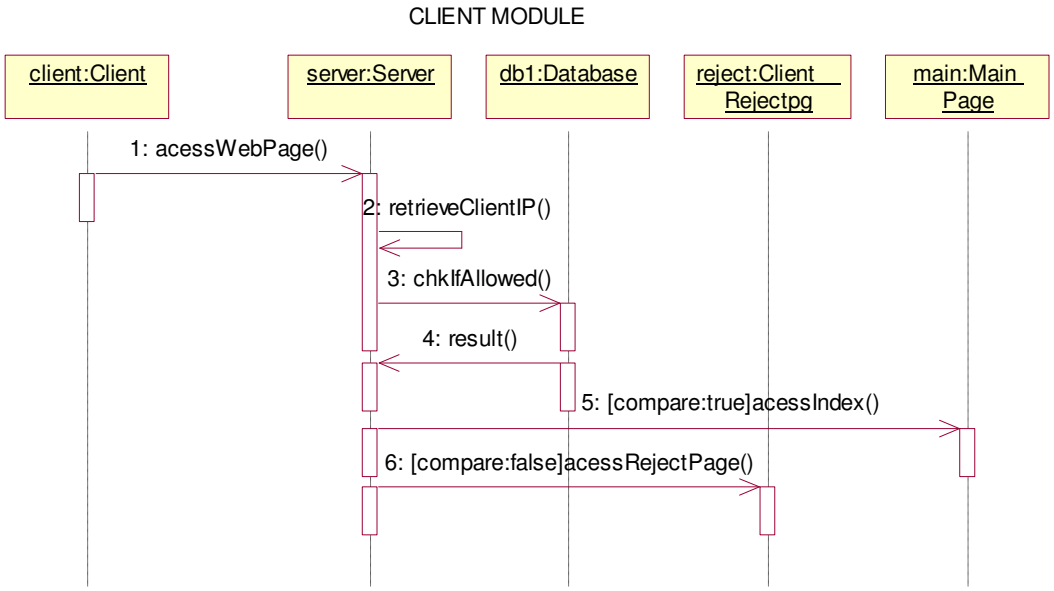
- When the administrator chooses to logout, he is directed to the main login page and his session ends.

6.3.11 Client Module

Once the IP address is captured by the application, the request is forwarded to the web server. Security features have been built into the "Project Home" to redirect requests to appropriate pages based on request IP address. Server side scripting has been used to recapture the request IP address from the following two variables "HTTP_X_FORWARDED_FOR" and "REMOTE_ADDR", the IP address is then queried against the database to determine if the request needs to be redirected to a blocked webpage or a authorized webpage. The blocked webpage displays the following message "Welcome client. Your IP 192.168.1.100 has been granted access to view the website" and the authorized webpage displays the following message "Welcome client. Your IP 192.168.1.100 has been blocked ".

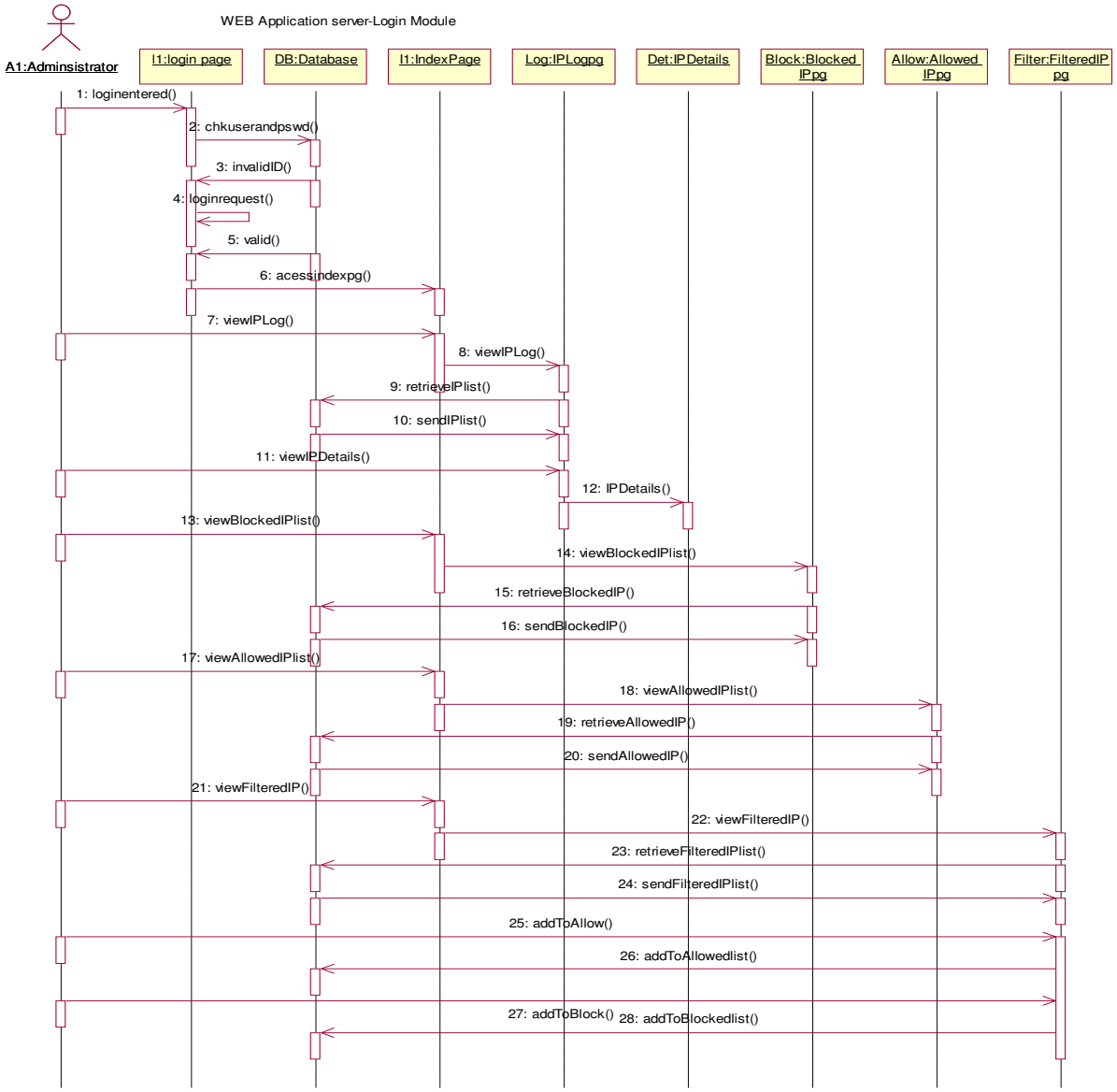
7. Sequence Diagrams:

7.1 Client Module:



Step Numbers	Command	Explanation
1	acessWebPage()	The client tries to access the web page.
2	retrieveClientIP()	The server retrieves the client's IP.
3	chkIfAllowed()	server queries the database to see if the Client is allowed access.
4	result()	The database returns the result for the above query(command).
5	[compare:true]accessIndex()	If the result is positive then the client is given access and the index page is displayed.
6	[compare:false]acessRejectPage()	If the result is negative the client is denied access and is redirected to the reject page.

