RADICL: A Reconfigurable Attack-Defend Instructional Computing Laboratory[1]

Submitted to the

International Conference on Security and Management
Las Vegas, NV, USA, June 20-23, 2005

by

Sergio Caltagirone, Paul Ortman, Sean Melton, David Manz, Kyle King, and Paul Oman*
Department of Computer Science
University of Idaho
Moscow, Idaho, 83844-1010
(*contact: oman@cs.uidaho.edu; 208-885-6899)

Abstract: In order to provide a safe and separate system on which to train, test, and teach computer and network security, a Reconfigurable Attack-Defend Instructional Computing Laboratory (RADICL) was designed and implemented at the University of Idaho Computer Science Department. The reconfigurable network system is relatively inexpensive to build even through it supports several operating systems in a myriad of network topographies. University faculty and students can use the laboratory to understand attack scripts and other *malware* while devising defensive strategies and tools. Product developers could use the laboratory to test the defensive aspects of their products, including computer applications, operating systems, and netware. And system administrators could test various security concerns and be trained in the proper application of security principles without jeopardizing day-to-day operations on existing systems. This paper describes the rationale, design and implementation of the vision for RADICL. An in-depth description of the design and implementation of the lab is included.

Keywords: Computer and Network Security, Attack/Defend Exercises, Reconfigurable Networks, Security Training.

I. Introduction

Each year the threats and dangers associated with using the Internet grow as the number of hackers, viruses, and other malicious entities increase [1]. In most academic computer security classes, cyber defense is the central topic and students of computer security acquire much of the requisite knowledge with respect to systems defense principles [2, 3, 4]. However, some academicians have identified a specific need for instruction on the tools and techniques of systems attack [5, 6]. While the University of Idaho is recognized as one of the Top 15 “wired” schools in the nation, everyday computer networks cannot be used for state-of-the-art pedagogy in attack-defend scenarios. Hands-on experience with these topics requires a laboratory computer network that is both isolated (so it won’t impact or contaminate day-to-day computing facilities) and reconfigurable (to allow diverse operating systems, network topologies, and defensive strategies). With a capacity building grant from the National Science Foundation (NSF), we designed and implemented RADICL as a student project.

The driving motive behind the RADICL facility is the need to provide hands-on computer and network security experience to undergraduate and graduate students studying various aspects of Information Assurance (IA). It is clear from the near-exponential increase in cyber intrusions and attacks documented by CERT over the last 10 years, that bookwork and classroom exercises are not sufficient to prepare IA professionals to adequately defend our computer networks and real-time control systems. By experiencing actual attacks – and implementing actual defenses – our IA students will gain the knowledge and insights that will enable them to design and implement more secure and survivable systems.

[1] The authors gratefully acknowledge funding support from NSF grants #DUE-0114016, Cyber Service Training and Education, and #DUE-0416757, Capacity Building for the UI SFS Program.

Development of laboratory in which students can study attack scripts and other forms of malicious software (aka malware) gives them increased exposure to the vulnerabilities and problems that are present in actual networks. Industry also has the need to Beta test new products and technologies on systems that are not currently in use for day-to-day operations. Academia and industry also face the difficulty of teaching and training people who may cause system failure by making unneeded or incorrect system changes.

The broader impact of providing hands-on attack-defend laboratory experiences to IA students is twofold. First, both large complex IT systems and smaller real-time control systems used in our digital society would be better managed by experienced IA graduates in the event of accidental or deliberate damage. This enables more stable and dependable infrastructures. Second, by better educating our upcoming IA professionals we move closer to a new generation of secure networks and computer systems. By giving them hands-on experience with today's IA tools we can increase the chance that they will develop tomorrow's security technologies.

The next section outlines the goals and objectives of the RADICL facility, while Section 3 provides project organization details and Section 4 provides implementation details. Sections 5 and 6 describe the current use of RADICL and conclude the paper with a summary of the benefits of implementing a reconfigurable network testbed.

