Vulnerability Protection
A Buffer for Patching

A Lucid Security Technical White Paper – February 2004

By Vikram Phatak, Chief Technology Officer
Santosh Pawar, Vulnerability Analyst

Lucid Security Corporation
124 South Maple Street, Suite 200
Ambler, PA 19002

http://www.lucidsecurity.com

Introduction

The purpose of this paper is to identify the problem facing the network security community regarding vulnerabilities and patches. It explains why current security technologies such as firewalls, intrusion detection and prevention systems, and automated patch management solutions have failed in preventing vulnerabilities from being exploited. Finally an alternative approach is proposed that incorporates and builds upon existing security technologies.

A Strategic Problem

The standard doctrine for network security states that the best practice for securing computer networks is a layered approach. Hardening the operating systems and applications on computers by limiting the services offered as well as installing the appropriate patches is the first step. Setting up access control to limit incoming traffic both at the boundary routers as well as through the use of firewalls comes next. The final step involves the use of intrusion detection and prevention systems to identify attackers and prohibit their access to the network.

Unfortunately, this strategy is failing and will soon be rendered inoperable due to a flaw in one of the assumptions on which it is based: patching computer systems. The root of the problem is the mounting inability for network administrators to keep computer systems patched. The issue that administrators are facing is that the rate at which vendors release patches for vulnerabilities is increasing substantially, while the time span between the announcement of new vulnerabilities and the release of exploits to those vulnerabilities is shrinking. The net effect is that system administrators have more patches to deploy and less time to do so; rendering the current strategy of mitigating vulnerabilities an overwhelming task.

Why Firewalls Fall Short

Firewalls are designed to deny all traffic and only allow certain traffic by explicit exception. This is a solid approach, but one that does not extend well into the application layer. One reason this approach does not translate into the application layer is that data payload has potentially infinite variations, and therefore it is unrealistic to state that all data payloads are denied unless explicitly allowed. The allowed rule set would be infinitely long.

Based upon this reality, most firewall vendors have avoided delving into the data payload and have instead focused upon the application protocol header. The protocol headers have acceptable usage standards which can be translated into rules for determining compliance which the firewall can then enforce. The drawback to this approach is that most attacks actually adhere to protocol standards as well as commonly accepted behavior, rendering the firewall blind to their malicious intent.

There is strong evidence to support these facts. The most undeniable evidence is that despite the widespread adoption of firewalls, attackers are getting into protected networks both directly and indirectly through the use of worms and trojans. They are getting through firewalls via ports that have been left open for applications such as web, email, DNS, and others; exploiting vulnerabilities in those applications to which the firewall allows traffic.

© 2004 Lucid Security Corporation.
All rights reserved.

The Struggling IDS/IPS

As mentioned above, most exploits today adhere to both protocol standards and commonly accepted behavior; necessitating inspection of the packet's data payload in order to determine whether or not the content is malicious. This plays squarely into the strengths of Intrusion Detection and Prevention Systems. IDS/IPS solutions are designed to allow all traffic and deny traffic by explicit exception. The benefit of this inverted logic (compared to the firewall) is that IDS solutions can delve into the data payload of packets in order to identify traffic that should be denied since they have a finite list of traffic to prohibit.

There are several problems with intrusion detection and prevention systems, however. Without regular updates to their list of attacks to watch for, they quickly fall out of date and unknowingly allow attacks to compromise vulnerable systems. Even when their attack "list" is kept up to date, IDS/IPS solutions need to be "tuned" due to the excessive number of items contained in their default "list". An untuned IDS/IPS can suffer from extremely poor performance since each packet's data payload contents needs to be inspected and compared with each item in its "list" of attacks. And even if the IDS/IPS system employs a state-machine-based parallel signature-matching engine (so that additional rules do not significantly impact performance), it still falls prey to false positives unless properly tuned. With thousands of existing attacks and many thousand more to come, tuning has become essential.

When tuning, however, most IDS/IPS administrators ignore the firewall's configuration, or group attacks based upon the service offered. For example, an IDS/IPS is tuned to identify attacks on telnet (port 23) even though the firewall only allows http traffic (port 80) or an IDS/IPS is tuned to watch for http (port 80) attacks and identifies IIS web server attacks when an Apache web server is in use. Both are examples of false positives results and not only create mountains of data for an administrator to sift through, they also negatively impact performance.

Finally, it takes a skilled individual to properly tune an IDS/IPS and that skill commands a premium in today's job market. Also, the expense associated with owning and properly maintaining an IDS/IPS solution should not be overlooked.

Patches and Patching

One solution to this problem is to patch the vulnerable system and close the security hole. This strategy has been utilized successfully for a number of years, yet it is now becoming impractical.

- **There are more patches that need to be deployed** – The rate at which vendors announce new vulnerabilities and release patches for those vulnerabilities is increasing substantially.
- **There is less time to deploy a patch before an exploit is released** – The time span between the announcement of new vulnerabilities and the release of exploits to those vulnerabilities is decreasing.
- **There are more computers that need to be patched** – The number of computers on the network has increased.
- **Deploying a patch can have an adverse effect on critical applications**
    - The operating systems of those computers vary widely, as well as their installed patches and service packs (even within all Microsoft environments).

© 2004 Lucid Security Corporation.
All rights reserved.

  - Applications are designed to work on specific builds of a given operating system. This means that when an operating system is patched, the software may or may not function properly from that point forward.
  - The applications themselves may contain vulnerabilities which require patching; if the vulnerability in the application is based upon the way it interacts with the operating system, the application may no longer function once it is patched.

- **The X Factor: The Internet**
  - Exploits are propagating more rapidly as more and more people have access to information that was once limited to small groups.
  - Companies are more exposed today than ever due to e-commerce applications which are integrated with back end business systems.
  - Just as web services have become prolific, so have organizations become reliant on those services.

To Patch or Not To Patch

The fact is that nearly every application and operating system running on a network has to be patched at some point. Many applications and operating systems require a lot of patches over time. Some of those patches themselves require patching. Patching requires some thought and analysis, since patches have been known to break applications as well as introduce new vulnerabilities themselves.

The security update that was released by Microsoft in January 2001 to patch a security exploit in Exchange 2000 server actually required a patching of its own.

Automated patch management is a promising, yet still developing technology. Some of the issues that need to be addressed by an automated patch management system are:

- What will be the affect of installing a patch on a system that's running a combination of these other applications? (for instance Apache and MySQL running on the same server)
- Is it possible to "roll back" a patch if it creates problems?
- The patch that was released last month was never applied. Can I install the latest patch directly without installing the previous one?

Traditionally, network administrators are not willing to install anything on a critical system that they don't feel confident about. As a result the idea of automatically installing patches has had a difficult time gaining traction. Automated patch management systems will fail to realize their potential as a tool until they can identify the various applications on a given system and guarantee that the patches that are automatically deployed will not conflict with other applications or the operating system, and that the system will function in its intended way after applying a patch.

"The average enterprise deals with up to 80 patches a year, and a stunning 95% of attacks and security incursions take place after the patch intended to prevent them has been announced" says Keith Ferrell on www.techweb.com
(ref: http://www.techweb.com/tech/security/20030611_security.)

Due to no fault of the administrator; many systems are left unpatched and vulnerable while he/she struggles to apply patches in a sane fashion.

© 2004 Lucid Security Corporation.
All rights reserved.

What is the solution?

It is clear that the administrator needs more time to determine which systems need patching, which don't, and which can't be patched. The answer is two-fold: Additional development in the automated patch management field as well as a new type of product that can act as a buffer by stripping malicious code from relevant traffic. The first is simply a matter of time and money. The second can be accomplished by adapting existing IDS/IPS technology so that it is a useful tool for the purpose of Vulnerability Protection.

Effective Vulnerability Protection requires an approach that focuses on what can harm a given system as opposed to looking for "who is attacking my network" (like an IDS). This shift in thinking about attacks in terms of vectors and layers is based upon certain assumptions:

- Not every attack harms every system
  - "Code Red" only effects Windows systems and does not impact Linux or Unix systems.
  - IIS web server "Cross Site Scripting" attacks do not impact Apache web servers.
- Not all traffic is allowed through the firewall
  - Telnet (port 23) is not allowed by the firewall; therefore attacks against telnet (port 23) will not get to the target system.

The key to this new approach hinges on a two-step process that automatically determines:

1) What resources are exposed to attack and which are not
2) … and for each exposed resource (host), what vulnerabilities exist and which do not.

This would enable the modified Intrusion Prevention System to identify which traffic to watch; preventing relevant exploits from getting to a vulnerable target by dropping the malicious data payload. The net effect of this solution is that it would protect vulnerable systems from being exploited until a patch is applied.

About ipANGEL™

ipANGEL approaches Intrusion Prevention differently than other IPS solutions by focusing on vulnerability protection. ipANGEL actively discovers and guards security holes; buying administrators the time to plan and test patches and upgrades. In addition, ipANGEL is easy to use and requires little maintenance due to its self-tuning and auto-update features. The result is that ipANGEL solves an important problem while making Intrusion Prevention practical and affordable for a wide variety of customers.

About Lucid Security

Headquartered in suburban Philadelphia, Lucid Security is a leading developer of next generation security software that defends networks against attacks; providing real-time vulnerability protection. Lucid Security's product, **ipANGEL**, is affordable, easy to use, and provides unmatched protection against attacks. In December 2003, **ipANGEL** was named **"Best Emerging Technology"** by Information Security Magazine.

© 2004 Lucid Security Corporation.
All rights reserved.