7.2 Administrator Modules:



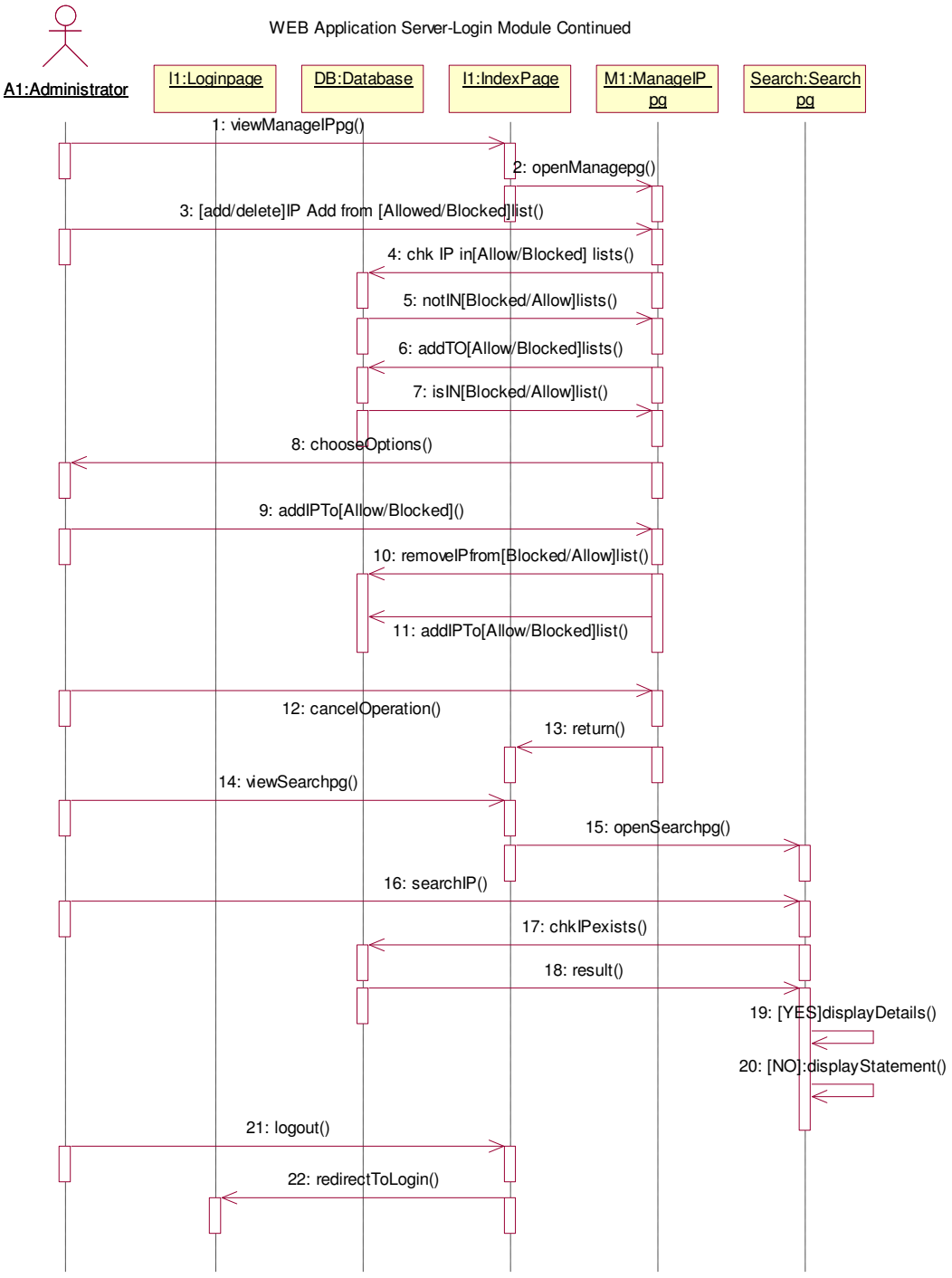
Step Numbers	Command	Expanation
1	loginentered()	The administrator enters the login.
2	chkuserandpswd()	The entered user and password is checked with the valid login data from the database.
3	invalidID()	If the login data entered doesn't match the actual valid data, an invalid command is sent by the database to the webserver.
4	loginrequest()	If the login data is invalid the login page is reloaded and the administrator is requested to enter valid data.
5	valid()	If the entered login data is valid then a valid() command is sent to the web

		server.
6	accessindexpg()	The Administrator is directed to the index page.
7	viewIPLog()	The Administrator chooses the view IPLog option from a list of options.
9	retrieveIPList()	The web server queries the database to retrieve the list of visited IPs.
10	sendIPlist()	The database provides the list of Client's IPs that viewed the site.
11	viewIPDetails()	The administrator chooses the option of viewing IP details.
12	IPDetails()	The administrator is redirected to the IPDetails web page where all the details of the Client's IP are specified.
13	viewBlockedPlist()	The administrator chooses the view Blocked IP's option.
15	retrieveBlockedIP()	The databse is queried to retrieve all blocked IPs.
16	sendBlockedIP()	The generated list of all blocked IPs is sent to the BlockedIPpg.
17	viewAllowedIPlist()	The administrator chooses the view Allowed IP's option.
19	retrieveAllowedIP()	The databse is queried to retrieve all Allowed IPs.
20	sendAllowedIP()	The generated list of all Allowed IPs is sent to the AllowedIPpg.
21	viewFilteredIPlist()	The administrator chooses the view Filtered IP's option.
23	retrieveFilteredIP()	The databse is queried to retrieve all IPs which are not there in either Allowed or Blocked IP's lists.
24	sendFilteredIP()	The generated list of all Filtered IPs is sent to the FilteredIPpg.
25	addToAllow()	The administrator chooses the option of adding the filtered IP to the allowed list.
26	addToAllowedList()	The selected Filtered IP is added to the allowed IP's table.
27	addToBlock()	The administrator chooses the option of adding the filtered IP to the Blocked list.
28	addToBlockedList()	selected Filtered IP is added to the Blocked IP's table.