II. Goals and Objectives

The main intent behind RADICL was to provide a hands-on attack-defend laboratory environment that allowed us to update and upgrade our computer and network security course materials so that our IA students were better prepared to enter the workforce. IA course at the UI include, but are not limited to: Writing Secure Programs, Applied Security Concepts, Computer Forensics, Intrusion Detection Systems, Real-time Control System Security, Network Data Communications, Network Security, Computer Security Concepts, Survivable Systems and Networks, Fault-Tolerant Systems, and Secure Web Programming.

Faculty members teaching the above courses, and other special topics courses, needed an isolated lab in which to conduct their classroom exercises. The facility had to have fast, flexible network reconfigurations to easily support a multitude of attack-defend exercises all within the same semester. For example, one instructor might need to run intrusion detection exercise and, in the same afternoon, the facility might need to be reconfigured for another instructor's fault-tolerant network topologies. All of the above classes needed a facility in which a variety of state-of-the-art attack-defend scenarios could be hosted. For example:

- Virus/worm propagation and detection
- DOS and DDOS detection, resistance, and recovery
- Password cracking detection and defense
- Intrusion resistance, detection, and recovery
- Exercises in vulnerability assessment and mitigation
- Applications of firewalls, proxy servers, perimeter defenses, and bastion hosts
- Exercises in system log file analysis
- Network cut-point and fail-over strategies
- Man-in-the-middle attacks and defenses
- URL and MAC address spoofs and filters

Thus, through reconfigurability the RADICL facility needed to support all existing IA faculty and classes with easy, fast, and flexible network configurations, including but not limited to the following:

- Uniform or multi-variant Linux domains
- Uniform or multi-variant Windows domains
- Linux domain to Windows domain interconnections

- Switched network in a Star topology
- Hub network in a Bus topology
- Hybrid Switch/Hub network in a Tree topology with up to 8 leaves per network hub
- A single Linux or Windows Server with up to 15 clients
- Between 2 and 4 Linux/Windows Servers with between 1 to 14 clients

Given the above need, objectives and goals, the following major design considerations for RADICL were identified:

1. The facility would host both Linux and Microsoft operating systems, and multiple variants of those systems.
2. The operating system for any given machine must be easily changed by rebooting the machine.
3. The network topologies (aka configurations) were to be dynamic. The network would be easily reconfigured for between 0 to 16 Linux variants connected with 0 to 16 MS Windows variants.
4. At least two of the machines would be equipped as servers with high-speed microprocessors, one gigabyte of memory, and dual Network Interface Cards (NIC's). The other machines will be equipped as application-level client machines with medium speed microprocessors, 1/2 gigabyte of memory, and single NIC's.
5. Network equipment would consists of a firewall/router, a 24 port Layer 2 switch supporting multiple bridge groups, two eight port hubs, and cabling to conveniently (re)connect the application machines to the network router, switch, and hubs.
6. A centralized short-rack hosting the networking equipment was necessary to permit fast, flexible network reconfigurations.
7. Cost of the hardware could not exceed $50,000.
8. There was no additional cost for software because public domain Linux operating system variants would be used and the university had site licenses for Microsoft products. Public domain attack and defend tools would be used in laboratory exercises.

III. Project Organization

The University of Idaho is recognized as a NSF Scholarship for Service (SFS, aka CyberCorps) school and a National Security Agency (NSA) Center of Academic Excellence in Information Assurance Education (CAE/IAE). Although the UI Department of Computer Science is the driving force behind these recognitions, other departments, most notably the Department of Electrical and Computer Engineering and the Department of Accounting, contribute much to the IA expertise at the university. The team that developed RADICL consisted of senior and graduate students from Computer Science, Computer Engineering and Accounting, all enrolled in a directed-study class in the Fall semester of 2004. The initial goal of the class was to design, build, and test a reconfigurable network testbed for attack-defend classroom exercises. The planning and deployment of the reconfigurable network testbed consisted of four subtasks: (1) room layout, (2) workstation/server needs analysis and acquisition, (3) netware needs analysis and acquisition, and (4) software needs analysis and acquisition. The students were divided into four teams (A through D) and each team was given the same task at the same time in a competitive nature. The teams were charged with solving the problem at hand and presenting their solution to the entire class. After all the teams had presented their solutions the class voted on the best solution or approach to take, and then all teams iteratively addressed the next issue or problem. It was in this friendly competitive manner that the RADICL lab was specified, designed, and implemented by the students, with minimal direction by the course instructor.

