Design of an Intranet for IIT Kanpur

A Report Submitted
in Partial Fulfillment of the Requirements
for the Degree of

Bachelor of Technology

by

Siddharth Nautiyal and Kshitij Gupta

to the

Department of Computer Science & Engineering

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

April, 1999

Certificate

Certified that the work contained in the report entitled "*Design of an Intranet for IIT Kanpur*", by *Siddharth Nautiyal and Kshitij Gupta*, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Dr. Pankaj Jalote and Dr. Dheeraj Sanghi

April, 1999

Preface

This report describes the methodology to be followed when designing an intranet for a network like IIT Kanpur's. Chapter 1 gives the motivation behind the project, and provides a brief overview of the problem being tackled. Chapter 2 deals with recommendations regarding the Network Architecture to be employed, together with the rationale for these recommendations. Chapter 3 moves on to describe means which can be incorporated in order to incorporate security into our network.

Acknowledgements

We would like to thank our guides, Dr Pankaj Jalote and Dr Dheeraj Sanghi, for the invaluable guidance, motivation and prodding they provided to spur us on to achieving whatever results we have. We would like to thank Mr Navpreet Singh, ACES, and Dr Sanjeev Agarwal, who provided us with valuable information about IITK's network architecture. We would also like to thank the CSE lab staff for providing us with the facilities to test and run our software.

Contents

Chapter 1

Introduction

The era of standalone machines doing all the world's computing is now almost at an end. For efficient use of information, expensive processing power and peripheral resources, machines need to be linked together by means of a network so that these expensive resources can be shared. Also, as the scope for the use of computers grows, the number of people using these machines will grow by leaps and bounds, as also will the demands on the capabilities of these machines.

These interconnections of computers – *computer networks*, as they are called – are becoming increasingly popular the world over. Amongst other things, they allow interaction between users in physically remote locations, they allow sharing of information between users and machines in different parts of the world, they allow users at one site to use the cheaper, or more powerful, processing power of machines at some other site. The idea behind the increasing deployment of computer networks is simple – it is a much cheaper, much more practical, and much more efficient solution than having entirely self-sufficient, powerful standalone machines.

IIT Kanpur has recognized these factors, and has correspondingly decided to go in for a networked campus. At the logical level, the campus network is composed of various subnetworks. The various academic and research departments have their own subnets – e.g. the CSE network, the CC network, the EE network, and the AE network are some of the larger subnetworks making up the campus network. The design of the network is such as to provide for interaction and communication between these subnetworks. Besides, there is support for dial-up connections from faculty residences. There is also a latent, unsatisfied demand for access to computing facilities to be provided to students' hostels.

Given the nature of the tasks performed in these subnetworks and the strong interdependence between them, it can be expected that voluminous traffic will flow between these subnetworks (e.g. between the AE network and the CC network), and the nature of this traffic might be, to a large extent, confidential. In order to efficiently utilize the resources available, the design of the network should be such as to provide the maximum possible rates of data transfer between the communicating entities, at the same time not compromising on the integrity and confidentiality of the data flowing on the network. An attempt should also be made to expand the campus network to encompass as many users as possible – by providing connectivity to students' hostels, for example, and by increasing the connectivity available to faculty.

Besides, a network of the size of IITK's has to be managed efficiently in order to ensure that all the resources present in the network are available for use by all authorized entities. Care should also be taken that no entity in the network is subverting the security of the network, or is accessing resources it is not authorized to. Also, the network should be designed keeping in mind the almost certain prospect for rapid growth in the future.

In this work, we focus on some of the issues relating to the design of an Intranet for IITK. The topics under which we have studied the design of our intranet are:

- **Architecture**
- **Security**

Though the report discusses these issues from IITK's context, the results and software are mostly general and can be used in other environments as well.

In the rest of this chapter we briefly discuss these two topics.

1.1 Architecture

While designing the architecture of the network, we had to keep in mind the following issues:

- Future campus network : The campus network had to be designed keeping in mind that the future demand for both internal and external connectivity will be much greater than it is at present. This involved studying not only the actual connection technology, but also the IP address allocation policy.
- Layout of the network : While designing the network layout we had to keep in mind the following points
  - Backbone design
  - Technologies to be used for the individual networks connected to the backbone
  - How the individual networks were to be connected to the backbone
- Feasibility of reuse of existing architecture and backbone.
- Widening the pool of users with access to the campus network by providing connectivity to students' hostels and increased connectivity to faculty residences.
- Mobility : Providing ability to access the campus network for nodes not on the campus network.
- The various network technologies which were available for the backbone (Fast Ethernet, ATM, FDDI) and the network technologies which could be considered for connecting end hosts (Ethernet, Fast Ethernet, Dial-up, DSL, ISDN, Wireless).

Decisions regarding the aforementioned parameters were made keeping in mind the cost, speed, bandwidth and quality of service of the main link and of the subsidiary network links.

The output from this section included recommendations for the backbone design, and on the methodology to be used to provide access to students' hostels and faculty residences – backed up with detailed cost-benefit analyses.

1.2 Security

The goal of this module was to make the working environment in the network more secure. In order to safeguard the network a security policy was specified and security decisions were taken on the basis of this policy. The policy dealt with

- kind of access to the network from the outside world
- log keeping and management for audit and analysis of security threats
- providing secure communication to all network users
- ensuring authentication
- checking authorization before granting access to network resources

On the basis of this policy, a network intrusion detection software was developed in order to pinpoint intrusive activity in the network. Also **ssh2** – the latest implementation of the IETF **secure shell** standard – was installed on machines in the network in order to ensure authentication, authorization, and secure communication. Scripts were also developed for regular monitoring of the network in order to detect any system or configuration loopholes. These scripts ran freely available loophole detection software which we configured to run on our network.

Chapter 2

Architecture

This module focuses on trying to determine the network layout and technology that provides the optimal cost-benefit ratio in the IITK setting. We studied various backbone technologies in order to determine which would best serve the campus' need for maximum connectivity at minimum additional cost, and which would best utilize the existing campus network. We also considered various technologies which can be used to provide connectivity to the students' hostels, both to entire hostels and to some portions of hostels. Provision of network connectivity to residences of faculty members was also considered. In each of these cases, an incremental cost-benefit analysis was performed, and the final recommendation was arrived at on the basis of this analysis.