The above sequence of steps of the Administrator module continues below :

Step Numbers	Command	Explanation
1	viewManageIPpg()	The administrator chooses the manage IP option.
3	[add/delete]IP from [Allowed/Blocked]list()	The administrator chooses the option of adding or deleting an IP from the [Allowed/Blocked]lists.
4	chkIPin[Allowed/Blocked]list()	The database is queried to check if the selected IP is either in allowed or Blocked lists.
5	notIn[Blocked/Allow]lists()	If its not in the [blocked/Allowed] lists this command is sent.
6	addTO[Allow/Blocked]lists()	If its not in the [blocked/Allowed] lists the IP is added to the [Allowed/Blocked] tables.
7	isIN[Blocked/Allow]lists()	This command is sent when the IP exists in [Blocked/Allow]lists.
8	chooseOptions()	The administrator is provided with certain options.
9	addIPTo[Allow/Blocked]lists()	The administrator chooses to add IP in [Allow/Blocked]lists.
10	removeIPfrom[Blocked/Allow]lists()	The IP is removed from its present location.
11	addIPTo[Allow/Blocked]lists()	The IP is added to the selected list or the action is denied.
12	cancelOperation()	The administrator selects the cancel option.
13	return()	The control is returned to the Index page.
14	viewSearchpg()	The administrator selects the option of viewing the search page.

16	searchIP()	The Administrator searches for an IP.
17	chkIPexists()	The database is queried to check the presence of the IP entered by the Administrator.
18	result()	The database returns the result of the above query(command).
19	[Yes]displayDetails()	If the result is positive then the IP's details are displayed.
20	[No]displayStatement()	If the result is negative then a statement is provided.
21	logout()	The administrator chooses the logout option.
22	redirectToLogin()	The control is redirected to login page.


WEB Application Server-Login Module Continued
A1:Administrator
I1:Loginpage
DB:Database
I1:IndexPage
M1:ManageIP pg
Search:Search pg
1: viewManageIPpg()
2: openManagepg()
3: [add/delete]IP Add from [Allowed/Blocked]list()
4: chk IP in[Allow/Blocked] lists()
5: notIN[Blocked/Allow]lists()
6: addTO[Allow/Blocked]lists()
7: isIN[Blocked/Allow]list()
8: chooseOptions()
9: addIPTo[Allow/Blocked]()
10: removeIPfrom[Blocked/Allow]list()
11: addIPTo[Allow/Blocked]list()
12: cancelOperation()
13: return()
14: viewSearchpg()
15: openSearchpg()
16: searchIP()
17: chkIPexists()
18: result()
19: [YES]displayDetails()
20: [NO]:displayStatement()
21: logout()
22: redirectToLogin()

8. Related Project Work

People also use packages like SharpPcap and Dot net for packet capturing. IP monitoring and filtering is commonly used by network administrators for controlling and managing the incoming traffic. It is widely used by websites to ensure the security and integrity.

9. Future Enhancements

We researched internal functioning of Firewalls and IP filtering software and determined that to implement a robust IP filtering mechanism we would have to filter IP addresses at the network layer rather than the application layer. This would allow the software to have capabilities of detecting and stopping any unwanted intrusions. Also this provides portability of the application and hence can be used in a variety of scenarios and combined with other applications to enhance functionality.

Future enhancements would involve developing mechanisms that decides which types of IP datagrams are processed and discarded. *Discarded* would mean to delete and ignore completely. Additionally different criteria can be applied to determine which datagrams are processed. Some examples of these are:

- Protocol type: TCP, UDP, ICMP, etc.
- Socket number (for TCP/UPD)
- Datagram type: SYN/ACK, data, ICMP Echo Request, etc.
- Datagram source address: where it came from
- Datagram destination address: where it is going to

In addition to IP address filtering port blocking would need to be implemented to restrict accesses from known as well as unknown IP addresses and provide the administrator additional capabilities to lock down the server from a security perspective.

10. A Study on Firewall:

A Firewall is a piece of software or hardware that is used against an unauthorized user accessing a network. A firewall only allows the Internet traffic that has been specifically permitted onto a company's local network. Firewalls use one or more of the following methods to control traffic flowing in and out of the network:

- Packet Filter: Although this technique is difficult to configure and vulnerable to denial of service attack, it is fairly effective and transparent to users. The way it works is that each packet leaving or entering a network is screened based on the rules set by the network administrator. The packet being mentioned about here is a message that is transmitted over a network and contains both a destination address as well as data.
- IP Addresses: This can be used in two ways. We can block IP addresses of certain web sites from being accessed through a corporate network. The second way is to block the IP address of machine that might pose a security risk. If it is found, for instance which port scans were being done from a particular IP address, it would be better to block the IP Address completely.

Firewalls are of two types

- Network layer Firewall: These types of firewalls make their decision based on the source, the destination addresses and ports. Network layer firewalls maintain internal information about the state of connections passing through them, the contents of some of the data streams and so on. They tend to be very fast and are transparent to users. Network layer operate at a lower level of the TCP/IP control stack, not allowing packets to pass the through the firewall unless they match the rules.

- Application Layer Firewall: These are generally hosts running proxy servers, which permit no traffic directly between and which perform elaborate logging and auditing of traffic passing through them. A proxy server is a server that sits between a client application, such as web browser and a real server. Proxies are often used to prevent traffic from passing directly between networks. These firewalls work on the application level of the TCP/IP stack and may intercept all packets traveling to of from an application.

11. Problems Encountered:

We tried to capture IP addresses and process them from the network port before other applications can capture them, but we were not successful in accomplishing that task. Java open source libraries like Jpcap provides options only to listen to the network port in non blocking mode and enabling blocking option was something we were not able to achieve and so we differed it to future enhancements. That's one of the problems that we encountered which stopped us from actually implementing a firewall in the network layer. So, we did implement something in the application layer that gives the administrator an option to block IP addresses.