IV. RADICL Implementation

Physical Layout. The physical configuration of RADICL played a large role in the usability of the lab and its ability to meet the requirements of diverse exercises. Each team was charged with designing a physical room layout given the dimensions of the room and the expected furniture and equipment racks. The "winning" design consisted of single-machine portable desks that could be organized in a variety of configurations, but normally would be two rows of eight desks facing each other. Each desk would have a monitor, keyboard, mouse, and a Keyboard-Video-Mouse (KVM) device connection that user station to the rack-mounted workstation. The machines were placed in racks in the corner of the room with KVM Ethernet cables running down the central corridor between the two rows of desks connecting to each KVM user station. Figure 1 shows the physical layout of RADICL.

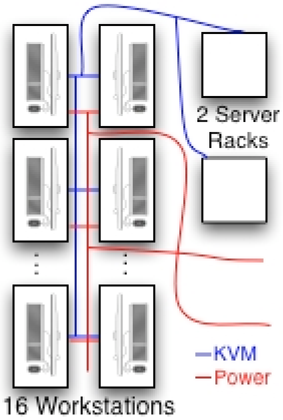

Figure 1. RADICL Room Layout

The desk arrangement can be broken up such that individuals, groups of two, groups of four, and groups of eight could be separated for purposes of classroom exercises. A pull-down screen and projector were placed on one end of the room to provide for presentational capabilities.

Workstation/Server Hardware. The lab contains sixteen 2GHZ/256MB/ 80GB workstations, four with dual network interface cards (NICs). There is also one 3GHZ Xeon server with 4x200GB drives on RAID 5 to act as the image and backup server — this is connected to an un-interruptible power supply (UPS). These components are divided onto two racks that are situated under an ventilation duct for cooling. Figure 2 depicts the server, workstation, and KVM arrangement.

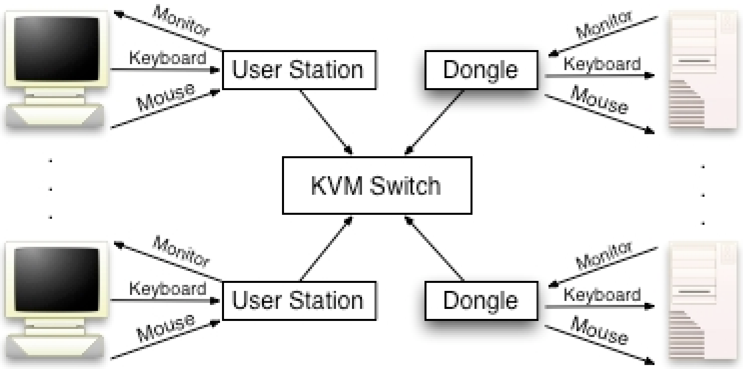

Figure 2. KVM Switch Configuration

To provide reconfigurability that is generally not available in most labs, an enterprise KVM solution from Raritan was implemented. Using KVM allows 16 workstations to access up to 128 workstations — all over CAT5 cable. This allows each user to control as many of the lab machines as necessary from one location. This means that two people could be playing "wargames" from several different machines without moving. This technology greatly increases the types experiments and exercises that can be performed.

In any situation, especially one that facilitates experiments and classroom exercises, a method of backup and recovery must be created. In RADICL there are two methods for file backup and recovery. The first is the RAID image server that allows for drive failures without loss of data. The second method is the inclusion of a DVD+R drive on the server that allows the laboratory to create 4.7GB DVD disks for backup purposes.

Netware. The lab network is not only a communication structure for each of the other components, but also satisfies the rapid transition requirements by enabling instructors and lab administrators to quickly change the lab's topology. To accomplish this, the lab uses a Cisco Catalyst 3550 Layer 3 switch. This switch provides us 48 10/100Mb ports for future expandability and a 10/1000Mb port directly connected to the backplane for a faster connection to the image server. The 3550 allows us the ability to partition the network into virtual local area networks (VLAN), configure spanning ports (to sniff traffic from multiple subnets and VLANs), and set access control lists (ACL).

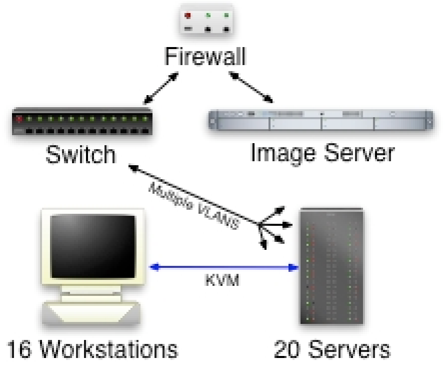

Figure 3. KVM Switch Configuration

Aside from the 3550 switch, the lab also has a Cisco PIX 501 firewall to further adapt the lab's network topology as well as giving instructors the ability to firewall without a dedicated machine. The PIX firewall also allows the ability to set up virtual private networks (VPN) between different VLANs on the network for additional configurations.

Software. One of the primary technologies used in the lab to make it as reconfigurable as possible is the use of disk imaging. This allows a snapshot of a hard disk to be made and reloaded on demand on any number of machines. It was decided to write management software for the laboratory based on a GNU/Linux platform due to its interoperable utilities. The end product was a series of shell scripts on the back end, and a dynamic web administration front end. These programs control the creation and deployment of software images for the client machines as well as network configurations for the room. Faculty and students can easily reconfigure the room for custom experiments. There are basically two main actions performed by users of the system, creating images and network configurations, and deploying these configurations to the room.

All machines in the lab must be initialized prior to their first use, and this process can be redone if all data on the machine is corrupt or there happens to be some hardware failure. To accomplish the initialization of a lab machine, the user first starts the computer from a bootable CD. The user then initializes the network interface and uses the FTP to connect to the image server and download our "master" script. Running this master script with the single parameter of default starts a process that connects to the server, downloads all necessary tools and then partitions the hard drive. It also pulls down images for the Windows boot and utility partitions and applies them to disk partitions. Finally, the boot configuration file is created and the master boot record is initialized. After this process is complete (~5min) the machine is completely initialized.

The process of creating an image is one of installing an OS (or using an existing OS image) and customizing it to fit the needs of the experiment. After installation and configuration is complete, the task of preparing the image for deployment takes several steps. The user will first reboot the machine into the utility partition where the rest of the process will take place. From within the utility partition the newly configured partition is modified to make it partition and machine agnostic. At the same time, a configuration script is written to apply machine and partition specific information back on to the image when it is deployed in the future. The final step in image preparation is to create a small snippet of Grand Unified Boot Loader (GRUB) configuration for the configured image. When this configuration is done, the master script is once more invoked, this time with the push parameter. This process then extracts the contents of the partition that was configured, bundles up the special configuration the user just created and pushes to the image server.

The process of deploying an image for a session of experimentation is anticipated to be the most common user interaction, and thus it has received the most automation and interface emphasis. From a web page interface on the imaging server, a user is shown the current configuration of every machine in the lab. This information includes which OS installs are on each partition, who installed them there, why, and if they are considered to be a dangerous install. Dangerous installs are installs where there was a chance of worm propagation, virus distribution, or any such nefarious process that might pose a danger to other machines or experiments. After viewing the configuration, the user can chose to change the configuration of any of the machine, including changing multiple machines configurations.

