First Year Experiences of NSF REU Grant: Emerging Issues in Computer Networking

Stan Kurkovsky
Columbus State University
4225 University Avenue
Columbus, GA 31906
1-706-565-3520
Kurkovsky_Stan@colstate.edu

Bhagyavati
Columbus State University
4225 University Avenue
Columbus, GA 31906
1-706-565-3519
Bhagyavati@colstate.edu

ABSTRACT
In this paper, we describe our first year experiences of administering the NSF-supported Research Experiences for Undergraduates program award. Emerging issues in computer networking were the main focus of our program, which supported research projects of eight undergraduate students at Columbus State University in the summer of 2004. The main focus of this paper is on the lessons learned for successful future administration of the grant by shining the spotlight on select student projects completed within this program.

Categories and Subject Descriptors
C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design – Network communications.

C.2.4 [**Computer-Communication Networks**]: Distributed Systems – *Distributed applications.*

C.2.5 [**Computer-Communication Networks**]: Local and Wide-Area Networks – *Ethernet, Internet*.

K.3.2 [**Computers and Education**]: Computer and Information Science Education – Computer science education.

General Terms
Management, Design, Experimentation, Security, Human Factors.

Keywords
Research experiences for undergraduates, computer networking.

1. INTRODUCTION
In summer 2004, we successfully administered the first year of a two-year Research Experiences for Undergraduates (REU) program funded by the National Science Foundation (NSF). The key goals of this program at our teaching-centric university are to broaden educational experiences for undergraduate students in Computer Science and to motivate them to eventually pursue graduate studies. Based on the feedback we received from students, their faculty mentors and our peers, the program was successful.

Due to the limited resources available at our university in general and the Computer Science department in particular, we faced non-trivial challenges in advertising the program, recruiting high-quality students, developing interesting projects, motivating mentors, finding tutors and assistance, planning social activities, providing adequate housing and other facilities, and obtaining administrative assistance for the REU grant. We were able to provide projects for eight students from different areas of the country, three of whom were from our university and five from other universities. Seven mentors agreed to oversee eight REU projects that were, on one hand, challenging to the students and, on the other hand, possible to implement during the eight weeks of the short summer term.

The students were involved on a daily basis with a full agenda of attending classes, invited speaker sessions, tutorials on preliminary knowledge required for their networking-based projects, and social activities and trips. Each student met with their mentor at least twice a week, with two major assessments scheduled at mid-term and end of term. Biweekly stipends were available only after student progress reports were approved by the mentors, who were free to impose additional requirements such as status reports and other progress reports. Stipend payments were contingent on mentors' approval. A mandatory presentation and written report was required at the end of the eight-week term to obtain final stipends. In this paper, we elaborate on what worked well and what we seek to improve as we enter the second year of the program. These lessons learned are highlighted by illustrating some of the most noteworthy projects completed by the students participating in our program.

This paper is organized as follows. Section 2 describes our REU program in general; it highlights the major organizational challenges that we encountered and lists all projects completed by the students in our program. Section 3 focuses on four sample projects providing more detail about each of them. Section 4 concludes the paper with a summary. In Section 5, we acknowledge those team members whose collaboration and support was integral to the success of our work.

43rd ACM Southeast Conference, March 18-20, 2005, Kennesaw, GA, USA.

2. PROGRAM OVERVIEW

The "Research Experiences for Undergraduates – Emerging Issues in Computer Networking" is a research program administered by the Computer Science department at Columbus State University (CSU) and funded by the National Science Foundation. The objective of this program is to provide an opportunity for eight undergraduates to conduct research in emerging issues in computer networking.

The program targets the students who have completed their sophomore or junior year of study, have prior knowledge of computer networking, and have significant knowledge of at least one high level programming language.

The main goal of our program is to provide the opportunity for undergraduate students to gain useful and hands-on experience in the exciting area of emerging research issues in computer networking. By participating and completing research projects in this area, students will obtain the experience and skills in general research methods, specific skills and techniques in computer networking research, as well as presentation and writing skills. Current and future research results gained from these projects have proven valuable in furthering research into computer networking and its applications. Participating faculty are interested in this subject and teach courses in computer networks, wireless communications, and network security. The experience from this program is carried directly into the course work in the form of student projects. The recruiting efforts of this program are designed to attract and increase the participation of underrepresented groups in higher education. The last, but not least, this program is designed to encourage undergraduate students to continue their graduate study and possibly dedicate a career to computer science research.