12. Conclusion

IP Filtering enables us to set rules to restrict access to certain sections of the web server. By following a policy of only allowing specific connections from known IP addresses, and denying everything else, we can help protect sensitive content from any unauthorized users. This can be used along side other applications and definitely enhances network security. This project has provided us an opportunity to understand the TCP/IP Network layer, its implementation and its functionality. It gave us an opportunity to work with Network packet capture applications like Jpcap/Winpcap that captures certain network packets for analysis.

13. References

[1] E.R.Harold, “Java Network Programming, Second Edition”, O'Reilly Publications,August 2000

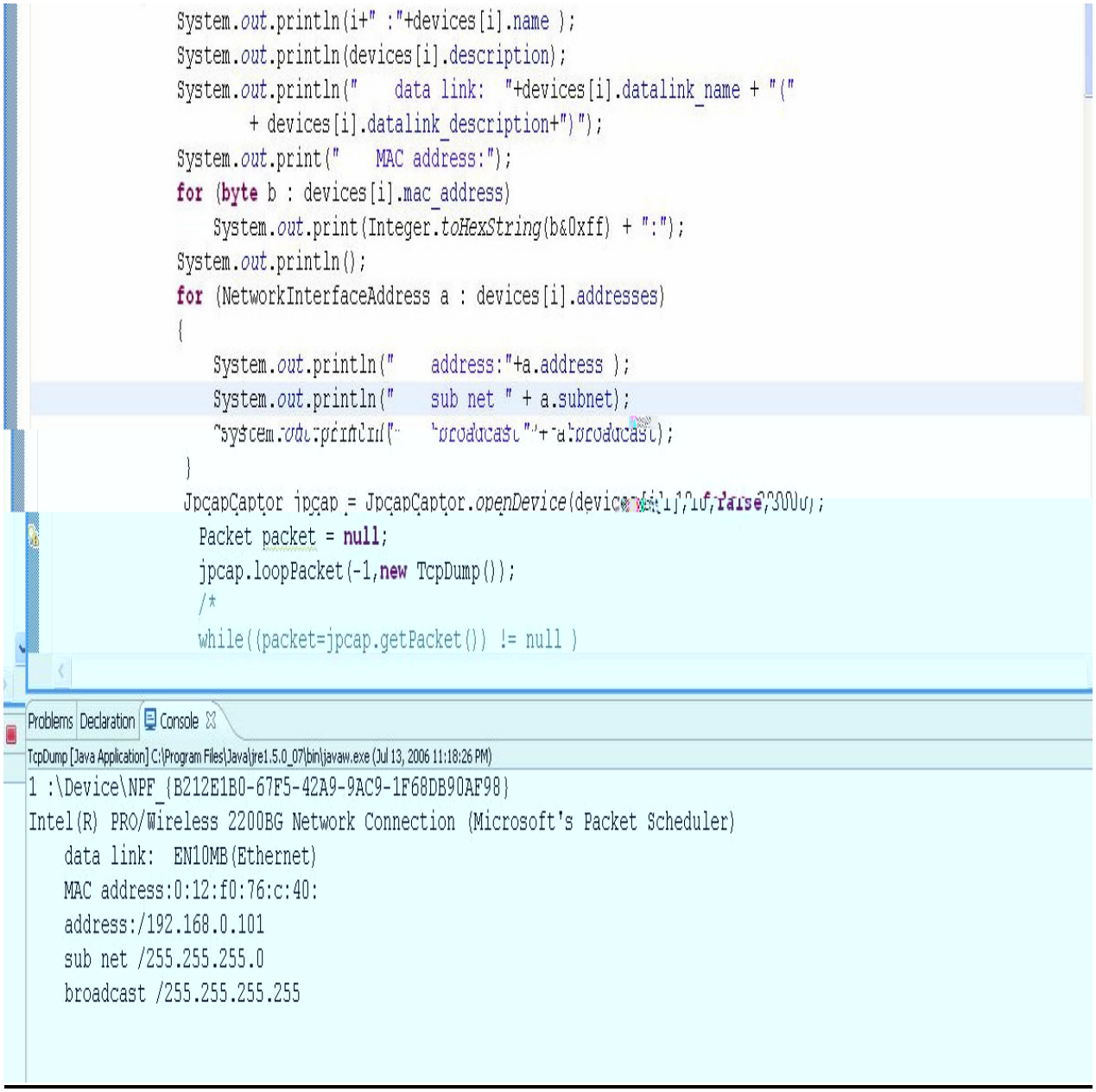
[2] B.Evjen, S.Hanselman, D.Rader, F.Muhammad, S. Sivakumar , “Professional ASP.NET 2.0 Special Edition”, Wrox Publications, September 2006

[3] http://netresearch.ics.uci.edu/kfujii/jpcap/doc/index.html

[4] http://java.sun.com/

APPENDIX A

Screen shots for packet capture:



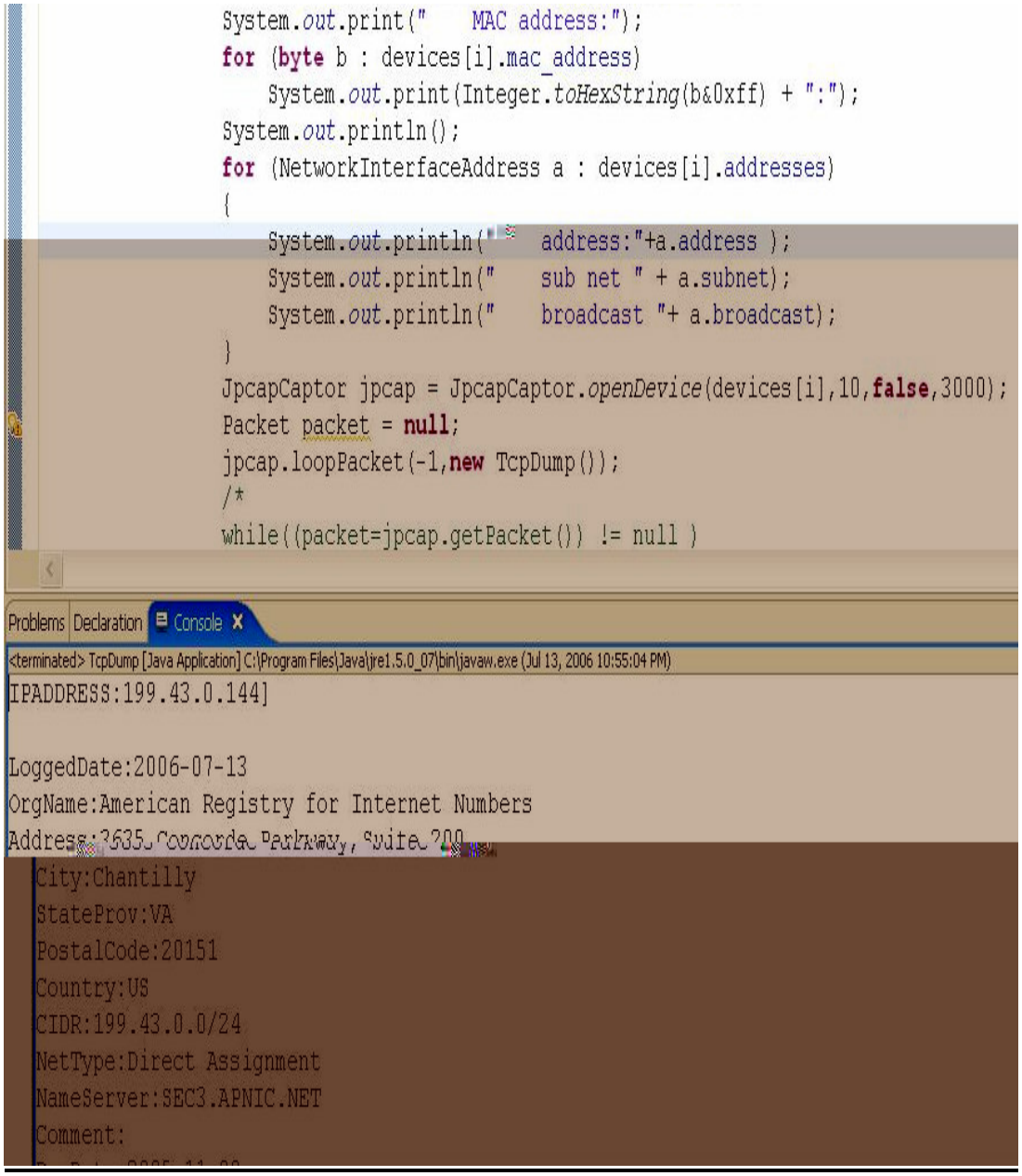
```
System.out.println(i+" :"+devices[i].name );
System.out.println(devices[i].description);
System.out.println("    data link:  "+devices[i].datalink_name + "("
       + devices[i].datalink_description+")");
System.out.print("    MAC address:");
for (byte b : devices[i].mac_address)
   System.out.print(Integer.toHexString(b&0xff) + ":");
System.out.println();
for (NetworkInterfaceAddress a : devices[i].addresses)
{
   System.out.println("    address:"+a.address );
   System.out.println("    sub net " + a.subnet);
   System.out.println("    broadcast "+a.broadcast);
}
JpcapCaptor jpcap = JpcapCaptor.openDevice(devices[i],10,false,3000);
  Packet packet = null;
  jpcap.loopPacket(-1,new TcpDump());
  /*
  while((packet=jpcap.getPacket()) != null )
```

Problems | Declaration | Console

TcpDump [Java Application] C:\Program Files\Java\jre1.5.0_07\bin\javaw.exe (Jul 13, 2006 11:18:26 PM)

```
1 :\Device\NPF_{B212E1B0-67F5-42A9-9AC9-1F68DB90AF98}
Intel(R) PRO/Wireless 2200BG Network Connection (Microsoft's Packet Scheduler)
    data link:  EN10MB(Ethernet)
    MAC address:0:12:f0:76:c:40:
    address:/192.168.0.101
    sub net /255.255.255.0
    broadcast /255.255.255.255
```

Screen shot for acquiring details from ARIN WHOIS database:

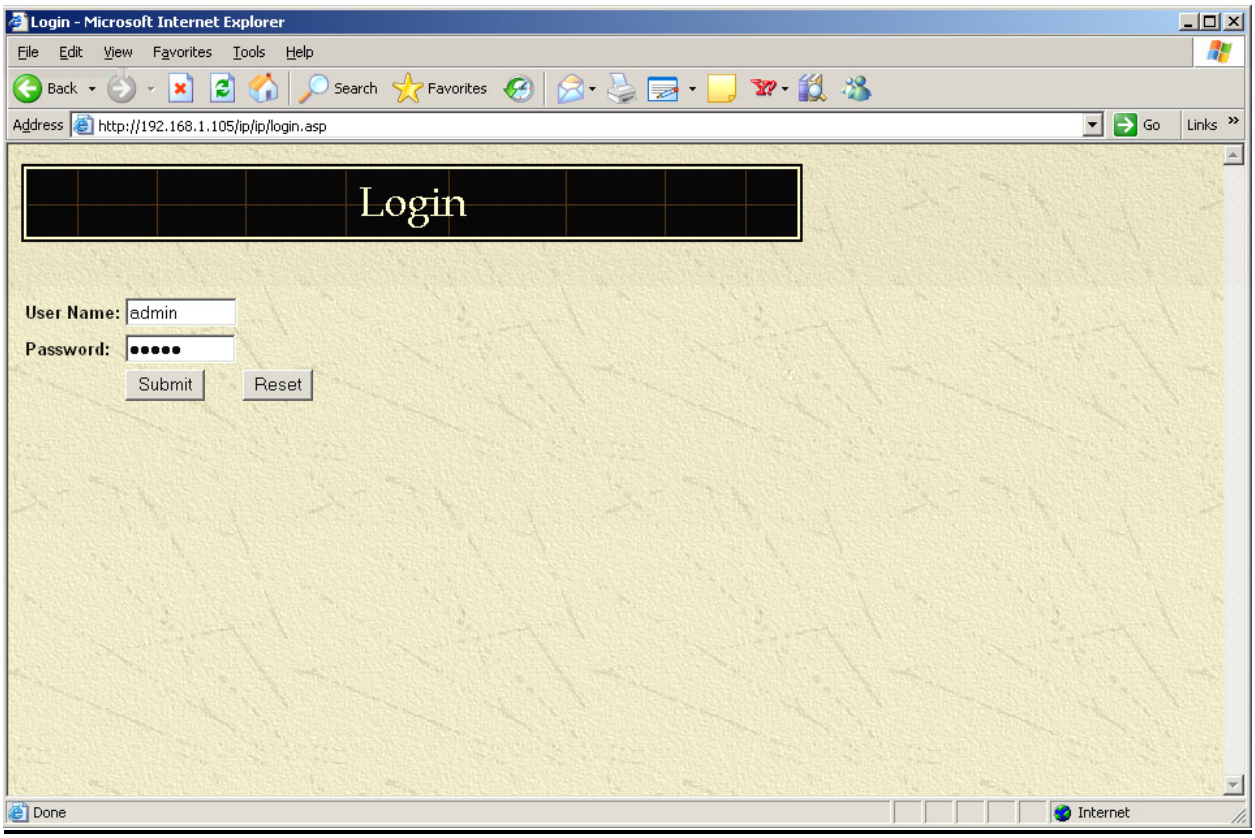


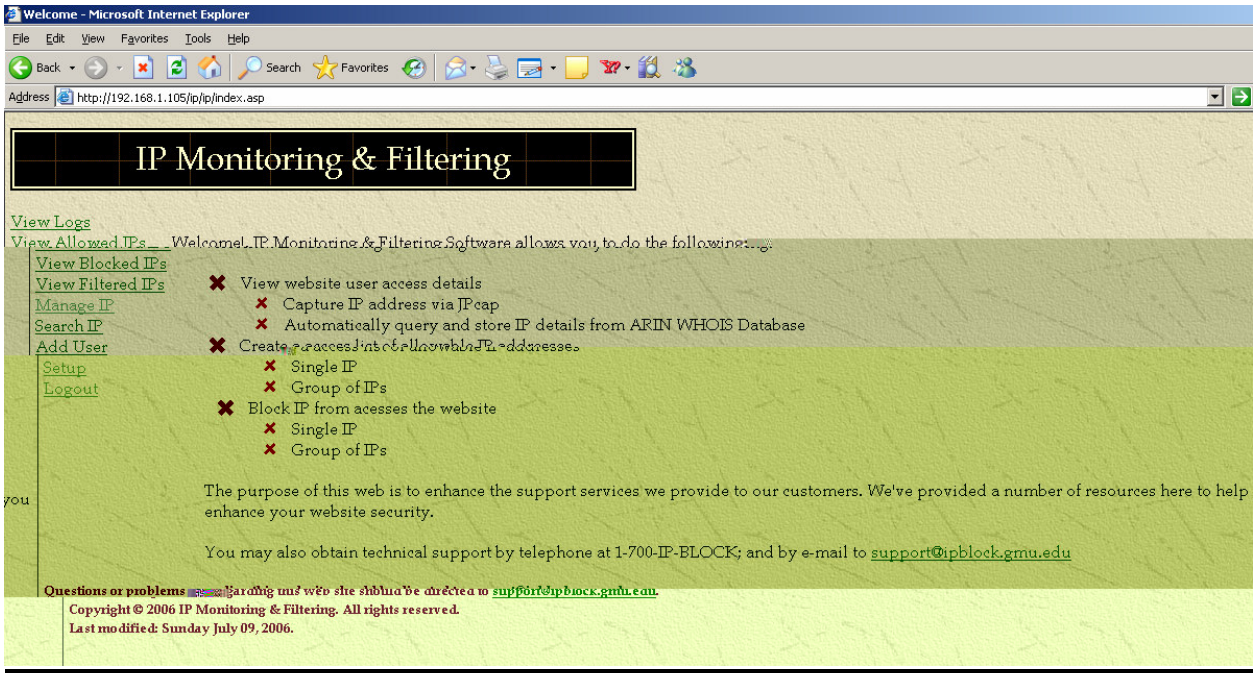
```
                System.out.print("    MAC address:");
                for (byte b : devices[i].mac_address)
                    System.out.print(Integer.toHexString(b&0xff) + ":");
                System.out.println();
                for (NetworkInterfaceAddress a : devices[i].addresses)
                {
                    System.out.println("    address:"+a.address );
                    System.out.println("    sub net " + a.subnet);
                    System.out.println("    broadcast "+ a.broadcast);
                }
                JpcapCaptor jpcap = JpcapCaptor.openDevice(devices[i],10,false,3000);
                Packet packet = null;
                jpcap.loopPacket(-1,new TcpDump());
                /*
                while((packet=jpcap.getPacket()) != null )
```

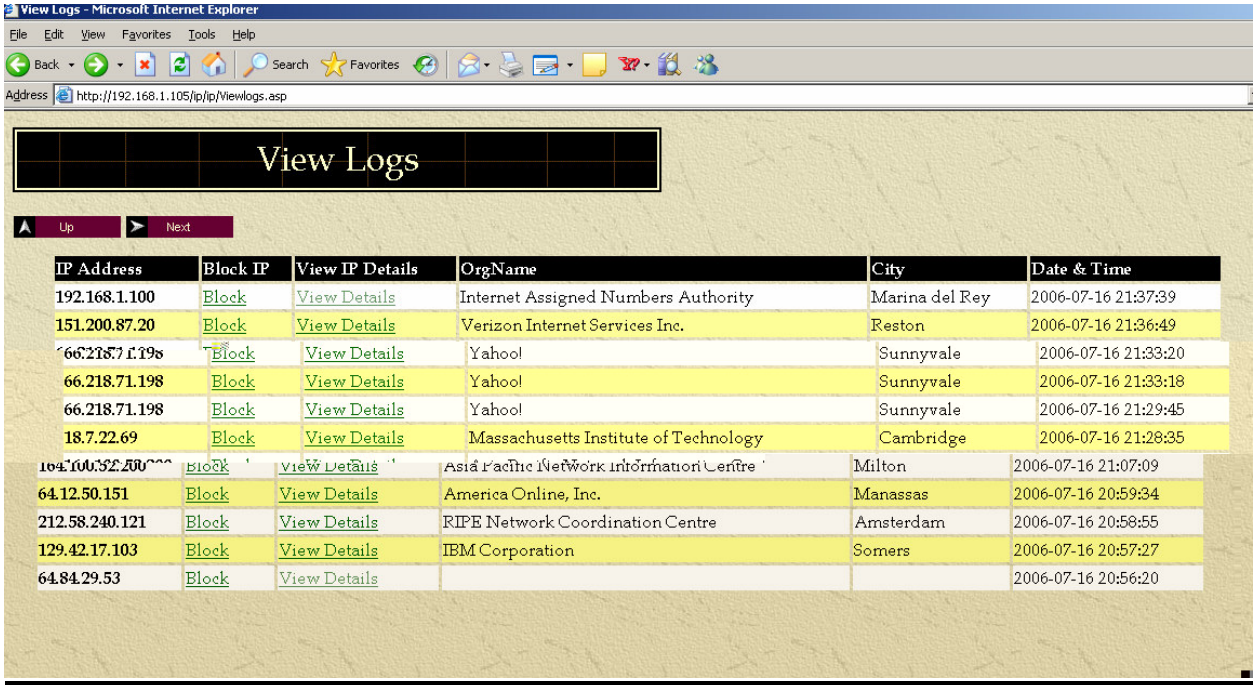
```
Problems  Declaration  Console X
<terminated> TcpDump [Java Application] C:\Program Files\Java\jre1.5.0_07\bin\javaw.exe (Jul 13, 2006 10:55:04 PM)
IPADDRESS:199.43.0.144]

LoggedDate:2006-07-13
OrgName:American Registry for Internet Numbers
Address:3635 Concorde Parkway, Suite 200
City:Chantilly
StateProv:VA
PostalCode:20151
Country:US
CIDR:199.43.0.0/24
NetType:Direct Assignment
NameServer:SEC3.APNIC.NET
Comment:
```