Our performance testing has shown that a typical OS install of a modern OS, without many additional utilities compresses roughly to a 500-1000MB file stored on the image server. The process of pulling this image to the client takes approximately four minutes. While this process is IO bound, specifically it will go as fast the lab hard drive can write data to disk, the required time is well within the limits needed to effectively reconfigure the lab between classes separated by a one hour block.

V. Using RADICL

Although RADICL is only three months old, we have already scheduled classroom exercises into it. The scripted reconfigurability and KVM architecture provides a mixture of network topologies, operating systems, and client/server configurations supporting several attack-defend scenarios, including:

- A single server behind a firewall being attacked by 15 clients
- Two servers (with or without a firewall between them), each with between 1 and 14 attacking or defending clients
- LAN sniffing of switched or hubbed networks, with and without firewalls
- Packet injection of switched or hubbed networks, with and without firewalls
- Attack-defend intrusion resistance, detection, and recovery
- DOS and D-DOS attack resistance, detection, and recovery
- Virus and worm propagation, detection, and eradication
- On-line password cracking detection and defense
- System log file analysis during active attacks
- Configuring and attacking firewalls, proxy servers, perimeter defenses, and bastion hosts
- Man-in-the-middle packet observation, hijacking and injection, with and without firewalls
- Attacks on weak cryptographically protected transmissions

VI. Summary and Conclusions

Computer security issues are coming to the forefront of a computer science education. Even so, actual hands on experience in computer security remains sparse in academia. RADICL was created to remove this gap between theory and practice. RADICL was designed to allow students to do attack/defend exercises. These exercises require a variety of operating systems, virtual LANs, and the ability control many computers from one workstation. The lab also allows students to have full administrative access to their machines. Its design makes RADICL one of the most useful labs in the country. The lab also has to remain highly reconfigurable. An exercise might require every operating system in the lab to be changed or re-initialized to some preset configuration. Toward that goal, a sophisticated but simple imaging scheme was designed in which users can upload an image of any partition to a central image server where an administrator approves it for safekeeping. These images can be downloaded and deployed on any computer in the local network in a matter of minutes. In this way, the lab will be used as dedicated classroom for a small number of network and computer security courses while remaining open for research groups and attack/defend exercises.

Pressures in industry for reliability and security can make deploying new technologies difficult and disruptive. New employees require significant training and exposure to proprietary systems in order to repair, modify or upgrade them. A reconfigurable network testbed allows companies to train new individuals to their specific systems and methods of integration so that when changes to the live system are made, they can be made with confidence and with a minimum of disruption to the system. Product developers can also use a testbed to simulate and test possible system settings changes and layouts. The testbed also allows for the testing of new technologies that may be applied to improve security. The impact of various technologies like encryption and Virtual Private Networks (VPN) can be studied to see how they will impact a live system and how they can best be deployed.

A reconfigurable network testbed, like RADICL, is both affordable and extremely valuable in both university and industry settings. The potential gain for students, teachers, industry, and manufacturers make it a necessary tool in the war against cyber terrorism.

References

1. CERT, Computer Emergency Response Team Coordination Center, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, www.cert.org/advisories.

2. M. Bishop, "Teaching computer security," *Proceedings of the 9th IFIP International Symposium on Computer Security*, 1993, pp. 43–52.

3. NSTISSI-4011, "National training standard for information systems security professionals," June 20, 1994.

4. S. F. Barnett, "Computer security training and education: A needs analysis," *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, Oakland, CA, 1996, pp. 26–29.

5. J. M. Hill, C. A. Carver Jr., J. W. Humphries, and U. W. Pooch, "Using an isolated network laboratory to teach advanced networks and security," in 32nd SIGCSE Technical Symposium on Computer Science Education. Charlotte, North Carolina, United States: ACM Press, 2001, pp. 36–40.

6. P. Mullins, J. Wolfe, M. Fry, E. Wynters, W. Calhoun, R. Montante, and W. Oblitey, "Panel on integrating security concepts into existing computer courses," SIGCSE Bulletin, vol. 34, no. 1, 2002, pp. 365–366.