Though this is our first year running the program, we have achieved promising results. First of all, we recruited eight students from CSU and other institutions. We started the student recruiting process in February 2004. We set up a comprehensive program web site at http://csc.colstate.edu/reu and posted the program announcement and application form on the program web site. We also used email and regular mail to distribute the program announcements to the department chairs of targeted schools. We followed up by personally contacting the faculty in these schools and asked them to recommend high-quality students to participate in our program. We visited a number of schools, including Fort Valley University and Georgia Southern University, to recruit students. With these efforts, in less than two months, we received twenty-nine applications, with twenty-one of them qualified for the REU program. Finally, we selected eight students, a female student and five underrepresented minorities among them. Three students were from Columbus State University and the rest came from other universities including Fort Valley State University (an HBCU), Auburn University, Jackson State University, Bard College, and California State University Long Beach. The matching of students and mentors was based on the interest and suitability of both the student and the mentor.

Our program consisted of two parts, the independent study session and the summer program. The independent study session took place from April 12 to May 21, 2004. After we notified the students of the selection results by April 12, we requested them to contact their mentors to start the independent study session, which is one of the noteworthy features of the program. The purpose of the independent study session is to prepare students for the summer project. During an approximately one month period, the students were assigned to read some background materials and discuss their future problems with their mentors. Due to the distance reason, it was not feasible for some students to meet with their mentors once a week. But email communication worked well. We received very positive feedback from students who told us that this study session helped them "to warm up" before coming to CSU.

The summer program started on May 24 and ended on July 31, 2004. During the summer program we undertook the following steps to provide a high-quality program. In the first week of the program, we provided a five-day crash course in networking. Although all students participating in our program were required to have prior knowledge of computer networking, taking this short course served as a good mechanism to refresh their knowledge and skills. In the mornings, the students attended a lecture detailing essential theoretical concepts, and, in the afternoons, the students gained hands-on experience at the departmental computer labs.

We chose a variety of research projects for students to work on. Research topics covered the fields of switch scheduling, wireless applications, network security, grid computing, and web services. The specific topics for our research projects included the following:

- Automated vulnerability scanning;
- Honeypots and honeynets: watching the hackers;
- Consumer-oriented wireless applications;
- Intelligent locally distributed web-server systems;
- ISP anomaly detection for home users;
- Infrastructure to support campus-wide wireless networking;
- Using mobile devices and image analysis to provide real-time parking information; and
- Efficient scheduling for multi-stage switches.

Each student was provided with a networked PC and an office shared by two students. We also arranged for two graduate students to assist the students participating in the REU program in programming. Our program participants were registered as transient students at CSU to receive one credit hour for participating in the program. The students also had full access to the university library.

To expose the students to common directions in research methods, specific skills and techniques in the area of computer networking, and presentation and writing skills, we provided students with a series of lectures, which included the following:

- A lecture on general research methods in computer science, which taught students the basics of working with research literature, finding relevant sources of information, brainstorming, etc.; and
- A series of five lectures on graph theory.

We also offered students a number of research seminars covering a diverse range of cutting-edge research topics in computer networking:

- Reliability in optical WDM networking;
- Wireless grids: a look at the future computing portals;
- Introduction to computer security; and
- TCP in wireless networks.

We broadened the impact of our program by specifically targeting small, primarily undergraduate institutions with limited research resources in the southeast area and by encouraging underrepresented groups and minorities to participate in our program. One female undergraduate student and five underrepresented minority students were among the eight students participating in our REU program in the summer of 2004. We also tried to develop connections between the students and the local community by organizing several field trips and tours. The first field trip was to TSYS, one of the largest credit card transaction processing companies. The students toured their campus and working environment. They were introduced to the credit card transaction processing flow, which was a first-time experience for most of the students. The second field trip was to Fort Benning, the largest infantry base in the US. The students watched the airborne and parachute jumping demonstration, visited the infantry museum and three virtual reality and computer simulation labs. Most of our students were very excited to see and test the virtual reality solider training facility in these simulation labs. The tour to Callaway Gardens provided a good opportunity for students and mentors to have a social gathering.

In order to provide a high-quality research program, we designed a complete assessment plan for the program to address both the research project and financial aspects. Each student was guided by a mentor, who met with the student at least twice a week for the duration of the entire summer program. Each mentor maintained a detailed portfolio consisting of the weekly meeting records, two-week progress evaluation forms, student reports, etc. In particular, mentors were required to monitor the successful progress of the student and confirm it by signing a bi-weekly progress evaluation form. Each payment of the student stipends depended on their successful work in the corresponding time period, which was certified when mentors signed their evaluation forms. Students were also required to prepare a midterm and a final report followed by a presentation of their research results.