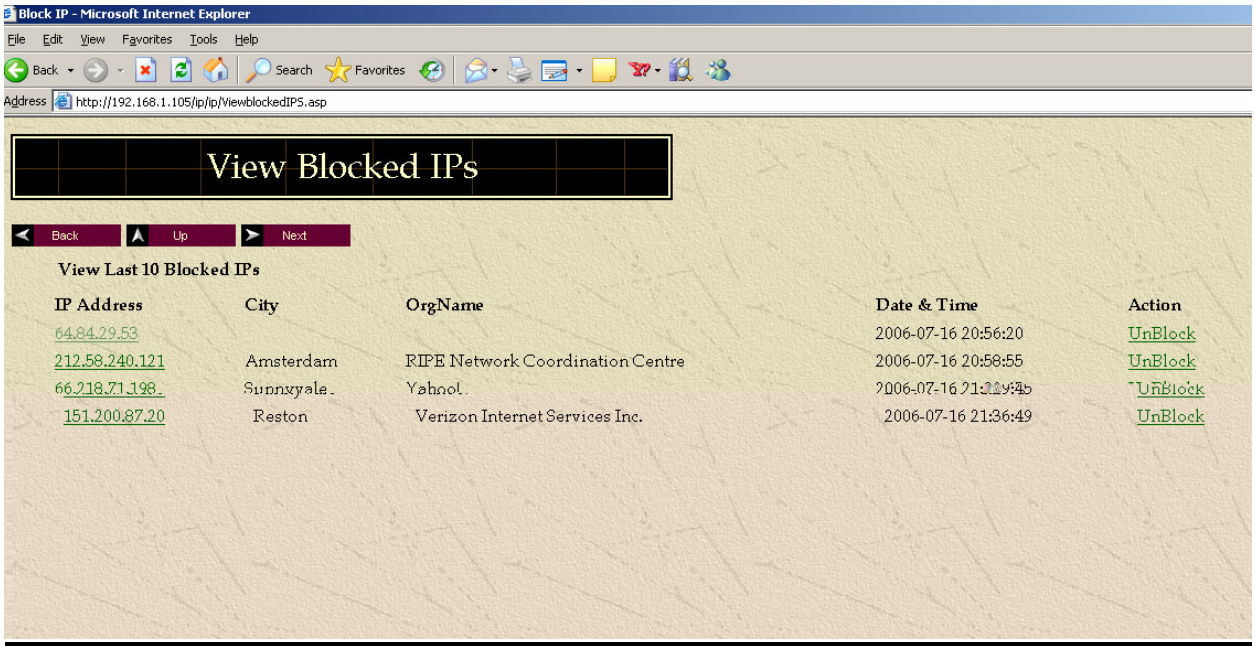
Screen shots for Administrator modules:



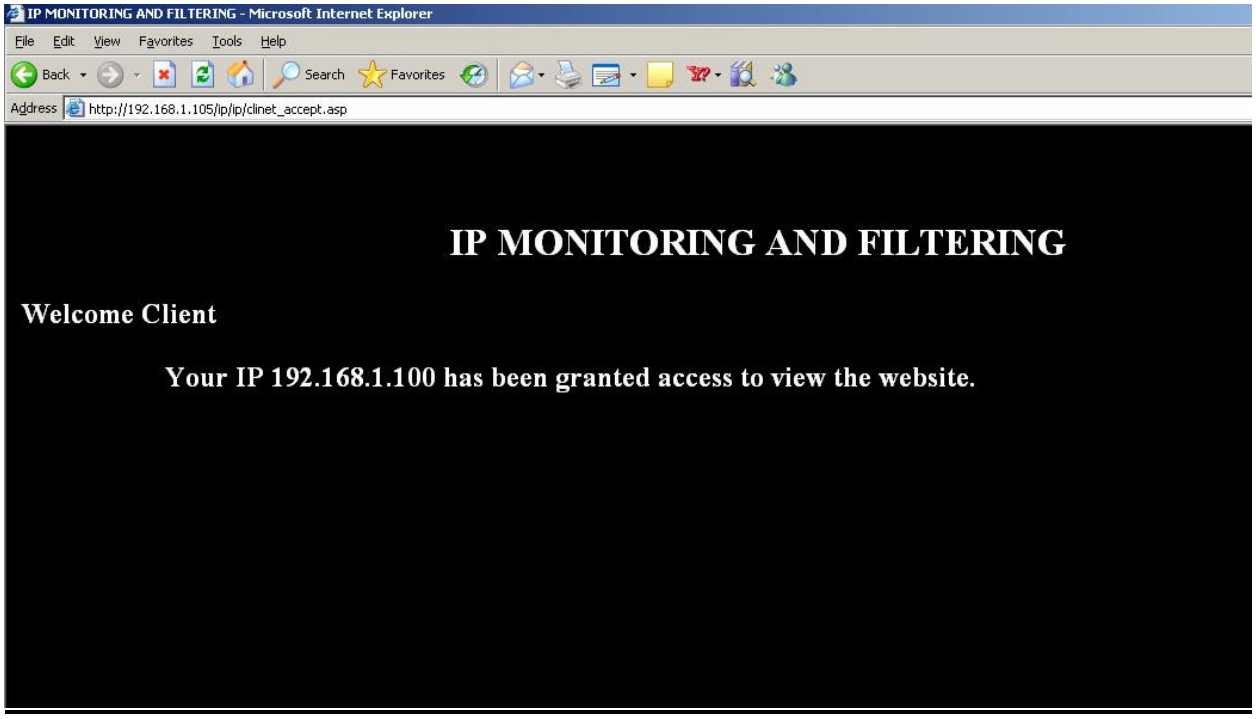


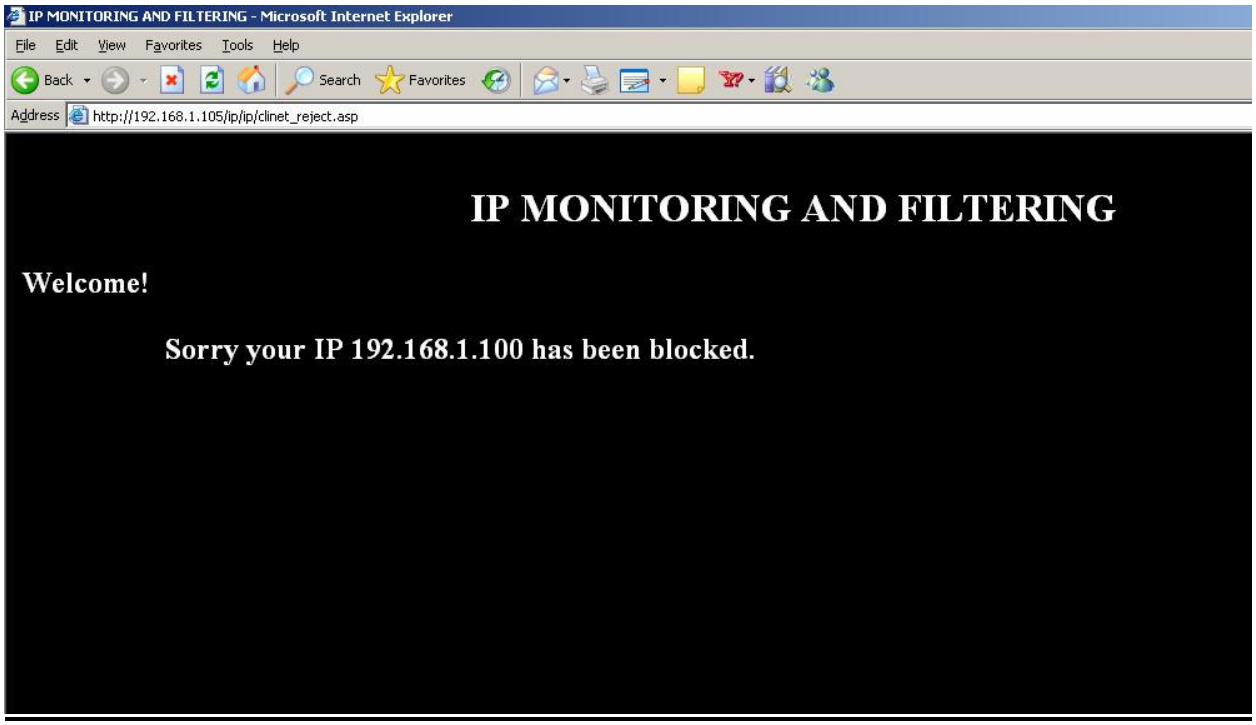


IP Address	Block IP	View IP Details	OrgName	City	Date & Time
192.168.1.100	Block	View Details	Internet Assigned Numbers Authority	Marina del Rey	2006-07-16 21:37:39
151.200.87.20	Block	View Details	Verizon Internet Services Inc.	Reston	2006-07-16 21:36:49
66.218.71.198	Block	View Details	Yahoo!	Sunnyvale	2006-07-16 21:33:20
66.218.71.198	Block	View Details	Yahoo!	Sunnyvale	2006-07-16 21:33:18
66.218.71.198	Block	View Details	Yahoo!	Sunnyvale	2006-07-16 21:29:45
18.7.22.69	Block	View Details	Massachusetts Institute of Technology	Cambridge	2006-07-16 21:28:35
[illegible]	Block	View Details	Asia Pacific Network Information Centre	Milton	2006-07-16 21:07:09
64.12.50.151	Block	View Details	America Online, Inc.	Manassas	2006-07-16 20:59:34
212.58.240.121	Block	View Details	RIPE Network Coordination Centre	Amsterdam	2006-07-16 20:58:55
129.42.17.103	Block	View Details	IBM Corporation	Somers	2006-07-16 20:57:27
64.84.29.53	Block	View Details			2006-07-16 20:56:20



Screen shots for Client modules:


IP MONITORING AND FILTERING - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://192.168.1.105/ip/ip/clinet_accept.asp
IP MONITORING AND FILTERING
Welcome Client
Your IP 192.168.1.100 has been granted access to view the website.


IP MONITORING AND FILTERING - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://192.168.1.105/ip/ip/clinet_reject.asp
IP MONITORING AND FILTERING
Welcome!
Sorry your IP 192.168.1.100 has been blocked.

APPENDIX B

Database Tables:

TABLE 1 : Database table for IP Logs

Field	**Type**	**Key**
ip_logs_id	int(10)	**PK**
ip_address	varchar(200)	
OrgName	varchar(200)	
Address	varchar(200)	
City	varchar(200)	
StateProv	varchar(200)	
PostalCode	varchar(200)	
Country	varchar(200)	
CIDR	varchar(200)	
NetType	varchar(200)	
NameServer	varchar(200)	
Comments	varchar(200)	
RegDate	varchar(200)	
Updated	varchar(200)	
RTechName	varchar(200)	
RTechPhone	varchar(200)	
RTechEmail	varchar(200)	
OrgTechName	varchar(200)	
OrgTechPhone	varchar(200)	
loggedDate	varchar(200)	

TABLE 2:Database for blocked IPs

Field	Type	Key
block_id	int(10)	**PK**
ipaddress	Varchar(200)	**FK**

TABLE 3:Database for allowed IPs

Field	Type	Key
allow_id	int(10)	**PK**
ipaddress	Varchar(200)	**FK**

TABLE 4:Database for Administrators

Field	Type	Key
User_id	int(10)	**PK**
User_name	Varchar(200)	
Password	Varchar(200)	
User_level	Varchar(200)	

TABLE 5:Database for Set up Mode

Field	Type	Key
Setup_text	Varchar(20)	**PK**

: