Global Information Assurance Certification Paper

CyberLaw 101: A primer on US laws related to honeypot deployments

GSEC Gold Certification

Author: Jerome Radcliffe, jay.radcliffe@gmail.com

Adviser: Jim Purcell

Accepted: February 1, 2007

Outline

1. Introduction

A Honeypot is defined as an Internet-attached server that acts as a decoy, luring in potential hackers in order to study their activities and monitor how they are able to break into a system[1]. These devices have created a confusing interaction of legal and cyber issues, as discussions of such devices are typically accompanied by a legal disclaimer, yet, these legal issues are not typically discussed due to time constraints or lack of experience in legal matters. At a recent SANS conference a lawyer in the group lectured for five minutes on the need to consult a legal team before deploying or using honeypots when the topic came up, and pointed out the many tricky legal issues surrounding such devices. Yet, the same lawyer was not able to specify the legal issues nor was he able to make suggestions on how to handle such issues. This legal gray area presents two interesting issues. First, honeypots are one of the more esoteric issues that a corporate counsel would have to address and, it is very possible, a corporate legal team might not have the required knowledge to answer questions on honeypot issues. Second, not all IT professionals have has access to corporate counsel and hiring a lawyer for advice on this specific issue is often not cost effective, again, due to the esoteric nature of the issue.

The goal of this paper is to provide a general primer on two legal issues related to honeypots, privacy right and entrapment, and to provide practical advice regarding prudent actions to take for legal due diligence. By no means should this paper serve as a replacement for legal advice from informed legal counsel. Instead this paper is meant to serve as a bridge between the legal and technical world on the honeypot issue.

Honeypot Background

The world of honeypots can be very complicated, as they are not designed to prevent attacks, nor are they designed to detect attacks which separate them from traditional Intrusions Detection Systems (IDS), Intrusion Prevention Systems (IPS) and firewalls (references)[2]. These devices are exceptionally useful tools for the computer security professional as they allow for a full end to end analysis of an attack, including all of the details that surround the attack. Through analysis of the log files it is easy to identify the pre-attack and post-attack actions that were taken by the intruder. This would include new rootkits or different payloads that are installed after the box has been compromised. In many cases this is how new exploits are found in the field.

Honeypots are often used in the process of catching and

prosecuting cyber criminals. This criminal aspect of honeypots has lead to a scrutiny of legal issues related honeypots use in the IT and field. The complexity of the law prevents many people from the using honeypots. One of the legal complications is that the law applies differently depending on who is "acting" (in this case who has actually established and monitored the honeypot)[3]. To understand the legal debate three distinct groups need to be noted; first those acting on behalf of the government, second those non-governmentally funded groups, and third individuals. Those acting from a governmental perspective include law enforcement, governmental agencies, or any federal or state funded group, such as the local police setting up a website targeted at online child predators. The second group includes groups that are not governmentally funded, such as corporations or private research groups. An example of that second category would be Joe Smith, senior system administrator for Acme Oil, setting up a honeypot on their company network. The third group would apply to individual, such as a person who has set up a honeypot on their personal internet connection (DSL/Cable,etc). In each of these groups the laws applies differently.

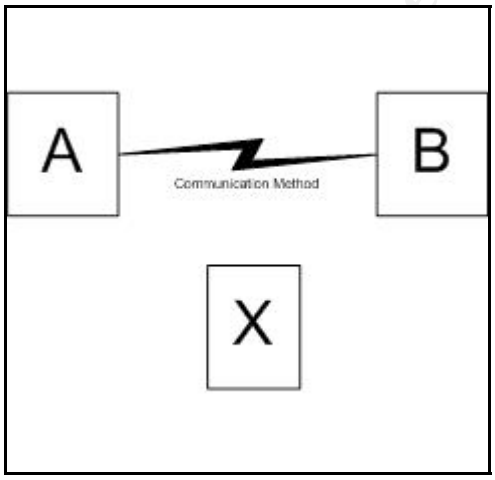
There, however, are some disadvantages to honeypots. Primarily, they are very time consuming to setup and maintain. When a honeypot is deployed properly, it attracts intruders quickly and plentifully.

These intruders and their actions can produce a mountain of data to be analyzed and dissected. One can easily spend days digesting the data that is produced by a single honeypot. As they are designed to be broken into honeypots are often victims of Denial of Service (DoS) attacks as well as other attacks that will require hands on administration. This can often lead to insufficient resources to maintain the production IT activities, or possibly distract staff from performing their primary tasks.

Privacy and The EPCA

The first legal topic that comes up in regards to honeypots is privacy. Privacy has a controversial history in US law. Compared to other issues in law though, privacy is a new concept. Privacy law itself did not become codified until the 1960's[4]. The ground breaking Supreme Court case Katz vs. U.S., 389 U.S. 347, 350 (1967), started the judicial systems wave of rulings on privacy rights. With honeypots, the privacy concern comes from the fact that a honeypot is recording all the activity that is occurring on that device. Upon first glace one might argue that this is very similar to the action of wire tapping, but this comparison would not be legally accurate. The US Federal law distinctly separates spoken communications from electronic communications[5]. The set of laws that defines these types

of communications is commonly referred to as the Electronic Communications Privacy Act or ECPA[6]. The EPCA outlines all of the details related to the interception and recording of electronic communications. To better understand these concepts it is necessary to define the agents involved in basic communications. First, there has to be a minimum of two agents involved in the communications (agent A and B). In this case it does not matter if A or B initiated the communication. This communication is conducted over a given method, which can take many different forms, and the details of that method can have an effect on the interpretation of the law. There can also be a third party that is not directly involved in the communication, referred to as agent X. The agent X may or may not be known to either A or B. The diagram below….



Applying this diagram to a typical single device honeypot where

"A" is the honeypot and "B" is the accessing user. The controlling party of "A" can be categorized into one of two groups under the EPCA: "A person acting under the color of the law"[7], and "A person not acting under the color of the law"[8]. In either case, because the honeypot is owned and operated by "A", they are considered a "party" involved in the communication. According to the EPCA it is legal to intercept and monitor such communications, with an exception. If those communications are intercepted for the purpose of committing a crime (for example, you were harvesting credit card numbers to perform fraudulent charges) then that action would not be legal.

Any parties that are not directly involved in the communication would be represented in the diagram as "X". There are two circumstances under the EPCA when it would be legal for an "X" party to intercept communications between "A" and "B". The first legal circumstance is referred to as the "Provider Exception" of the ECPA; in this circumstance "X" can intercept communications when "X" owns the infrastructure; such as an ISP. In such a circumstance, the company would have a legal right to intercept the communications for the purposes of protecting the ISP's service and to allow the companies providing service(s) to the public to monitor their systems for potential failures or quality issues. The second circumstance by which one could gain legal status to intercept communications from a

non-party agent is through consent.

An additional factor in the legal ability to intercept a communication is the method by which the communications takes place. Wireless communication has become a popular method of networking, both in work and home environments. As these forms of wireless communications can be recorded and monitored with RF monitoring devices, they are legal to intercept and monitor[9]. The exception to this is if you the use of any type of encryption with a wireless communications (such as WEP). The use of "scrambled or encrypted radio communication"[10] changes the communication to become *not* "readily accessible to the general public"[11] and, thus, not legal to intercept. Monitoring a neighbor's wireless activity, if they are transmitting unencrypted, is perfectly legal under the ECPA. If the neighbor's wireless AP is using WEP monitoring that network is a violation of the EPCA. The action of capturing the transmission is the interception, so the use of tools to determine the WEP key via packet analysis is also, arguably, in violation of the EPCA. These, however, are very gray areas of the law and there are no established court rulings with which to clarify these murky areas. Authors, such as Orin Kerr, have argued that there should be no reasonable expectation of privacy of the actual encrypted data (or crypt-text)[12] due to the fact that it is just plaintext characters. The knowledge

that any encryption is “breakable” via bruteforce attack is enough of a vulnerability to establish that someone might be able to translate the encrypted data into the original message thereby makes it less private. The EPCA however specifically states that encrypted or scrambled transmissions are protected from interception.

Consent

Consent is defined as “a voluntary agreement to another's proposition”[13]. This becomes an exceptionally difficult topic due to number of different services that run on any given computer system. On most systems there are interactive services like telnet, SSH, and terminal services. It is fairy simple to create viewable consent messages pre and post login to these interactive services. On a UNIX based device the consent message is in /etc/banner for pre-login and /etc/motd for post-login could cover the legal requirements of consent. This consent banner tells the user that, by logging into the computer system, they are consenting to the recording and monitoring of all communications sent and received by that user. Entering their password to gain system access is an acceptance of the terms given in the banner message. To reinforce the agreement further, the post-login method of alerting the user to monitoring and recording message is given before the user is allowed to interact

with the system. On a windows based system, one can perform the same pre-login banner message. It is a little more complicated, as it requires manual changes to the registry, but there are directions located at Microsoft's website[14] and other widely available websites with similar information.

This issue of consent becomes more complicated with services that are not as straightforward as authenticated services. With a web server, for example, it is very simple to add a link to the bottom of every page published that point the user to the consent warning. The problem is that there is no assurance that the user saw that link or that consent page. This problem has no simple solution. There is a discussion about this topic here (http://www.webdeveloper.com/forum/showthread.php?t=12057) that suggests several technology based solutions to the link based consent issue. An even more complicated situation comes from services that run "behind the scenes" where direct user contact should never occur. An example would be SMTP services where when a user sends an e-mail out and there is no interaction with the actual user in the process of the delivering that mail. There is no method of delivering a consent warning in this type of situation. Another condition where a consent warning cannot be delivered is to unauthorized backdoor services, such as Sub-Seven or Back Orifice 2000. In the world of

computer security there are always ways of getting on a system that bypass the authentication method and thereby avoid the consent banner. In this case there is usually a persuasive argument that, in the process of bypassing the standard method of entry, the user knowingly breaking the law and forfeits their privacy protection. In many cases the act of implementing banners on all banner-able services is enough to legally carry over to the banner-less services.