Logistics was also a very important aspect of a program such as REU. We worked very hard to provide students with the best financial, facility, room and board arrangements. Before the students arrived on CSU campus, we had to work very closely with the corresponding business entities at our campus. Since each student had a different situation, we had to work out a payment schedule for each individual student. The students' stipends and allowances were distributed every two weeks. We prepared a welcome package for each student, which included their first payments, a welcome page with their office assignments, a tentative program schedule, biweekly evaluation forms, and a map of the campus. We distributed these packages at the welcome meeting in the beginning of the program (May 24, 2004). We gave the students a tour of our campus, showing them the facilities of the department, library, cafeteria and the athletic facilities. Each student received a student ID card for accessing the library and gymnasium facilities. We also arranged a welcome party and a farewell party for the students and invited mentors and other faculty and students to attend.

Although mentor-student communication was excellent during the course of the project, communication between the students and the mentors after the completion of the summer REU project posed a significant challenge. The grant proposal committed the recipients to dissemination at regional and national professional meetings. Students and mentors were both encouraged to share their REU experiences with peers and professional audiences. However, we faced great difficulties when we attempted to contact REU students after summer 2004; post-project communication has been an ongoing problem. Students were unable to sustain the level of interest in the REU project after the completion of the short summer term. We plan to weave dissemination expectations into our recruitment efforts for summer 2005 REU applicants.

In the next section we discuss four sample projects completed by students in the frameworks of our summer 2004 REU program.

3. STUDENT PROJECTS

Each of the eight students participating in our program successfully completed a research project during the summer REU program on CSU campus. In this section, we briefly describe some of these projects and the results achieved by the students.

3.1 Automated vulnerability scanning

The goal of this project was to assess a range of vulnerabilities posed by the computers connected to the Internet through Internet Service Providers (ISPs) and to design a comprehensive solution to address the corresponding problems. For example, it is possible that a computer connected to the Internet through the ISP's servers may have a vulnerability that might enable a malicious person to damage the ISP's machines, or damage other networks through the ISP's servers. ISPs might be held liable for such damages to other networks because they failed to address vulnerabilities in their own networks. While the ISP can formulate and implement security policies and fortify its system against vulnerabilities, the weakest link in its network is the user's machine, which connects to its servers. One way to strengthen the ISP's network is to perform routine vulnerability scans of all the computers connected to and through its network, and act upon the results of the scans.

Although several companies offer vulnerability scan services to consumers, some of them free of charge, most consumers are either not aware of the damage caused by vulnerabilities, or not aware of the services, or are not willing to pay for these services. It is in the interest of the ISPs to provide vulnerability scan services at the outset, so that users are spared the pain of paying for it and taking the initiative to use it, both in monetary and technical terms. The vulnerability scan service can run on the ISP's servers and monitor the users connected to the servers at all times.

This project was conducted in three stages. In the first stage, a thorough study and analysis of vulnerabilities was undertaken. While there is a large number of existing vulnerability scanners, none of them has proven to be able to fully extend the meanings

of scanning accuracy, usability, and dependability [2]. The open-source Nessus scanner [3] was identified as the best among vulnerability scanners based on these criteria (Figure 1). However, it lacks an automated scan feature, which can be implemented on the client's system. It is also in the ISP's interest to convert the highly technical error codes and messages of Nessus to simple and user-friendly language. For example, if the user's machine has outdated antiviral definitions, the project goal was to provide meaningful messages so that the user could download the latest definitions from a specified website.

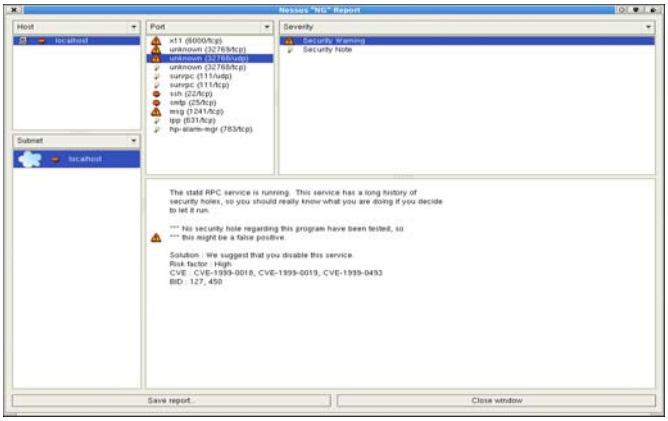


Figure 1. Screenshot of Nessus automated client extension (project result).

This stage yielded a survey that served as the theoretical framework for the analysis of vulnerability scan services; it also provided a basis for the second stage, which involved the design and development of a system that addresses the weaknesses in currently available scan services. In particular, it addressed the problem of lack of transparency in existing vulnerability scan services, and the necessity of user initiative to start the monitoring service. Because the ISP proactively scans client machines requesting services, the user is unaware of the underlying issues. The lack of transparent scans imposes additional burdens on the casual user. Automating the process and running it unobtrusively in the background enhances security while causing less frustration to the user.

In the third and final stage, a prototype for the design obtained in the second stage was developed. The top 10 vulnerabilities were downloaded on a regular basis from the SANS website and checked for on the client machines requesting services from the ISP's servers. Toward the end of the project, a client-server based vulnerability scan service with clients available for both Linux and Windows systems was prototyped. The end product was envisioned as an automated scan and notification system that ISPs could use to monitor individual connections. Although such a system was successfully prototyped, its implementation was not fully completed due to time constraints.

Nessus is a client-server model-based tool for scanning computers to pinpoint their weaknesses or vulnerabilities. It requires a server to be set up on Linux before setting up clients on any machine. Nessus is scalable and used for remote audits; it is a free, open-source tool that is feature-rich and checks for audits. Fast and reliable, Nessus provides a modular architecture for users.

3.2 Honeypots and honeynets

The objective of this project was to study and survey a number of existing honeypots. According to [4], a honeypot "is an information system resource whose value lies in unauthorized or illicit use of that resource." Honeypots can be used to do everything from detecting encrypted attacks in IPv6 networks to capturing the latest in on-line credit card fraud. It is this flexibility that gives honeypots their true power and also makes them challenging to define and understand. There are many ways honeypots could be deployed and used.

In general, honeypots can be divided into two general categories, low-interaction and high-interaction honeypots. Low-interaction honeypots have limited interaction, and they normally work by emulating services and operating systems. The advantage of a low-interaction honeypot is its simplicity. These honeypots tend to be easier to deploy and maintain with the minimal risk. They involve installing software and selecting the operating systems and services to emulate. The main disadvantages with low interaction honeypots is that they log only limited information and are designed to capture known activity. The emulated services are limited in their capabilities and may allow a skilled hacker to know that s/he is on a honeypot and is being watched. Examples of low-interaction honeypots include Specter, Honeyd and BOF (BackOfficer Friendly).

High-interaction honeypots usually use complex solutions as they involve real operating systems and applications. They do not employ any emulated services, therefore inviting the attacker to a real operating system and computer hardware. The advantages of such a solution are two-fold. First, extensive amounts of information can be captured. By giving attackers real systems to interact with, the full extent of their behavior can be observed. The second advantage of high-interaction honeypots is that they make no assumptions on how an attacker will behave. Instead, they provide an open environment that captures all activity. This allows high-interaction solutions to learn behavior that may not be expected. Examples of high-interaction honeypots include Symantec Decoy Server and Honeynets.

Honeypots can help prevent automated attacks, such as worms or auto-rooters, which are based on tools that randomly scan entire networks looking for vulnerable systems. If such systems are found, these automated tools take over the system. "Sticky honeypots" monitor unused IP space and when probed, these honeypots interact with and slow the attacker down.

Honeypots can help detect a failure or breakdown and help prevent it. Detection of an attack allows a quick reaction to it, stopping or mitigating the damage. Technologies such as IDS sensors and systems logs haven proven ineffective for intrusion detection because they generate far too much data with a large percentage of false positives, they are unable to detect new attacks or work in encrypted environments. Honeypots excel at detection, addressing many of these problems.

There is usually little information on who the attacker is, how they got in, or how much damage they have done. In these situations detailed information on the attacker's activity is critical. Often the honeypots are designed to capture anything thrown at them, including tools or tactics never seen before.

The advantages of honeypots are that they are designed to capture any malicious activity; they can collect in-depth information that any other technologies cannot match; they require minimal resources; they are conceptually very simple. The disadvantages of honeypots are that they can only track and capture activity that directly interacts with them; they cannot capture attacks against other systems, unless the attacker or threat interacts with the honeypots; just like other computer systems, honeypots can also be taken over by an attacker and used to harm other systems.