The detailed calculations for students' hostels have actually been performed for a particular hostel on the campus (Hall 1), but the analysis can easily be extended to any hostel, on the campus or elsewhere, by simply varying the parameters relating to the dimensions of the hostel. Also, the technology that emerges from the analysis has such a distinct price advantage that varying the dimensions of the hostel will finally have no real effect on the final recommendation.

In case of faculty residences, calculations were carried out on the basis of actual distances measured from a map of the campus.

In all the cases, the principles underlying the design, and the constraints to be kept in mind, are specified before the analysis is performed.

2.1 Backbone Selection

Most local networks are centered around a backbone. Currently, an optical fiber backbone has been laid in the campus, in a star topology centered at the Computer Center and extended to all academic departments, hostels, and the health center. The fiber is capable of gigabit speeds, and the switches on the network are 10/100 or 10 Mbps, depending on their location (e.g. the central CC switch is 100 Mbps while the switch at the LHC is 10 Mbps). The network can thus support 100 Mbps Fast Ethernet at present.

In this section we consider the issue of selecting an appropriate backbone for IITK. The backbone technology chosen should provide good service at the least possible cost, and should utilize as much of the present backbone structure as possible. The design should also be scalable and should be easily and economically upgradable in order to meet increased needs.

2.1.1 Overview of Technology Options Considered

- *ATM (Asynchronous Transfer Mode)*

ATM technology is based on a circuit switched mechanism, where a connection is established between end points before any data transfer starts and is broken when the data transfer is complete. It uses small constant-sized cells that permit rapid switching in hardware which can be statistically multiplexed together on the same link. ATM protocols are designed to carry both isochronous (time critical) data and conventional data communications. ATM is mostly fiber-based but can be run over any media. It provides inorder delivery of packets but the basic connection oriented layer does not provide reliable delivery.

- *FDDI (Fiber Distributed Data Interface)*

FDDI (ANSI X3T9.5) delivers 100 Mbps over Category 5 UTP (Unshielded Twisted Pair, capable of supporting MHz), multimode fiber (capable of carrying many signals simultaneously), and single-mode fiber-optic cable (capable of carrying only a single signal) . FDDI employs a token passing scheme as its network access method. FDDI can also use a dual-ring architecture, and the combination of these two features makes FDDI a self healing LAN technology. FDDI supports both single attachment station (SAS) to one ring for connectivity, or dual attachment station (DAS) for connectivity and redundancy. In addition, FDDI has a station management (SMT) layer, which defines an extensive set of statistics designed for troubleshooting the LAN. FDDI supports frame lengths up to 4500 bytes for greater throughput. The two greatest advantages of FDDI are its dual fault-tolerant rings and slow degradation of performance with increasing traffic load. It thus provides fast and reliable connections. FDDI is much more expensive than other technologies which provide comparable backbone speeds e.g. Fast Ethernet.

- *Fast Ethernet*

100 Mbps Fast Ethernet is a logical extension to 10 Mbps Ethernet retaining the famous CSMA/CD (Collision Sense Multiple Access / Collision Detection) protocol. It is recognized as the IEEE 802.3u protocol and supports star topology. Fast Ethernet supports UTP cabling and fiber but not coaxial cable. There are three variations - 100 Base FX (100 Mbps Baseband Ethernet on fiber), 100 Base-T4 (100 Mbps Baseband Ethernet on UTP 4-pair) and 100 Base-TX (100 Mbps Baseband Ethernet on UTP, STP 2-pair). Fast Ethernet provides higher bandwidth and the ability to operate on common Ethernet wiring. 10/100 autosensing ports ensure complete interoperability with 10Mbps Ethernet.

2.1.2 Comparison of Technologies

Refer to the attached Table No.1 for a brief comparison of the technologies considered for the backbone.

2.1.3 Recommendation

As the Fast Ethernet backbone is already in place and most of the networks on campus use Ethernet as their LAN technology, the most cost-effective solution is to continue with the present backbone, especially as the bandwidth offered by it is at present far from exhausted (enquiries have revealed that not more than 10 % of the backbone bandwidth is used at any instant). When the need arises, a transition can be made to a gigabit backbone by upgrading all the network switches and hubs.

2.2 Providing Connectivity to the Hostels

The main requirements for networking the hostels are:

- Moderate bandwidth at end points (in this case hostel rooms)
- On-demand connectivity to the rooms or connectivity to all the rooms
- Low cost of installation and maintenance

2.2.1 A Sample Hostel Architecture

In our study, we have performed the cost analysis for providing connectivity to students' hostels by using a particular hall of residence (Hall 1) for our calculations.

This hostel consists of 432 rooms, distributed over 6 wings, with 3 floors in each wing. The height of each floor is around 5m. The optical fiber connection is terminated in the hostel office, and connections to the various wings and thence to the individual rooms are drawn from here

The actual dimensions of the hostel, and its physical layout are shown in the figure 1.

Currently, a fiber optic cable (10 Mbps - from a 10Mbps switch at the Lecture Hall Complex) up to Hall I connected to a hub near the Hall Office through a media converter.

2.2.2 Overview of Technology Options Considered

■ *Ethernet on UTP*

Ethernet is the most popular LAN technology in use today. It strikes a good balance between speed, price and ease of installation. There are two general topologies/configurations

- bus topology
- star topology.

The star topology is more reliable as a break in a link in the network affects only that link and not the entire segment as in bus topology. The bus topology on the other hand is more scalable. Switches can be used for connecting several bus segments in a star topology and this topology offers reliability and scalability.

Several media are in use today

- thick wire
- thin coaxial
- unshielded twisted pair (UTP)
- fiber optic

Thin coax and thick wire are both coaxial cables and are more expensive than UTP. UTP is a twisted cable and requires less care during installation. UTP comes in a variety of grades offering support for transmission rates up to 100Mbps.

Several alternatives were considered (Appendix A):

1. UTP network in the entire hostel : A switch is installed near the Hall Office and UTP wire drawn to each room using hubs which are cascaded from the switch.

2. UTP network to a wing or a floor in a wing : Connectivity is provided in an incremental fashion starting with a wing or a floor in a wing. UTP wire is drawn to a room using hubs which are cascaded. A switch has to be installed for providing connectivity to more than one wings.

3. UTP network to provide on-demand connectivity : A switch is installed near the Hall Office, a hub is installed in each wing for cascading and a hub is installed in each floor in a wing. Connectivity is provided on-demand by drawing UTP cable to the required room from the nearest hub. More hubs are installed on a floor if there are no free ports available on a floor and another hub is added in a wing for cascading if there are no free ports available for cascading in the wing.

- *Dial-up*

Dialup is the name given to the technology by which an end user connects to a network using the copper cables that provide voice connectivity. In order for an end user to be able to connect to the network using dialup, he must have a modem at his end connecting the telephone line and the computer. The computer center (the other end) must also have the necessary hardware and software to connect the user to the required system. Dialup works on the 4 KHz spectrum available to voice traffic and so is incapable of supporting speeds greater than 56 KHz (without compression).

- *Wireless*

This technology uses electromagnetic waves for communication between a mobile node and another possibly stationary node. The end user must have a wireless modem at his end capable of transmitting and receiving wireless signals. The other end must also have the necessary software and hardware to connect the user to the required system. This type of connectivity is restricted by the range of wireless transmission (˜ 1500m of aerial distance from the central site i.e. the other end).

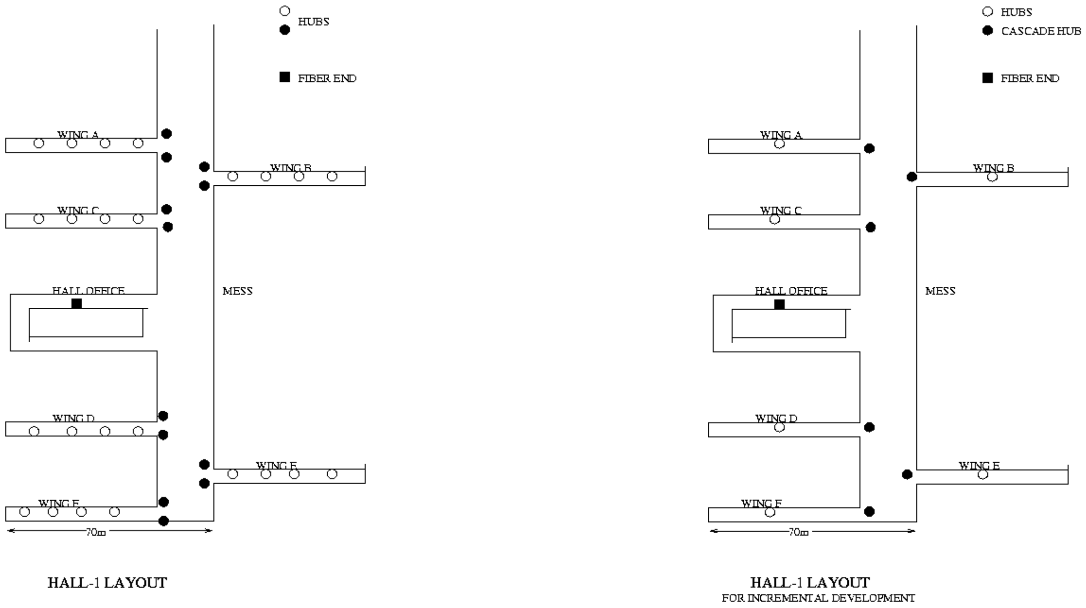

Figure 1: Layout and Dimensions of Hall 1

The bandwidth supported by this technology is in the range 1.2 Mbps to 2 Mbps. Another possibility is to have a remote hub connected wirelessly to the central site and to draw cables from this hub to the end users.

2.2.3 Comparison of Technologies

Refer to the attached Table No.2, Table No.3 and Appendix A for a brief comparison of the technologies considered for the backbone.

2.2.4 Recommendation

The solution that provides the best balance between cost and bandwidth is - providing a UTP network in each hall. On the basis of the analysis we can chalk out the following strategy for providing connectivity to the hostels:

- Keeping in mind the fact that the number of people desiring connectivity in the hostels will increase only gradually, we can reserve a floor in a wing of the hall for such people.

- As the demand for connectivity increases, another floor can be similarly wired. The costs involved are moderate enough to be borne by the wired students themselves (less than Rs 1500 each).

- When demand exceeds the level which can be satisfied by the above solutions, an entire wing or more will have to be wired, and this will involve investing in a switch.

- If a student desires to draw up a link to his room without having to shift to a new wing, there does exist a cheap solution, provided a hub exists with a free port.

- One-time wiring of the entire hostel, though not very expensive, is not immediately required as that level of demand will most likely not arise in the near future.

2.3 Providing Connectivity to the Faculty Residences

Presently, dial-up connectivity is available from the residences. At present, the modules in CC can support 48 connections simultaneously, but almost all of them remain idle.

The main requirements which need to be satisfied are:

- Moderate bandwidth at end points (in this case residences)
- On-demand connectivity to the residences or connectivity to all the residences
- Scalability
- Low cost of installation and maintenance

2.3.1 Overview of Technology Options Considered

■ *Ethernet on UTP/fiber backbone*

Ethernet is the most popular LAN technology in use today. It strikes a good balance between speed, price and ease of installation. There are two general topologies/configurations :

- bus topology
- star topology.

The star topology is more reliable as a break in a link in the network affects only that link and not the entire segment as in bus topology. The bus topology on the other hand is more scalable. Switches can be used for connecting several bus segments in a star topology and this topology offers reliability and scalability.

Several media are in use today :

- thick wire

- thin coax
- unshielded twisted pair (UTP)
- fiber optic

Thin coax and Thick wire are both coaxial cables and are more expensive than UTP. UTP is a twisted cable and requires less care during installation. UTP comes in a variety of grades offering support for transmission rates up to 100Mbps. Fiber optic segments offer very large bandwidths and can be up to 2km. But, fiber optic cables are very expensive Following alternatives were considered :

- Drawing the fiber backbone to a residential lane and laying down UTP network to the residences: The fiber is terminated in a media converter and connectivity is provided to the houses in the lane through hubs and UTP cable. (Appendix B).
- Drawing the fiber backbone to a residential block (6 lanes) and laying down UTP network to the residences: A switch is installed in the block and fiber is drawn to each lane where it is terminated in a media converter. Each house is connected to the network using hubs and UTP cable. (Appendix B)

Dial-up

Dialup is the name given to the technology by which an end user connects to a network using the copper cables that provide voice connectivity. In order for an end user to be able to connect to the network using dialup, he must have a modem at his end connecting the telephone line and the computer. The computer center (the other end) must also have the necessary hardware and software to connect the user to the required system. Dialup works on the 4 KHz spectrum available to voice traffic and so is incapable of supporting speeds greater than 56 KHz (without compression).

■ *Wireless*

This technology uses electromagnetic waves for communication between a mobile node and another possibly stationary node. The end user must have a wireless modem at his end capable of transmitting and receiving wireless signals. The other end must also have the necessary software and hardware to connect the user to the required system. This type of connectivity is restricted by the range of wireless transmission (˜ 1500m of aerial distance from the central site i.e. the other end). The bandwidth supported by this technology is in the range 1.2 Mbps to 2 Mbps. Another possibility is to have a remote hub connected wirelessly to the central site and to draw cables from this hub to the end users.

■ *DSL (Digital Subscriber Link) Technology*

Digital Subscriber Link (DSL) technology uses the copper lines of the Plain Old Telephony Service (POTS) to provide high data rates to end users. POTS signals operate in the lower 4 KHz region of the copper cable's bandwidth, limiting the total possible bandwidth available to normal modems to 56 Kbps only (without compression). DSL is able to operate at faster speeds by using much higher frequencies for communication – up to 1 MHz. The range of frequencies utilized by DSL is isolated from the frequencies used by POTS by means of an analog circuit called Splitter. DSL, thus doesn't affect POTS in anyway and so it is possible to make a telephone call and use DSL services at the same time. A DSL service requires two devices to function – one on either side of the line. A typical configuration is for the service provider to place a DSLAM (DSL Access Multiplexer) in the telephone company's central office and a DSL router, bridge, or modem at the user end.

The more popular types of DSL services include:

- SDSL (Symmetric Digital Subscriber Line) - This is capable of speeds up to 1.5 Mbps both up and downstream on a single copper pair. The effective range is up to 22,000 feet, with speed deteriorating at longer distances.

- ADSL (Asymmetric Digital Subscriber Line) - ADSL is really geared towards the standard end-user data application. Data rates differ up and downstream, with downstream rates between 1.5 Mbps and 9 Mbps, and upstream rates from 16 Kbps to 640 Kbps. ADSL has an effective distance of up to 18,000 feet on a single copper pair. Data rates deteriorate as the line gets longer.

- HDSL (High-data-rate Digital Subscriber Line) - HDSL is very similar to SDSL. It delivers a symmetric 1.544 Mbps over two copper pairs. HDSL has an effective range of 18,000 feet. HDSL is very useful, in that it provides a well-known data rate over two pair of copper at a distance great enough to serve just about everyone.

- VDSL (Very-high-rate Digital Subscriber Line) – VDSL is an asymmetric service with a short operating range - 1,000 to 4,500 feet. Although its range is short, VDSL is able to deliver from 13 Mbps up to 52 Mbps of data over a single copper pair. VDSL will only be used in certain niche applications, such as delivery of digital television services.

▮ *ISDN (Integrated Services Data Network)*

Integrated Services Digital Network (ISDN) is the name for digital telephone service that works over existing copper wiring. There are several types of ISDN services, but the type most appropriate for individual computer users is the ISDN Basic Rate Interface (BRI). BRI ISDN divides the telephone line into 3 digital channels : 2 "B" (Bearer) channels and one "D" (Delta) channel, each of which can be used simultaneously. The B channels are used to transmit data at rates of up to 64 Kbps. The D channel does the administrative work, such as setting up and tearing down the call and communicating with the telephone network. With 2 B channels, 2 calls can be made simultaneously. ISDN services can be provided only within a given distance of the serving telephone company equipment (typically 18000 feet) , and might even be affected by other anomalies near the wiring that might interfere with the transmission. ISDN requires adapters at both ends of the transmission - an adapter at the user's end and another at the access provider.

2.3.2 Comparison of Technologies

Refer to the attached Table No. 4 , Table No. 5 and Appendix B for a brief comparison of the technologies considered for the backbone.

2.3.3 Recommendation

We recommend that dial-up connectivity be provided to faculty residences. The bandwidth provided by dial-up is sufficient to meet the needs of almost all non-multimedia applications, and the cost of installation is also substantially less. We recommend that the number of dial-up connections be increased from the present number of 48 to about a 100 in the next 3-4 years. Besides, it would be a good idea to provide facilities to handle a few (say 10) wireless connections.

2.4 IP Address management for future growth of the network

In order to meet the increased demand for IP addresses which will inevitably result from the expansion of the campus network, we suggest IITK go in for a **private internet**.

IANA has reserved the following three blocks of the IP address space for private internets:

- 10.0.0.0 – 10.255.255.255 (10/8 prefix) – 1 class A address
- 172.16.0.0 – 172.31.255.255 (172.16/12 prefix) – 16 class B addresses
- 192.168.0.0 – 192.168.255.255 (192.168/16 prefix) – 256 class C addresses

Any enterprise that decides to use IP addresses out of this address space can do so without any coordination with IANA or any Internet registry. Addresses within the private address space will be unique within the local network, but can be replicated globally.

Private hosts will be able to communicate with all hosts within the network, but will not be visible outside the network. They can still access external services (eg FTP, E-Mail, rlogin) through mediating gateways.

Such a change involves a change of IP address, changes to the appropriate DNS entries, and changes to configuration files on other hosts that reference the host by IP address.

Because private addresses have no global meaning, routing information about private networks should not be propagated on inter-network links, and packets with private source or destination addresses should not be forwarded across such links. Routers in networks not using private address space are expected to be configured to reject (filter out) routing information about private networks.

Taking this option will not only help conserve the globally unique address space , but will also enable IITK to gain flexibility in network design by having more address space at its disposal than it could obtain from the global pool

On the flip side, this reduces the enterprise's ability to access the internet. As one commits to using a private address, one commits to renumbering part or all of the enterprise, should one decide to provide IP connectivity between that part or all of the enterprise and the internet. Cost of renumbering is linear in the number of hosts to be renumbered.

A possible strategy would be to first design the private part of the network first and use private address space for all the internal links. Next plan public subnets at the locations needed and design the external connectivity.

Later, if a group of one or more hosts requires to change status (from private to public or vice versa), this can be accomplished by renumbering only the hosts involved and changing the physical connectivity.

Changing the status of all hosts on a whole (sub)network can be done easily and without disruption for the enterprise network as a whole. As a result, it is advisable to group hosts whose connectivity needs might undergo similar changes in the future on their own subnets.

If an enterprise uses the private address space, or a mix of private and public address spaces, then DNS clients outside of the enterprise should not see addresses in the private address space used by the enterprise, since these addresses would be ambiguous. One way to ensure this is to run two authority servers for each DNS zone containing both publicly and privately addressed hosts. One server would be visible from the public address space and would contain only the subset of the enterprise's addresses which were reachable using public addresses. The other server would be reachable only from the private network and would contain the full set of data, including the private addresses and whatever public addresses are reachable the private network. In order to ensure consistency, both servers should be configured from the same data of which the publicly visible zone only contains a filtered version.

2.5 Conclusion

On the basis of the analysis we have performed in our study, we have come to the conclusion that the network configuration which would provide the maximum benefit for the minimum cost would be to have an optical fiber backbone for the campus (which is already in place) running Fast Ethernet, providing network connections to students' rooms by drawing UTP cable through hubs from the optical fiber terminated in the hostels, and providing faculty with incrementally increasing dial-up connectivity. In making our recommendations, we have considered not only the cost of the technology, but also the bandwidth it provides, the number of connections it can support, the cost of incrementally adding new connections, and past and future network usage patterns.

Though our calculations have been performed for very specific examples (e.g. we have studied the problem of providing connectivity to students' hostels by studying the Hall 1 layout only), yet the recommendations are general enough to be used in the entire IIT context, and even beyond it. The cost benefits involved in connecting students' rooms through UTP, for example, are so pronounced that even substantial variations in dimensions from those of the example studied will not affect the recommendation. In case of faculty residences, too, the recommendation that providing dial-up access is the best solution would not change for any campus with a computer usage pattern and layout similar to IIT's.

Also, in order to plan for future expansion of the IITK network, we recommend that IITK goes in for a private internet, and we also suggest a possible plan of action to make this transition.

Chapter 3

Security

The prime concern in any networked computing environment where data and authorization information flows over unsecured lines, as in the case of IITK's network, is to ensure that the confidentiality and integrity of data is maintained. It is also essential to be able to provide access to network resources and services in a manner such that only those users who are authorized to access the resources be provided permission to do so.

In order to ensure that the aforementioned concerns are met, it is necessary not only to apprise the network users of the privileges they are entitled to and the rules and regulations with which they must abide , but also to continuously monitor that no violation of the specified policies is taking place. It is also essential to ensure that the system configuration itself is secure, and does not provide opportunities to potential hackers to use glaring loopholes in the network to subvert the security of the network.

The approach we have followed in tackling this module of network design has been to first specify a sufficiently detailed security policy, and on the basis of this policy to monitor the usage behaviour of network users by means of the network intrusion-detection program we have designed. Also, we have installed mechanisms to monitor network and system configuration at regular intervals to check for exploitable loopholes, and have installed software which ensures confidentiality and integrity of network traffic. In each of these cases, we considered several possible options before deciding on the particular option we decided to adapt/implement to meet our requirements.

The entire exercise of securing our network has been done keeping the network of the Computer Science and Engineering department in mind. Extending this design to handle the institutewide network is a relatively simple matter.

3.1 Security Policy

The first task when defining a security policy is to actually define what a secure system is. In general, a secure system will control, through use of specific security features, access to information such that only properly authorized individuals, or processes operating on their behalf, will have access to read, write, create, or delete information. The following fundamental requirements are derived from this basic statement of objective, dealing with what needs to be provided to control access to information.

- The policy must specify explicitly the dos and donts that any user of the system must adhere to, and also what guidelines the administrators of the system should follow in order to maintain security.

- The security policy should be made well known.

- Individual subjects must be identified. Each access to information/service must be mediated based on who is accessing the information and what classes of information they are authorized to deal with. This identification and authorization information must be securely maintained by the computer system and be associated with every active element that performs some security-relevant action in the system.

- Audit information must be selectively kept and protected so that actions affecting security can be traced to the responsible party. A trusted system must be able to record the occurrences of security-relevant events in an audit log which must be kept safe from modification.

Based on these above criteria, we have made the following recommendations for the security policy of our network:

3.1.1 Network security

- Ensure only those services that are required from outside the CSE domain are allowed in through the router filters.
- If *ftp* is to be provided from outside, only anonymous ftp should be allowed.
- When using the *r* commands, care must be taken to allow only those which are needed, and even then only the most secure versions should be used, or the use of such commands should be replaced by the use of more secure alternatives (e.g. ssh).
- */etc/hosts.equiv* should be used with care.
- No user should be allowed to have a *.rhosts* file in his/her home directory.
- NFS should be used only if needed, and then with caution. Files should be exported with care.
- All system files should be owned by the root and should have their appropriate permissions.
- Only the latest versions of exploitable software, e.g. *sendmail*, should be used.

3.1.2 Password and account security

- A well defined password policy should be used, which specifies who can be provided access to the network and lays down the guidelines that any user of the network will have to follow.
- When providing access to a user, he/she should be explicitly made aware of the security policy the network uses and what action is to be taken in case of non-compliance with the policy, and his/her willingness to adhere to the security policy should be ascertained.

- In order to ensure strong passwords, the password changing program should verify at password change time that the new password is not easily crackable. Password aging should be used.

- Shadow passwords should be used.

- The system should regularly be checked for dormant accounts, and any account which has not been used for a reasonably long stretch of time, say 3 months, should be temporarily deactivated. Also, the password file should be checked at regular intervals for inconsistencies and for the presence of accounts with no passwords.

- Regular checks should be made for : successful and unsuccessful *su* attempts, repeated login failures, and LOGIN REFUSED messages.

- All persons using the *root* account should be careful not to compromise system security by unsafe practices.

- Regular backups of system areas should be made – both for recovering data in case somebody removes it, and for maintaining a record of what services/processes a user runs or uses.

3.1.3 File system security

- Ensure that any *.forward* files in the user areas do not execute any unauthorized command or program.

- All system files should have their correct permissions.

- All files run by root should be owned by root, should not be world or group writable, and should be located in a directory where every directory in the path is owned by root and is not group or world writable.

- Run software like Tiger/COPS/Tripwire to check the status of files in the system, and to detect the presence of loopholes in the system.

The means we have devised to enhance security are in keeping with the recommendations of this policy.

3.2 Security Audit of the Network Configuration

To regularly check for exploitable loopholes, we have installed freely available software which check the system for known bugs. These software are to be invoked at regular intervals in order to be able to continuously monitor the configuration.

We provide here brief descriptions of the functions performed by these software. More descriptions on how to use these software are provided in the user manual.

3.2.1 SATAN – Security Analysis Tool for Auditing Networks

SATAN helps in security analysis by gathering as much information about remote hosts and networks as possible by examining such network services as *finger*, *NFS*, *NIS*, *ftp* and *tftp*, *rexd*, and other services. The information gathered includes the presence of various network information services as well as potential security flaws – usually in the form of incorrectly setup or configured network services, well-known bugs in system or network utilities, or poor or ignorant policy decisions. It can then either report on this data or use a simple rule-based system to investigate any potential security problems. Users can then examine, query, and analyze the output with an HTML browser, such as Mosaic, Netscape, or Lynx. While the program is primarily geared towards analyzing the security implications of the results, a great deal of general network information can be gained when using the tool - network topology, network services running, types of hardware and software being used on the network, etc.

SATAN has a target acquisition program that uses *f*ping to determine whether or not a host or set of hosts in a subnet are alive. It then passes this target list to an engine that drives the data collection and the main feedback loop. Each host is examined to see if it has been seen before, and, if not, a list of tests/probes is run against it (the set of tests depends on the distance the host is from the initial target and what probe level has been set.) The tests emit a data record that has the hostname, the test run, and any results found from the probe; this data is saved in files for analysis. The user interface uses HTML to link the often vast amounts of data to more coherent and palatable results that the user can readily digest and understand.

SATAN can be used to perform heavy, normal, or light checks on the targeted hosts. The intensity of the check determines the amount of information gathered and the number of security flaws revealed about the targeted machines.

Once SATAN has been used to probe a target host or network, the data it has collected can be analyzed to check for security vulnerabilities. The probed hosts can be grouped together by approximate danger level, by the type of vulnerability, or by the vulnerability count. For example, when cd3 was subjected to an intensive scan, the vulnerability that was revealed was that it exports /3d5 to everyone.

SATAN can be used very easily by merely following the instructions mentioned in the online documentation.

SATAN can be used once a month, say, by the system administrator to perform a heavy check on the vulnerable machines of the network in order to highlight any incorrect network management policies or the possibility of external tampering with the system.

3.2.2 Tiger

TIGER, or the 'tiger' scripts, is a set of Bourne shell scripts, C programs and data files which are used to perform a security audit of UNIX systems. It is designed to hopefully be easy to use, easy to understand and easy to enhance.

TIGER has one primary goal: report ways **root** can be compromised. While checks are performed for other purposes, most of the checks are directed at this goal. The primary assumption made is that any uid other than 0 can be obtained and that *any* gid can be obtained by unauthorized persons.

In order to meet its primary goal, paths into **root** (*cron*, *inetd*, setuid executables, PATH etc) are all checked to see if anyone other than *root* can alter that path. Any such instances are flagged.

The amount of time required to execute TIGER varies on system size and the TIGER configuration settings.

TIGER performs the following tests :

- *cron* entries are checked
- mail aliases are checked
- NFS exports are checked
- *inetd* entries are checked
- PATH variables are checked
- *.rhosts* & *.netrc* files are checked
- Specific file & directory access permissions are checked
- File system scans locate unusual files
- Pathnames embedded in any files reported by most of the other checks are checked.

3.3 Intrusion Detection

Most existing systems have security flaws and are vulnerable to security violations ranging from attempted break-ins from outside the network to system penetrations and misuse by users inside the network. Intrusion detection attempts to detect such undesirable activities. Most systems implement some form of log keeping to store information about the system state and user activities which are security related. This audit data can be analyzed for detecting undesirable activities and system compromises.

Here we describe a program that we have developed for intrusion detection. The program employs two major approaches to intrusion detection: *rule based intrusion detection* and *a*nomaly detection.

Some general patterns of user behaviour have been identified as being indicative of an intention, if not an attempt, to misuse network resources. Rule based intrusion detection looks for patterns in the system logs which indicate suspicious and/or illegal user activity. These patterns of behaviour are apriori known to be suspicious. This approach allows one to detect intrusion, or possibilities of intrusion, if the observed behaviour matches this suspicious behaviour.

The anomaly detection approach is a more general approach and attempts at detecting intrusion, or possibilities of intrusion, even when the suspicious user behaviour pattern is not predefined. In this approach, initially user profiles are created for all users in the network. These profiles can be created by observation of individual user behaviour (by studying logs) over a period of time, or manually (by specifying the behaviour of a typical user). These profiles represent the usage patterns of "normal" system activity. These profiles are continually updated by the user's actual behaviour. If a user's behaviour in some session or over some period of time differs widely from the behaviour specified in the user's profile, it can reasonably be concluded that some other user has obtained, or is trying to obtain, access to the user's account, or that the user himself is trying to penetrate the system.

Another approach which we had studied, but which we did not incorporate into the program, was the *state transition based intrusion detection approach.* This approach models system compromises as a series of state transitions which take the system from an initial secure state to a target compromised state. State transition diagrams correspond to the actual states of the system and some key actions are defined which can prevent the attack from completing successfully.

3.3.1 Overall Architecture

In order to provide increased efficiency and ease of access to remote resources, we have envisaged a distributed design for our program. The program will run in one of the two possible modes: the central-mode and the subsidiary-mode.

There are some checks which can be run on only the central machine and some which can be run on other machines in the network. All machines in the subsidiary-mode send logs to the central machine at regular intervals, besides running the

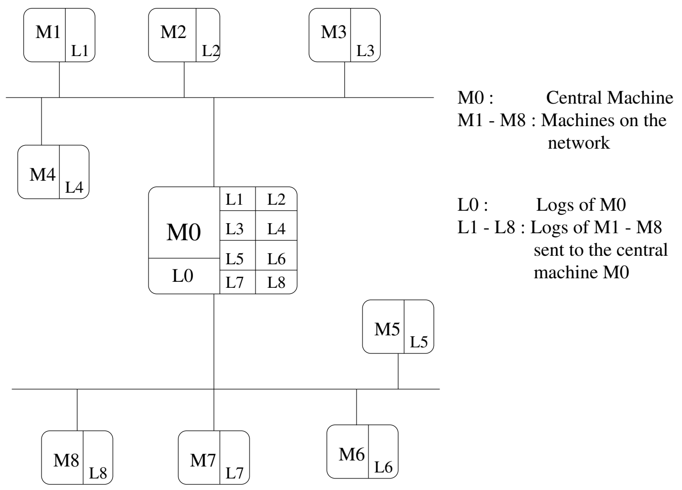

Figure 2: `The program architecture`

checks they are enabled to run. The central machine processes all the logs and performs the various checks. This includes the generation and updation of the user profiles from the log data, and detection of anomalies and intrusion events.

The logs maintained by all the machines in the network are sent to a central machine at regular intervals. The central machine thus has a repository of the latest logs of all the machines in the network. It uses these logs to provide information for the rule based intrusion detection and anomaly detection.

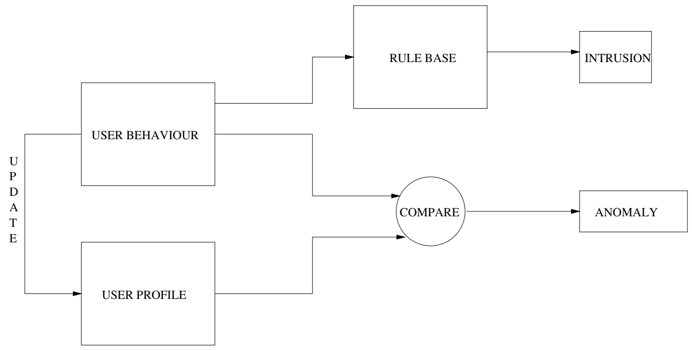

Figure 3: `The program design`

3.3.2 Rule Based Intrusion Detection

Each of the rules in this module to detect intrusion, or the possibility of intrusion, is implemented in the form of a check in a separate program. Each check-program has with it an associated specification file, which contains the values of the parameters which control the action and severity of the check. In case new rules are to be added, new {check-program, specification file} pairs have to be provided. The checks include possession of system password files in user areas, existence of .rhosts file in user home directories, change in permissions of files owned by root, authentication failures etc. If any of these events are encountered, some prespecified action is taken (e.g. a mail might be sent to the user warning him to desist from such activity in the future, a mail might be sent to the system administrator apprising him of the event and of the identity of the user concerned, the login rights of the user concerned might be withheld.

The checks which are implemented under this are:

- *Check for password files*

This check looks for password files in the user areas. The parameters for this check are list of user areas to check, the depth from the home directory to which the user area is to be checked, whether hidden files and directories are to be checked, the intensity of the check, the action to be taken if a match is found and which areas to check other than user areas. The checking of user areas is done by the central machine and the checking of other areas is done on the corresponding machine.

- *Check for .rhosts file*

This check looks for .rhosts file in the user home directories. The parameters for this check are list of user home directories and the action to be taken if .rhosts file is found.

- *Check for changes in permission of files owned by root*

Information about the permission of files owned by root is maintained and any changes in the permissions of such files are reported. The parameters for this check are the list of user areas and the directories which contain files owned by root. In case new files are added, the set of standard permissions is updated and are reported.

- *Authentication failures*

There are system logs which contain information about authentication failures. A lot of authentication failures in a short interval of time suggests system compromise. In case such an event occurs, the name of the user and the machine from which the request originated is reported.

- *Login from non-primary machine*

Some users have their private machines on which they normally work. In case there is a login on such a machine from a machine other than the primary machine, the list of users logged on the other machine at that time is reported. The parameters are the {user name, primary machine} pairs and the action to be taken in case of logins on primary machine from other machines.

- *Logging in as root*

In case there is a login as root on any machine in the network, the program first determines from which machine the login was made. A list of all users working on that machine, together with a list of the recent commands executed by these users on that machine are then mailed to the system administrator and are also added onto a file.

- *Running Sniffers*

Packet sniffers run by placing a network interface in the promiscuous mode. Our program detects a network interface running in the promiscuous mode (which might signal the possibility of a sniffer being run) and mails a list of users present on the system at that time, and also writes this information at some time. It is possible for the system administrator to decide to turn off this test for certain machines in case the network interface on those machines is deliberately being turned on in the promiscuous mode for administrative purposes.

3.3.3 Anomaly Detection

The program relies heavily on statistical techniques to determine the presence or absence of anomalies between the observed behaviour and the expected behaviour.

The parameters for which we construct the user profiles are :

- Login Frequency

  This counts the login frequency for a specified time interval on a given day of the week. This profile is especially helpful for detecting masqueraders who are likely to login to an unauthorized account during off-hours when the legitimate user is not expected to be there.

- Location Frequency

  This keeps a measure of the frequency of logins from different locations. The rationale behind this is that several class of users, for example faculty, might always log onto their accounts from a certain class of machines. If such a user logs on from a machine not in this class, it might signal a masquerader.

- Time between logins

  This measures the time since the last login and is useful for detecting a break-in on a "dead" account.

- Session Elapsed Time

  This is a measure of the elapsed time per session. A masquerader would typically login in short bursts, and this would likely create a deviation from the user standard profile.

- Command Execution Frequency

  This counts the number of times certain classes of programs are executed during a specified time period. A masquerader's command execution pattern might be expected to be significantly different from that of the user and so his behaviour would likely deviate from the user's standard profile.

3.4 Enhancing Network Security

In order to increase the level of security available in the network, we tested and studied various software alternatives.

We looked at **firewalls** as means to regulate access to the network from outside, and to the outside from within it. We studied **Kerberos** as a means of ensuring authentication and secure communication, and installed **ssh2**, the latest implementation of the IETF secure shell standard, to provide the same.

3.4.1 ssh

ssh is a protocol for secure remote login and other secure network services over an insecure network. It consists of three major components:

- Transport layer protocol provides server authentication, confidentiality, and integrity. It may optionally also provide compression. The transport layer will typically be run over a TCP/IP connection, but might also be used on top of any other reliable data stream.

- User authentication protocol authenticates the client- side user to the server. It runs over the transport layer protocol.

- Connection protocol multiplexes the encrypted tunnel into several logical channels. It runs over the user authentication protocol.

The client sends a service request once a secure transport layer connection has been established. A second service request is sent after user authentication is complete. This allows new protocols to be defined and coexist with the protocols listed above.

The connection protocol provides channels that can be used for a wide range of purposes. Standard methods are provided for setting up secure interactive shell sessions and for forwarding ("tunneling") arbitrary TCP/IP ports and X11 connections.

As **ssh** encrypts all network traffic and provides strong authentication, it meets our stated policy goals in these regards. We have installed and tested **ssh2** on machines in the network, and it does satisfy our goals of providing secure, unsniffable network communication. We recommend that **ssh** be used in place of the UNIX **r** commands.

3.4.2 Kerberos

Kerberos is a secure authorization and authentication method which uses a series of encrypted messages to prove to a verifier that a client is running on behalf of a particular user. It uses timestamps to reduce the number of messages need for basic authentication, a "ticket-granting" service to support subsequent authentication without re-entry of a principal's password, and mechanisms for cross-realm authentication.

All network traffic is encrypted, and so is not susceptible to traffic sniffing. Both the client and the service he is using need to authenticate themselves, so the danger of a user's data security being compromised by an unsafe application is minimized

On the downside, Kerberos security relies heavily on password security. Weak passwords compromise system security. Also, a safe password entry mechanism is needed. Kerberos needs to be integrated with other parts of the system. In order to provide complete security, all applications which the system supports need to be kerberised i.e. converted to a form which is compatible with the Kerberos security mechanisms.

Since Kerberos adequately meets our policy goal of providing safe authentication and identification together with guaranteeing confidentiality and integrity of network traffic, we attempted to install it on machines in our network. However, the requirement of kerberising all network applications made this a very time-consuming and laborious task, and we were unable to get the system up and running in totality.

Still, in order to ensure strong security in the network, we strongly recommend that we shift to Kerberos in the near future.

Kerberos can be downloaded from http://web.mit.edu/kerberos/www. It can be integrated with the system provided all the applications understand its format and security mechanisms. Kerberised versions of most of the applications can be downloaded from the net.

3.4.3 Firewalls

These look at the contents of data packets passing through them, and either drop or allow packets depending on

- source or destination IP address
- destination TCP/IP port

Using this method, one can control

- what destinations a user in the network can access (by filtering on destination IP address)
- what services a user can avail of (by filtering on destination TCP/IP port)
- who can access the network from outside (by filtering on source IP address)
- what services a particular user outside the network can access

However, in each of the above cases, user is synonymous with IP address.

Application level firewalls:

These work by scanning the data that flows through them. It uses special purpose code for each application that it is designed to handle. If a firewall recognizes some discrepancy in the format of a specific application, it can disallow traffic on that connection. Such firewalls also support real-time monitoring of the network traffic through them and help detect suspicious patterns.

Application level firewalls can help to:

- log and control all incoming and outgoing traffic
- specify individual machines or entire groups of machines outside the network that can gain access to individual machines or groups of machines inside the system, and vice versa
- specify the time of day when some network service is permitted and when it isn't
- authenticate users
- restrict the type of information that can be retrieved, even though the service that allows such information retrieval is allowed
- monitor the network in real-time to check for suspicious behaviour (eg too many file transfers/login-logouts to be considered normal)

Firewalls can help to control traffic into and out of a network. However, they are not useful in controlling traffic within the network. In the CC-CSE context, the CSE network is a switched segment of the CC network. Installing a firewall between the CSE network and the CC network will not be of much help, apart from allowing us to maintain continuous logs and to monitor suspicious behaviour.

At present, there is no firewall installed between the CSE network and the internet. However this does not pose much of a threat as the CSE network is not visible to the external connection. But if we envisage providing access to the CSE network from outside in the future, a firewall is a must.

Firewalls also do not provide security against an external machine which gains access to the network through a dial-up link, as such a link will bypass the firewall completely.

3.4.4 TCP Wrapper

This package monitors and filters incoming requests for the SYSTAT, FINGER, FTP, TELNET, RLOGIN, RSH, EXEC, TFTP, TALK, and other network services. It supports both 4.3BSD-style sockets and System V.4-style TLI.

The package provides tiny daemon wrapper programs that can be installed without any changes to existing software or to existing configuration files. The wrappers report the name of the client host and of the requested service. The wrappers do not exchange information with the client or server applications, and impose no overhead on the network.

Some optional features are

- access control to restrict what systems can connect to what network daemons
- client user name lookups
- additional protection against hosts that pretend to have someone else's host name
- additional protection against hosts that pretend to have someone else's host address.

The usual approach is to run one single daemon process that waits for all kinds of incoming network connections. Whenever a connection is established, this daemon (usually called inetd) runs the appropriate server program and goes back to sleep, waiting for other connections. The wrapper programs rely on a simple, but powerful mechanism. Instead of directly running the desired server program, the inetd is tricked into running a small wrapper program. The wrapper logs the client host name or address and performs some additional checks. If there are no problems, the wrapper executes the desired server program and goes away.

Two major advantages of this approach are that the wrappers are application-independent, so that the same program can protect many kinds of network services, and there is no interaction with client and server applications meaning that the wrappers are invisible from outside.

Another important property is that the wrapper programs are active only when the initial contact between client and server is established. Once a wrapper has done its work there is no overhead on the client-server conversation. The simple mechanism has a major drawback that wrappers are of little use with network daemons that service more than one client.

Chapter 4

Conclusion

In this project, we have tried to tackle the major issues which emerge when designing an intranet – network architecture and network security.

The primary problem of network architecture design is to arrive upon a solution which provides maximum possible transmission rates and transmission quality at minimum possible cost. A feasible network design can be arived at only after an extensive cost/benefit analysis. Using this guiding principle, we considered various possible technology alternatives and arrived at a sufficiently general solution which, though designed to serve the needs of IITK, can quite easily be extended to any network of roughly the size and complexity of the IITK network. However, it must be kept in mind that network technologies evolve rapidly, and future architecture designs must try to incorporate the latest technology available at those times.

In case of network security, we have tried to make an inherently insecure system more safe for the general user. Two major approaches have been combined in order to meet this goal. On the one hand, regular auditing of the network is envisaged, which will apprise the system administrator of any exploitable loopholes in the system software and configuration. Simultaneously, we have developed a software which will actively seek out intrusive user behaviour which signals an attempt to subvert system security. Besides, we have tried to make the nature of communication within the network more secure by installing ssh, a communication mechanism which uses strong encryption. Here, too, it must be kept in mind that the objective of system security is to always be one step ahead of possible intruders. As hacking techniques become more sophisticated, our software, too, will have to be refined in order for it to reliably perform its stated goal.

Bibliography

[1] D. E. Denning, *An Intrusion Detection Model*, IEEE Transactions on Software Engineering, Vol. SE-13, No. 2, February 1997

[2] K. Ilgun, R. A. Kemmerer and P. A. Porras, *State Transition Analysis: A Rule-Based Intrusion Detection Approach*, IEEE Transactions on Software Engineering, Vol. 21, No. 3, March 1995

[3] Documents on Network Equipment and Technology, ACES, IIT Kanpur