Firewire forensics in modern operating systems

P. Hannay[1] **and A. Woodward**[1]
[1]School of Computer and Security Science, Edith Cowan University, Perth, Western Australia, Australia

Abstract – *This research looked at whether the FireWire direct memory access function tool would work with three modern Windows operating systems. The tool requires local access to the PC and allows the logon to be bypassed, and also allows for memory dumping to be performed on the target computer. It was found that Windows XP allowed for full access and memory dumping, while Windows Vista and Windows 7 allowed for memory dumping only. The inability to unlock the two newer operating systems appears to be a product of a change in memory location of the target data, rather than a fix. This has implications for digital forensics in that keys to some encryption programs can be found in memory.*

Keywords: FireWire, IEEE 1394, Windows, digital forensics

1 Introduction

This paper builds on previous work that developed the Firewire unlock / memory dumping tool for use with Windows XP [1]. The idea was originally published by Dornseif to bypass security features of Apple Macintosh operating systems, and extended also to work with linux operating systems [2]. The tool itself, called Winlockpwn [1], has several modes of operation, the most relevant to this investigation being the ability to bypass a windows password. Implications for use of this tool to in digital forensics to access Windows XP based operating systems have already been investigated [3]. It was found that the tool was useful for unlocking locked workstations, enabling the logon password to be circumvented, allowing access to files and folders. It also allowed access to encrypted file system (EFS) protected files and folders under certain circumstances.

The tool also allows for the running of a command prompt without having to log on to the computer, and also to perform a memory dump. This type of attack is possible because the FireWire protocol, IEEE 1394 [4] is designed to be an expansion bus device. An expansion bus is designed to allow for additional functionality to be added to a computer with effectively full access to on-board RAM and CPU, as would any motherboard component. As part of the requirement of an expansion bus, and in particular the FireWire protocol itself, direct memory access is required [4]. Whilst this allows for devices connected using this method to transfer data at high speed, it is also the reason as to why the password unlock tool is able to function.

Because the Firewire standard allows for memory to be mapped, and thus manipulated, the Winlockpwn tool is able to search memory to find the location of the values and routines which are responsible for authenticating passwords. It can then over-write or change parts of the programs responsible for authentication, effectively bypassing the logon requirement.

There are numerous methods available to protect or encrypt data at rest. There is encrypting file system (EFS), which is included in Microsoft Windows operating systems from Windows 2000 onwards [4]. This is capable of protecting both files and folders. Additionally, there are some versions of Microsoft's latest operating system, Windows Vista, contain whole of disk encryption system known as BitLocker [5]. There are also third party applications available to protect data such as TrueCrypt which creates a virtual encrypted disk, which can be mounted as though it were an actual disk [6]. This program also contains two additional features which effectively hide the encrypted volume, making it both hard to detect and decrypt. Another utility which provides similar functionality is FineCrypt [7]. This tool also provides the ability to encrypt files, directories and trees. Given that the software uses 256 bit AES encryption, brute force is not really a viable option [8].

While the above tools were created with a view to protecting legitimate information at rest on a computer, these tools are also available to those who wish to use it to cover up traces of illegal activity. This creates an issue for law enforcement as they cannot view files on a suspect machine if they have been protected using either a password or some form of encryption. There are several methods available to circumvent these protective measures. These include brute force password crackers such as John the Ripper [9], password reset utilities such as chntpw [10] and even self booting CDs with a collection of security tools such as BackTrack [11].

This research examined the effectiveness of the Winlockpwn tool against a range of parameters to determine the extent of its usefulness as a forensic unlock tool. The Factors to be examined included whether the target computer was a member of a Windows workgroup, or whether it was a member of an Active Directory domain. Another factor to be examined was whether the tool would allow for the viewing of files protected with EFS on the target machine. Also of interest was the memory dump mode, which may allow for keys to be retrieved from the memory of a target computer.

2 Methodology

The main aim of this research was to establish whether the tool worked as stated by the author. To this end, the following computer setup was used:

Attacker: IBM Intellistation 3.4GHz Pentium 4 CPU, 2GB RAM, onboard FireWire card. Debian Linux

Target: IBM S50 desktop PC 3.0GHz, 1GB RAM, Microsoft Windows XP (SP2), PCI FireWire card.

Operating Systems: Windows XP (SP3), Windows Vista (SP1) and Windows 7 (beta build 7000)

The efficacy of the tool was tested in several ways. Firstly, its ability to obtain access to a computer that had been switched off: referred to in this paper as cold boot. Secondly, the tools ability to unlock a password protected and locked workstation. The tool was tested for both local and domain accounts for these scenarios. Thirdly, the ability to examine or view EFS protected files and folders were examined using both a locked workstation and a cold boot situation for both local and domain accounts. Lastly, all operating systems were tested with the tool to determine whether a memory dump could be performed.

Various other behaviours of interest and relevance to the topic were observed as part of the testing process and are also reported here.

3 Results

There were a number of parameters tested to determine the extent to which various security measures could be bypassed using the FireWire memory tool.

3.1 Password bypass

The initial testing for this phase was to determine whether the password bypass feature of Wihnlockpwn would work for each of the three operating systems being tested. It was quickly determined that it would only work for Windows XP, and not for Windows Vista or Windows 7. As such, the following results relate to Windows XP only.

The first test was to determine the efficacy of the tool to circumvent the password logon for a computer that was a member of a workstation can be found in Table 1.

Table 1 – Efficacy of the FireWire Windows XP unlock tool against a computer that was a member of a Windows workgroup.

Tool attack mode	Workstation Locked	"Cold Boot" logon
Unlock	Yes	Yes
Unlock with command prompt	Yes	Yes

The next tests conducted were to determine whether the tool would unlock a computer that was a member of an Active Directory domain (Table 2).

Table 2 - Efficacy of the FireWire Windows XP unlock tool against a computer that was a member of an Active Directory Domain.

Tool attack mode	Workstation Locked	"Cold Boot" logon
Unlock	Yes	No
Unlock with command prompt	Yes	No

3.2 EFS

Files protected by EFS could be accessed but only under a finite set of circumstances, and only under Windows XP. No EFS access was possible for either Windows Vista or Windows 7 as the password bypass did not work. All subsequent results for EFS bypass reported here refer to Windows XP only. Effectively, access to EFS protected files under Windows XP was only possible if the legitimate user had already viewed the file and locked the workstation. Rebooting the workstation or logging off removed the ability to view EFS protected files.

Bypassing the Windows login from cold boot provided access to the underlying file system, but did not allow access to files protected with EFS.

A folder protected with EFS could be viewed, and the contents of the folder also viewed, but as pointed out previously, EFS protected files could not be viewed, unless they had previously been viewed by the legitimate user.

3.3 Memory Dump

The Winlockpwn tool was able to perform a memory dump from a running computer for all three of the operating systems tested, namely, Windows XP, Windows Vista and Windows 7.

3.4 Other observations

Once the computer has been unlocked using the tool, the ability to log in without a password was still existent even after a legitimate log off and log on.

The stated aim of mode four of the tool is to allow for a command prompt to be opened on the target system. This was found to be the case, and further investigation showed that an alternate desktop could be run from this command prompt. Further to this, the task manager showed that this tool was running with system privileges, a level of access higher than that granted to an administrator.

4 Discussion

The FireWire tool was found to be successful in a number of scenarios against a password protected computer running Windows XP, but not against either Windows Vista or Windows 7. It allowed for a number of different attacks to be made against a computer, including bypassing the windows logon password, and running a high privilege command prompt without logging in. However, the memory dump feature of the tool did work against al three operating systems investigated. The attack itself was reasonably trivial, but some modification of the code was necessary to get it to run on our test equipment, but once running, it takes only a few seconds to unlock a locked workstation, or to perform a memory dump of a target computer.

4.1 Password unlock

The password bypass feature works well, and while it seems to be the most prominent feature of the tool, it could also be argued that from a forensic perspective, it is the least significant. This statement is based on the observation that EFS protected files could not be viewed other than for a limited set of circumstances. Additionally, it is difficult to guage what impact the memory fuzzing to achieve the password unlock has on the forensic validity of any information obtained.

While the lack of ability to bypass the password in the two most recent Windows operating systems appears to be a disadvantage, this is not necessarily the case. There are other existing means of bypassing or circumventing security measures, particularly if you have local access. One such tool is the chntpw utility which was designed to quickly and easily reset the passwords for any local user account on a computer [11]. The chntpw utility also requires local access to the machine, similar to the Firewire unlock tool used in this research. The disadvantage of such an approach over the Firewire tool used in this research is that resetting the passwords in the local security account manager (SAM) database will render any EFS encrypted files and folders permanently unreadable.

4.2 Memory dump mode

While it would appear that the password bypass is probably the most useful of these attack modes, the memory dump has greater implications for digital forensics. If the target computer is using some form of cryptography, then it may be possible to retrieve the keys from memory. There has been other research which has looked at this process, the most recent and popularized being the co called ‘cold boot’ attack [14]. In this attack, the physical memory modules are literally frozen and then dumped using a memory dump is then analysed to extract any cryptographic keys that may be resident. In this case, the researchers claim to be able to extract Windows BitLocker keys from memory. BitLocker is Microsoft technology incorporated into Windows Vista that provides whole of disk encryption.

4.3 Operating system specific issues

It is worth noting that the code used to perform the attacks, the so-called Firewire unlock tool, was at proof of concept stage, and required extensive modification of both attack platform and the tool itself in order to achieve the results obtained. As to why the tool was not able to unlock Windows Vista or Windows 7 computers, the answer lies in the way that memory is allocated in these newer operating systems. It has been reported that the Firewire tool can successfully unlock Windows Vista operating systems [17]. In this paper, the author has successfully modified the code in order to get it successfully unlock a Windows Vista computer. The password bypass is performed by patching the memory location that contains the instruction belonging to Msv1_0.dll, which is responsible for authenticating against the SAM database.

4.4 Mitigation

Whilst this paper has researched the Winlockpwn tool from a legitimate purpose, the reality is that it is an attack tool which could easily be used for illegitimate or illegal purpose. From that perspective, the tool is effectively a Windows attack tool, and one which has potential for great harm as it allows a bypass of the primary security measure: the logon password. What then is the possibility that the operating system will be patched to prevent this exploit? The likelihood is that there will be no fix, and this is due to a number of reasons. Firstly, the exploit makes use of the required functionality of Firewire, which is direct memory access. Without this access, you would not achieve the high speed throughout that is achieved by Firewire devices, and the standard would cease to exist, and a great number of Firewire enabled devices would be rendered largely useless. Additionally, the exploit requires local access, and local access means that a computer is then subject to a great number of other exploits, most of which would be easier to perform than the Winlockpwn attack. Therefore, whilst the Firewire standard makes use of the extension bus, and while

Firewire devices are still supported by Windows (and other) operating systems, it seems unlikely that the exploit will be fixed any time soon.

5 Conclusion

This paper has demonstrated that the FireWire exploit tool has the potential to be used by forensic investigators. The main advantage of using such a tool would be to allow digital forensic practitioners to bypass the logon password, potentially saving hours or days of time cracking a password which could otherwise be used for investigation. Whilst its primary use would be to circumvent password protection, there are a number of other uses it could be put to, such as identification of stolen laptops, and viewing of EFS protected files.

Possibly of greater potential is the ability to retrieve cryptographic keys from memory with the password bypass feature. It was encouraging that the Winlockpwn tool was able to successfully perform a memory dump against all target operating systems.

Future research by the authors will focus on modifying the tool to work with Microsoft Windows Vista and Windows 7. The main question to be examined from an investigative perspective is the possibility of locating cryptographic keys for encryption or other security programs that may be protecting files or folders on the hard drive.

6 References

[1] Adam Boileau, “Hit by a bus : Physical access attacks with FireWire”, 2006. [Online]. Available: http://www.ruxcon.org.au/2006-presentations.shtml#14 [Accessed January 23, 2009]

[2] Maximillian Dornseif, “FireWire - all your memory are belong to us”, 2004. [Online]. Available: http://md.hudora.de/presentations/firewire/PacSec2004.pdf [Accessed January 23, 2009]

[3] Woodward, A. & Hannay, P. (2008). Forensic implications of using the FireWire memory exploit with Microsoft Windows XP. In Proceedings of the Las Vegas, Nevada 2008

[4] IEEE (1996). IEEE standard for a high speed serial bus. New York, USA: The Institute of Electrical And Electronics Engineers, Inc.

[5] Microsoft Technet, “Protecting Data by Using EFS to Encrypt Hard Drives”, 2009. [Online]. Available: http://www.microsoft.com/technet/security/smallbusiness/topics/Cryptographyetc/protect_data_efs.mspx [Accessed January 23, 2008]

[6] Microsoft Technet, “BitLocker drive encryption”, 2009. [Online]. Available: http://technet.microsoft.com/en-us/windows/aa905065.aspx [Accessed January 23, 2009]

[7] TrueCrypt, “Free open source disk encryption”, 2009. [Online]. Available: http://www.truecrypt.org/ [Accessed February 10, 2009]

[8] FineCrypt, “FineCrypt – Professional encryption tool”, 2009. [Online]. Available: http://www.finecrypt.net/index.htm [Accessed February 10, 2009]

[9] J. Siegfried, C. Siedsma, B-J Countryman & C.D. Hosmer, “Examining the encryption threat”, International Journal of Digital Evidence, Winter 2004, Vol 2, Issue 3

[10] Openwall, “John the Ripper password cracker”, n.d. [Online]. Available: http://www.openwall.com/john/ [Accessed February 10, 2009)

[11] Nordhal, P. “Office NT password and Registry Editor” Change nt password”, 2008. [Online]. Available http://home.eunet.no/pnordahl/ntpasswd/ [Accessed February 10, 2009]

[12] Remote-Exploit, “BackTrack”, 2008. [Online]. Available: http://www.remote-exploit.org/backtrack.html [Accessed February 13, 2009]

[13] Microsoft Help and Support, “Cached credentials security in Windows Server 2003, in Windows XP, and in Windows 2000”, 2009. [Online]. Available: http://support.microsoft.com/kb/913485/en-us [Accessed February 15, 2008]

[14] Irongeek, “Cracking Cached Domain/Active Directory Passwords on Windows XP/2000/2003”, 2006, March. [Online]. Available: http://www.irongeek.com/i.php?page=security/cachecrack [Accessed February 15, 2009]

[15] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum, and E. W. Felten, “Lest We Remember: Cold Boot Attacks on Encryption Keys”, 2008. [Online]. Available: http://citp.princeton.edu/pub/coldboot.pdf [Accessed January 28, 2009]

[16] AUSCERT, “2006 Australian computer crime and Security survey”, 2006. [Online]. Available: http://www.auscert.org.au/images/ACCSS2006.pdf [Accessed February 15, 2009]

[17] Panholzer, P. (2008). Physical security attacks on Windows Vista. Available https://www.sec-consult.com/files/Vista_Physical_Attacks.pdf [Accessed February 15 2009]