3.3 Campus wireless infrastructure

This project explored the use of mobile computing devices in supplementing university course content and deploying a supporting wireless infrastructure. Completion of this project required a survey of the available types of mobile devices; a study of the infrastructure and techniques needed to support connecting mobile devices to the Internet and local networks; creation of the content suitable for display on the various types of mobile devices; and a study of the security options for mobile devices in an academic environment.

Anywhere, anyplace computing is becoming commonplace in the world of business supported by a wide availability of mobile computing devices used in conjunction with wireless networking systems. Unfortunately, little has been done to fully use this computing capability in the world of education [1]. The infrastructure needed to support wireless technology is not in place in many colleges and universities; many of them are unaware of the requirements needed to support wireless technology.

Six types of existing mobile computing devices can be identified: web-enabled phones, wireless handhelds, two-way pagers, voice portals, wireless appliances, and wireless PCs.

The number of access points required to create a campus wide wireless network was calculated by determining the area in which wireless local area network (WLAN) coverage is needed; defining the size and the number of users in each area; estimating the total bandwidth needed for each area and the corresponding number of wireless access points. It was estimated that introducing this infrastructure on CSU campus will require 40 indoor access points, 30 repeaters, and 4 outdoor access points.

There are three types of wireless Internet networks that mobile devices can connect to: Wide Area Networks (WAN), a licensed public wireless network used by Web cell phones and private radio frequency (RF) digital modems in handhelds, which uses cellular towers to transmit information and has the most powerful signal with a ten-mile range; Local Area Networks (LAN) operating on the unlicensed spectrum and most often used by computers, handhelds, and some cell phones, but not by pagers, which uses wireless base stations or access points as transmitters; and Personal Area Networks (PAN or ad hoc networks) that can be created with two or more wireless devices within each other's range.

In order to thoroughly secure a wireless LAN, an authentication, authorization, and accounting (AAA) framework should be used. The AAA approach to secure mobility uses information from 802.1x client authentication to map users to their native Virtual LAN (VLAN), regardless of where they are connected in the WLAN. This design enables IT organizations to locate and follow users as they move, and applies security context unique to each user.

Students will constitute the bulk of users accessing the envisioned wireless infrastructure. Most likely, they will be trying to access information that is necessary for them to contact the instructor, find out when homework/tests are due, and what date is the assignment or test on. Due to the fact that mobile devices are not currently able to provide the speed and processing power of a desktop or laptop computers the size of the data transmitted to mobile devices must pages be kept to a minimum. Furthermore, as the users may have mobile devices with different capabilities, the campus infrastructure will need to provide services accessible by a wide range of devices.

Finally, a mobile computing application was developed using Visual Studio .NET 2003 that demonstrates the integration of course content with mobile computing devices with varying capabilities.

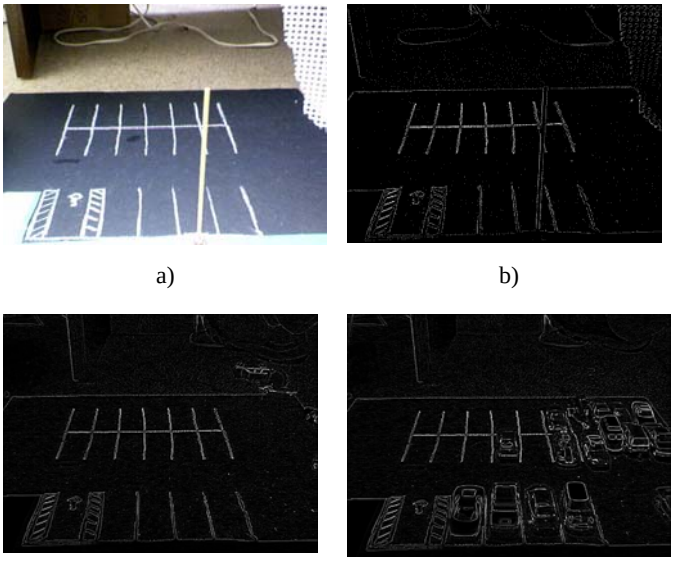
3.4 Using mobile devices and image analysis to provide real-time parking information

This project explored the use of mobile computing devices and video imaging analysis to provide real-time parking information to faculty, staff, students, and visitors on campus. It consisted of investigating image capturing techniques, developing techniques for analyzing images to determine vacancy of parking spaces, developing XML Web services that will display the parking availability information on mobile devices, and investigating the security and privacy issues inherent in implementing the project.

One of the common problems with large universities, hospitals, and other institutions is parking. Parking problems are caused by poor design and parking space inadequacies. The objective of this project was to develop a technique for analyzing parking spaces for vacancies and provide this information to drivers before arriving to their destination. This information could allow drivers to spend less time driving around frustrated looking for a vacant parking space and more time doing the task at hand.

The challenges encountered in this project were how to obtain the data, process it, work with multiple parking lots, and provide useful information to the drivers. A scale model of a parking lot was built to test the system design and provide more control compared with testing the system with the images acquired from a real parking lot. A web camera was used to obtain still images of the parking lot model. These images were then processed using an assorted amount of image processing filters applied over the still images to reduce noise and derive the essential information (Figure 2). The system was designed from the beginning to work with multiple cameras placed around campus and targeted at the same or various parking lots. Finally, ASP.NET Mobile Controls were used to transform the information according to the specification of the mobile device accessing this information by the driver such as mobile phone, PDA, web page, or other mobile devices.

It is possible that some day parking vacancy information systems will be as common as the GPS navigation systems found in modern cars. Today's GPS navigation systems could easily be integrated with this system. Therefore, it provides drivers with not only directions to place where they want to go, but also with directions to vacant parking spaces.



a) b)

c) d)

Figure 2. Image analysis for parking information.
a) Vacant parking lot;
b) Processed image of the same lot;
c) Processed image of a vacant parking lot used as a reference;
d) Processed image of a parking lot with cars.

4. SUMMARY

One of the best gauges for measuring the success of a program such as our summer REU program is the quality of the student projects completed within its framework. The four projects briefly described here provide a sense of the quality and depth of research that students undertook while working this summer on our campus. Two of our 2004 REU student projects, one of which is described here, have recently won first prizes at the student research competition held at the 2004 Midsoutheast ACM Conference in Gatlinburg, TN, from November 10 to 12, 2004.

We believe that we have completed a successful program this summer. We have gained useful experience, which we are planning to use and broaden in the second year of our program in the summer 2005. Although we achieved our goals for this year, we feel that we could further improve the process of student recruitment. Also, we would like to organize more social activities and encourage more mentors, faculty and students to attend these activities.

Another area where we seek improvement during the second-year administration of this grant is in the area of post-project communication with REU students. As PI and co-PI of the grant, the authors have decided to incorporate active measures to increase student and mentor participation in professional meetings to disseminate and share the work done under the REU grant. For example, we plan to make dissemination a part of the requirement for the successful completion of REU projects next year. We will also aim recruiting efforts at students who are enthusiastic about presenting their knowledge to regional and national audiences. This will help us in soliciting REU alumni feedback after the completion of the program.

With the successful completion of the first year administration of the NSF REU grant at our university, we have experienced the challenges and issues involved in student recruitment, progress project completion and dissemination of results. We have resolved some of the challenges and look forward to applying the lessons learned during the second year of the grant. Our main focus as we move forward will be to strengthen the quality and motivation of students during recruitment and to encourage post-project sharing and presentation of results by both mentors and students at professional conferences and gatherings.

5. ACKNOWLEDGMENTS

Our REU program is financially supported by NSF award CNS-0354144. We would like to thank all students who participated in our 2004 REU program – Brian Bolton, James Glass, John Hudson, Todd Johnson, Mayauna McCullough, Ayodejo Olatunji, Erik Seijo, and Evan Tran. We would like to thank Dr. Saad Biaz for delivering the lectures in the computer networking crash course, Dr. Edward Bosworth for his series of lectures on graph theory, and REU mentors: Dr. Edward Bosworth, Dr. Ronald Linton, Dr. Wayne Summers, Mr. Christopher Whitehead, and Dr. Mei Yang.

6. REFERENCES

[1] Eaton, A. Technology-supported pedagogy in business, technical, and professional communication. Business Communication Quarterly, 66(3), pp. 113-117, 2003.

[2] Forristal, J. and Shipley, G. (2001, January 8). Vulnerability Assessment Scanners. http://www.nwc.com/1201/1201f1b1.html

[3] Nessus. http://www.nessus.org

[4] Spitzner , L. Honeypots: Tracking Hackers. Addison-Wesley Professional, 2002.