CS 491/591 Spring 2010 Lab Policies

There will be one machine dedicated as the *dedicated machine* for carrying out your projects, I'll give you an account on it. The domain name is:

shasta.cs.unm.edu

No data on this machine or any of the virtual machines running on it is private, I reserve the right to view and copy your files on this machine and the virtual machines for forensic or other purposes. Keepin mind, also, that your classmates may have elevated privileges on these virtual machines, so assume nothing is private (if they steal your source code out of your directory that's their fault, not yours, though, so it's okay to keep your source code on these machines). All of the projects can be carried out in simulated environments without any need to do anything malicious, illegal, or against university policy. Thus, anything that is malicious, illegal, or against university policy falls outside of the scope of this class and you will be held personally responsible for it.

The following rules apply to all assignments and everything we learn in the class, including, but not limited to, the projects.

Rule #1: All forged packets, *e.g.*, for ARP injection or TCP fragmentation, are to be forged only on the tuntap interfaces of the dedicated machine. A tuntap is a virtual network that connects the virtual machines, so your forged packets should never actually go out on the physical Ethernet cable. We will install safeguards to protect the outside network if you should accidentally route forged packets or other malicious traffic to eth0, but you may still be held responsible if these safeguards fail and you were doing something other than what the assignment says to do, so do not forge packets on the eth0 interface of the dedicated machine and do not forge packets in ways not specified in the assignment without prior permission. Note that the tuntap interface is eth0 within the virtual machines, so for some labs your code should forge packets to interface eth0 for the virtual machines but you should never run that code directly on shasta, only in the virtual machines. You are not to forge packets on any university-owned network or computer other than the dedicated machine. You also take legal and ethical responsibility for any packets you forge off campus, *e.g.*, on your private network at home.

Rule #2: All malicious traffic and scanning traffic (remote exploits, nmap, *etc.*) is to be sent only from a virtual machine to another virtual machine connected by the tuntap, all of this contained within the dedicated machine. Malicious or scanning traffic other than prescribed by the assignment is not permitted without prior permission. The policies about safeguards, university-owned computers and networks, and off-campus networks from Rule #1 also apply to this rule.

Rule #3: You are to develop and execute source code for privilege escalation (*e.g.*, gaining root with a race condition or installing a kernel rootkit) only on virtual machines specified for this purpose that will be hosted on the dedicated machine. Do not exploit vulnerabilities on the other virtual machines

designated for other purposes. You may use your own private systems and virtual machines that you have root access for whatever purpose you wish but then you take personal responsibility for attack containment. Do not develop or execute code, scripts, or commands to exploit vulnerabilities on university-owned computers other than the dedicated machine or any computers anywhere that you are not permitted to have administrator access on. This is against the law.

Rule #4: Do not store or execute malicious code samples (viruses, worms, *etc.*) on university-owned computers other than the dedicated machine. I will place malicious code samples that you need on thededicated machine. Do not transfer malicious code samples to or from this machine. Do not execute malicious code samples in virtual machines without my explicit permission. If you analyze and/or execute malicious code on your own personal machine you take personal responsibility for its containment, so I strongly recommend that you yot do this unless you know what you are doing.

Rule #5: Do not use any cryptanalysis, cracking, sniffing or other inference algorithm, tool, or technique to violate the privacy of others, the security of any system, or to violate any laws. This includes dictionary attacks, keystroke logging, password cracking, and anything we learn that involves inferring private data, passwords, or other information you are not granted access to.

Rule #6: You are personally responsible for any actions you take that (1) diverge from what is necessary to complete the assignment as specified or (2) violate the laws, policies, and ethical standards of the university and applicable jurisdictions, or violate the security of any system or the privacy, integrity, or availability of any person's data. This applies to all actions, not just those covered by rules 1-5.