

Project Dreddnaught

George Bobeck

25th October, 2005

*Computer Science Department
Loyola University Chicago, Chicago, Illinois, U.S.A*

gbobek@acm.org

Abstract

Network security is a vast and nebulous topic. As time progresses, this topic becomes infinitely more vast and nebulous. As a result, it becomes more difficult to implement tools to help secure networks. Project Dreddnaught is one attempt to design and implement a system to aid in securing the local network and to alert the local administrator to possible threats.

Keywords: Gentoo Linux, Knoppix, Network Security, Intrusion Detection System, Passive Ethernet Tap.

Introduction

Many administrators have attempted to have a completely secure network. This is similar to attempting to juggle ninja throwing stars, razor blades, claymore mines, or other dangerous objects. The effect is usually rather disastrous for the juggler, causing a great loss of free time and a great deal of harm to both the self and the local area. A somewhat intelligent juggler would soon realize that the task is both suicidal and practically impossible to achieve. This juggler decides to save their remaining fingers and limbs by either attempting an easier juggling task or goes out and orders a pizza.

The preceding paragraph graphically illustrated an important fact: attempting to create a completely secure, hack-proof network is practically impossible. It is much easier and successful to implement a network with the proper safeguards in place to ensure a secure, hack-resilient network. To accomplish this goal, the administrator must build and deploy an intrusion detection system which also features network and host auditing tools. This is the goal of Project Dreddnaught.

Organization of the Rest of the Paper

The rest of the paper is organized in the following manner. The first item to be discussed is a network security primer. Following this, is a discussion about the implementation of Dreddnaught. Finally, the usage of special tool will be discussed. This discussion will focus on passive ethernet taps and Linux LiveCDs.

Network Security Primer

Network security is often described with the metaphor of an onion. This metaphor is commonly referred to as the “Onion Model”. This model is used because both onions and network security are made up of layers. In network security, there are three main layers: the network layer, the paranoia layer, and the system layer. These three layers are all interconnected and equally important. A breach in one layer can weaken, if not completely devastate another layer.

The network layer is the actual network along with all connected devices. These devices include switches, hubs, routers, firewalls, virtual private network endpoints, and other miscellaneous pieces of equipment. This layer is the actual physical implementation of the network. Careful thought should be used in the planning and implementation of this layer. Ideally, this layer should be fault tolerant. This layer is where demilitarized zones are implemented. This layer is the easiest to implement.

The paranoia layer is the layer where tools for network intrusion detection, network auditing, remote logging, and event notification reside. This layer is responsible for monitoring the traffic which traverses the network layer and logging suspicious traffic and other relevant information. Likewise, this layer should alert the administrator when suspicious and malicious events (i.e. a network compromise) occur. This layer can be difficult to implement depending on the tools used and other factors.

The system layer is the remainder of the network. This layer includes the operating system and installed software on each machine on the network. This layer also includes password and security policy, software maintenance, operating system updates, and enforcement of all of items mentioned above. This is the most difficult layer to implement and maintain.

Implementation of Dreddnaught

Project Dreddnaught was designed to be a network intrusion detection system with network auditing capabilities. The information logged by the network intrusion detection software was to be logged to a database which would be accessed by a tool which would display the logged data in a human readable and search-able fashion.

Dreddnaught was implemented on Gentoo Linux [1] using the generic Gentoo kernel. The generic kernel was used because the system Dreddnaught would run on would feature a graphical user interface, which, unfortunately the security enhanced kernel could not currently support. This decision to use the generic kernel has not been detrimental to the security of the project.

Dreddnaught is using Snort [2] for network intrusion detection system software. This software set is described as a lightweight network detection system. This description is rather misleading, as Snort is quite powerful and is capable of handling a great deal of network bandwidth (depending on system hardware). In this implementation, Snort was configured to log all alerts to a MySQL [3] database. This database is queried by a software packaged named BASE [4] which then outputs the logged data in a web based human readable manner.