Just adding the consent banners to servers is not enough. A time might come where evidence of the installation of consent banners is needed. There are several steps that should be taken to address this issue. First, good documentation of build procedures for servers is needed to provide a clear baseline of what a server looked like when it was built. A legal team can use that documentation as proof that consent banners were in place. The documentation can be very simple; an example would be using comments in kickstart or hardening scripts used for automated build procedures. Adding the comment "Installing Consent Warning in pre and post login files for SSH" right before creating the banners or copy them from another server is often all that is needed. Be sure that the build procedures are being followed. If the documentation for the build procedure is in a binder on a shelf, then it is going to become out of date very quickly. Policies that are not followed are useless, and even

damaging, in a legal environment. A second banner verification method is a full system backup, this is read-only and dated and can also be used in a legal environment to verify that the consent warning were in place.

The next question is what exactly one should put in the consent message. This is where the consultation of legal counsel is needed. As an example here is a consent banner from a Department of Defense website:

> "This is a Department of Defense computer system. This computer system, including all related equipment, networks and network devices (specifically including Internet access), are provided only for authorized U.S. Government use. DoD computer systems may be monitored for all lawful purposes, including to ensure that their use is authorized, for management of the system, to facilitate protection against unauthorized access, and to verify security procedures, survivability, and operational security. Monitoring includes active attacks by authorized DoD entities to test or verify the security of this system. During monitoring, information may be examined, recorded, copied, and used for authorized purposes. All information, including personal information, placed on or sent over this system may be monitored. Use of this DoD computer system, authorized or unauthorized, constitutes consent to monitoring of this system. Unauthorized use may subject you to criminal prosecution. Evidence of unauthorized use collected during monitoring may be used for administrative, criminal, or other adverse action. Use of this system constitutes consent to monitoring for these purposes."

http://cap.public.msg.wpafb.af.mil/ncbanner.htm

This sample banner is a good place to start. While it is very long, it is likely that the DoD legal team has covered all the bases and can provide a good place to start building a consent message.

Entrapment

Entrapment is defined as “the act of law enforcement officers or government agents inducing or encouraging a person to commit a crime when the potential criminal expresses a desire not to go ahead”[15]. This legal gray area of entrapment is often misunderstood and can be a confusing area of law. The first area of confusion relates to the fact that entrapment is *only* a legal defense and not something that you can sue someone for. This means that the concept of entrapment is used by the accused (AKA the defendant) to avoid conviction. The US legal system’s presumption, or X, sides with the prosecution, meaning that the court assumes that the accused was not entrapped into the action of which they are accused. This is an important fact as presumption is difficult to overcome and, most of the time; the accused has the benefit of presumption in all other arenas. Further, entrapment is a very narrowly defined circumstance. To prove entrapment as a defense one needs to prove that the accused would not have taken the criminal action without the influence of the agent

acting under the color of law. Here are two examples:

Scenario One: Jake (a fictional skilled computer administrator) goes to a 2600 meeting and there he meets Judy and her friends. Judy and her friends are talking about hacking into Acme Inc.'s web server. Judy asks Jake if he wants to take a try at hacking into the server, and Jake politely declines. Judy pesters Jake, calling into question his "skillz" and general manliness. Jake accepts the offer and proceeds to hack into Acme's web server and starts monitoring its traffic. Little does Jake know that Judy is working as an undercover agent.

Scenario Two: Matt (also a fictional computer administrator) goes to a computer security convention and meets Tom at one of the courses. Tom mentions to Matt the he is going to hack into Acme's web server and he needs Matt's help. Matt says he doesn't do that sort of thing. After quite a bit of prodding and insulting, Matt still insists that he uses his skills only for good. Tom then takes out a pistol and threatens Matt's life unless he helps Tom. Matt concedes and hacks into Acme's web server. Tom is also an undercover agent.

Both Jake and Matt are charged with various computer crimes and go to trial. The question at hand is if either Jake or Matt have a strong entrapment defense. Jake's entrapment argument is going to be

very weak as it did not take much to get Jake to change his mind and commit a criminal act. If Jake chose not to commit a criminal action there would have been minimal consequences, other then a bruised ego. Matt, on the other hand, felt that his life was in danger and that the only alternative he had was to commit the criminal action to escape the threat. What makes entrapment such a difficult defense is it is impossible to determine if the accused is pre-disposed to commit the crime in question.

The entrapment issue arises with honeypots because the intention of a honeypot is to attract intruders. This is similar to law enforcement using undercover agents masquerading as drug dealers to attract drug users. There are some significant differences though. There is no recruitment of people to interact with the honeypot nor is there any interaction with the users that are interacting with the honeypot. As there are no interactions with people, it makes the defense of entrapment exceptionally difficult to establish. What is important, in terms of entrapment, is any communications regarding the existence of the honeypot. If a message was posted on several internet message boards, as an anonymous user, exposing your honeypot and encouraging others to hack into it the action, i.e. making the honeypot known, increases the ability of the accused to use an entrapment defense in the event of a criminal case. There is a direct

relationship between the amount of communication with the accused, and their ability to use an entrapment defense. Ideally one would want to limit the amount of communications about the honeypot.

The purpose of a honeypot might not be for catching criminals. Honeypots are often used to learn from in a research setting but the setting does not change how the entrapment issue is approached. There are many different scenarios in which a research-intended honeypot is deployed and then a criminal case is forced upon the operator of the honeypot. One example would be in the case of child pornography. There are several jurisdictions where, in the event that an individual witnesses child pornography, it is a criminal offense to NOT report it. Even though the intent of the honeypot was purely educational, the lack of adherence to good practice might diminish the chances of prosecuting the criminal. Another possibility is that the honeypot might be used in attacking other computers outside of the honeypot network. This is another scenario where one's procedures and systems will result in possible court involvement. There are a number of other reasons that the honeypot might be used in court that are beyond the scope of this discussion, many of which are civil based lawsuits.

Checklist of Protectionary Measures

There are four steps that should be taken to assure that in the event of a legal situation to cover the issues of due diligence; document, add banners, consult acceptable use policies and finally, containment. In the documentation step there are many things to consider. A short checklist of items to document include; a network diagram that is accurate at the time the honeypot was deployed, any communications regarding the honeypot with management, a full backup of the honeypot at the time of deployment, a copy of the Access Control List (ACL) and firewall rules at the time of deployment, and an attempt should be made to document the intent and purpose of deploying a honeypot. Other items that could be useful to document would be the current policies that might apply to computer usage or anything that might change on a frequent basis related to computer use. Legal proceedings can often occur long after the honeypot has been taken out of commission and there might be a legal need to recall what the policy was four years in the past at the time of deployment. The second step that should be taken is the installation of warning and consent banners on systems where ever applicable. This step helps ensure that there is a legal right to record and intercept traffic related to that device. Be sure to include the banner in your documentation. The third step, closely tied to the second step, is a review of Acceptable Use Policies and Terms of Service. These

policies also help ensure the legal right to record and intercept traffic as well as defining the policy for enforcing those that violate the policy that the honeypot might detect. Since these documents change frequently the Acceptable use Policy and Terms of Services should be documented at the time of deployment. The final step is to employ some form of containment for the honeypot. The containment of intruders into the honeypot will help stop any attacks that those intruders might launch from your network. Creating firewall rules that limit outbound access is a simple and effective strategy for containment. Firewalls rules are also easy to document and verify.

Conclusion

The unknown legal implications should not be a deterrent to the use of honeypot technology in your computer security toolset. The two major legal issues that we are aware of with the use of honeypots are privacy and entrapment. Both issues have significance with relation to honeypots. As with all legal situations there is safety in the form of documentation. By providing documentation, you are providing the tools that the legal system needs to defend your actions.

CyberLaw 101: A primer on US laws related to honeypot deployments

2. References

---

[1] http://honeypots.sourceforge.net/

[2] http://www.honeypots.net/

[3] http://www.lectlaw.com/def/e024.htm

[4] http://www.rbs2.com/privacy.htm

[5] 18 USC 2510 (2) http://www.usiia.org/legis/ecpa.html

[6] **ECPA** Pub. L. 99-508, Oct. 21, 1986, 100 Stat. 1848, 18 U.S.C. § 2510

[7] 18 USC 2511 (2)(a)(ii)(B)(c)

[8] 18 USC 2511 (2)(a)(ii)(B)(d)

[9] 18 USC 2511 2 (A)(i)(i)

[10] 18 USC 2510 (16)(A)

[11] 18 USC 2510 (16)

[12] Kerr, Orin, Connecticut Law Review, Winter 2001, 33 Conn. L. Rev. 503

[13] http://dictionary.law.com/default2.Asp?selected=299

[14]

http://www.microsoft.com/technet/scriptcenter/resources/qanda/jan05/hey0117.mspx

[15] Defined on http://dictionary.law.com/

Last Updated: July 27th, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
Community SANS Indianapolis SEC401	Indianapolis, IN	Oct 09, 2017 - Oct 14, 2017	Community SANS