For network auditing purposes, a number of software tools were installed. These tools include arpwach [5], dsniff [6], etherape [7], ethereal [8], iptraf [9], nbtscan [10], and nmap [11]. As a result, Dreddnaught has the ability to monitor ARP traffic, perform penetration testing, create graphical visualizations of the network, analyze network traffic, compile network traffic statistics, map Windows based machines running NetBIOS on the network, and perform other auditing tasks through port scans. These features are important for a detailed audit of any network. In a skilled administrator's hands, these tools can be used to both discover and diagnose network troubles, and also to test the internal security of the local network. It should be noted that some of these tools are the exact same tools used by malicious users wishing to compromise the network.

Project Dreddnaught also features two custom written python scripts. These scripts are faker.py [12] and logger.py [13]. Faker.py acts as a miniature tar-pit. It listens on a given port, and when it receives traffic, it sends a small amount of data and then times out for a given time, thus resulting in the attacker's ssh brute force software or other probing software encountering a long delay. As a result, the attacker's compromise or probing attempts return no valid information and consume a great deal of time. Logger.py is a log file processor, which takes a given log file, checks to see if it has changed, converts the file to a web page, and finally emails the log file if a change occurred.

Passive Taps and Remote Snorting

In order to place Dreddnaught silently inline on a network between an internet gateway and the firewall, a passive ethernet tap used. A passive ethernet tap is useful when installing an intrusion detection system sensor or when snooping Ethernet traffic. The tap allows the intrusion detection system to monitor all traffic between two hosts in simplex. As a result, the intrusion detection system is able to monitor all traffic without itself being exposed to attack. It should be noted that the passive ethernet tap is not compatible with gigabit ethernet due to the fact that it only listens on one pair of wires.

Project Dreddnaught utilizes Linux LiveCD environments in order to perform remote intrusion detection. The data generated is logged to the database on the Dreddnaught machine. The Linux LiveCD environment used for this project is Knoppix [14]. This distribution was remastered and renamed Loyoppix [15] in honor of Loyola University Chicago. Loyoppix includes some of the same auditing tools as the Dreddnaught box.

Conclusion and Current Status

Project Dreddnaught is currently successfully implemented the network intrusion detection system software and auditing tools. It has successfully logged data. Currently, the project is in the process of writing extensive documentation in order to aid in the installation of the software used on the project box. Future goals for this project include improving install documentation, Snort rule research, and general network security guides to educate people unfamiliar with network security practices.

Bibliography

- [1] Gentoo Linux is managed and maintained at <http://www.gentoo.org/>
- [2] Snort is managed and maintained at <http://www.snort.org/>
- [3] MySQL is managed and maintained at <http://www.mysql.com/>
- [4] BASE is managed and maintained at <http://sourceforge.net/projects/secureideas/>
- [5] Arpwatch is maintained at <http://www-nrg.ee.lbl.gov/>
- [6] Dsniff is managed and maintained at <http://www.monkey.org/~dugsong/dsniff/>
- [7] Etherape is managed and maintained at <http://etherape.sourceforge.net/>
- [8] Ethereal is managed and maintained at <http://www.ethereal.com/>
- [9] Iptraf is managed and maintained at <http://cebu.mozcom.com/riker/iptraf/>
- [10] NBTScan is managed and maintained at <http://www.inetcat.org/software/nbtscan.html>
- [11] Nmap is managed and maintained at <http://www.insecure.org/nmap/>
- [12] Faker.py is managed and maintained at <http://www.cs.luc.edu/projects/comp412/dredd/downloads/>
- [13] Logger.py is managed and maintained at <http://www.cs.luc.edu/projects/comp412/dredd/downloads/>
- [14] Knoppix is managed and maintained at <http://www.knoppix.org>
- [15] Loyoppix is managed and maintained at <http://webpages.cs.luc.edu/~loyoppix/>