
Steganographic Information Hiding that Exploits a Novel File System Vulnerability

Avinash Srinivasan* and Satish Kolli

Volgenau School of Engineering
George Mason University, Fairfax, VA 22030
Email: [asriniv5, skolli]@gmu.edu (*Corresponding author)

Jie Wu

Computer and Information Sciences Department
Temple University, Philadelphia, PA 19122
Email: jiewu@temple.edu

Abstract: A widely used benchmark for evaluation of information systems security focuses on three core goals— *Confidentiality*, *Integrity* and *Availability* of information (a.k.a. **CIA** of security). In a file system, it is the integrity component that ensures all files and folders have unique names and/or paths. In this paper, we present and discuss *DupeFile*, a simple yet critical security (integrity) vulnerability in numerous file systems. By exploiting *DupeFile*, one can store two or more files with the same name and path, with different contents, inside the same volume. Consequently, data exfiltration that exploits *DupeFile* vulnerability, which we call *DupeFile Hiding*, becomes simple and easy to execute. In *DupeFile Hiding*, a known good file is chosen whose name serves as the cover for hiding the malicious file. Hence we classify *DupeFile Hiding* as a Steganography technique. *DupeFile Hiding* is neither a file-compression nor a data-embedding technique, unlike contemporary Steganography techniques. It merely exploits the name and reputation of a known good file to store malicious file(s). The same vulnerability can also be exploited for legitimate applications such as hiding password files, project blueprints, product license, DRM, etc. This vulnerability was first uncovered on a FAT12 formatted disk on Windows 98 VM. Nonetheless, the vulnerability exists in numerous file systems, including NTFS, HFS+, and HFS+Journaled. Finally, in this paper, we discuss our customized tools: *DupeFile Detector*- detecting hidden files; and *DupeFile Extractor*- recovering hidden files; tools we have developed to counter *DupeFile Hiding*. We have also developed *DupeFile Creator* for hiding files in legitimate applications; built solely for research purposes.

Keywords: Data hiding, integrity, security, steganography, vulnerability.

Reference to this paper should be made as follows: Srinivasan et. al. ‘Steganographic Information Hiding that Exploits a Novel File System Vulnerability’, *Int. J. Security and Networks*, Vol. x, No. x, pp.xxx-xxx.

Biographical notes: Prof. Avinash Srinivasan is currently a faculty member in the Computer Science Department at George Mason University. His research interests include network security and forensics, forensic analysis of file systems, forensic file carving, and security in WSNs & MANETs. He has published 30+ papers in scholarly conferences and journals including IEEE INFOCOM and ACM SAC.

Satish Kolli received his M.S in Computer Science from Johns Hopkins Univ. Currently he is a PhD student in Info. Security and Assurance at George Mason University. His research interests include information security and protocol analysis.

Prof. Jie Wu is the Chair and a Laura H. Carnell Professor in the Department of Computer and Information Sciences at Temple University, USA. Prior to joining Temple Univ., he was a program director at the National Science Foundation and Distinguished Professor at Florida Atlantic University. His research interests include wireless networks, mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. Dr. Wu’s publications include over 600 papers in scholarly journals, conference proceedings, and books. He has served on several editorial boards, including IEEE Transactions on Computers and JPDC. Dr. Wu was general co-chair for IEEE MASS-2006, IEEE IPDPS-2008, and IEEE DCOSS-2009 and was the program co-chair for IEEE INFOCOMM-2011. Currently, Dr. Wu is an ACM Distinguished Speaker and a Fellow of the IEEE.

1 Introduction

Steganography comes from the Greek word *steganos* meaning covered writing. It is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message. This is also referred to as *Security through obscurity*. The idea and practice of hiding *information exchanges*— aka Steganography— has a long history. Traditional techniques of steganography ranged from tattooing the shaved head of a trusted messenger (as reported by the 5th century Greek historian Herodotus) to using “Invisible Ink” and “Microdot” during the two World Wars.

Steganography includes information hiding within computer files, such as an image file, audio file, or a video file. It uses a simple and seemingly harmless file as the cover file, hiding the malicious data underneath. The hiding process does not alter the content of the cover medium to an extent that is easily recognizable. More advanced techniques hide with such effectiveness that even statistical methods of detection can be evaded seemingly easily. Several techniques have been developed to detect information hiding; accomplished by various Steganographic tools that employ a limited number of Steganographic algorithms. However, the adversary has been consistently successful in developing new techniques to achieve evasion. Figure presents the taxonomy of information hiding techniques, while Figure presents the taxonomy of steganographic techniques.

Modern steganography employs digital media content as camouflage, powerful computers and signal-processing techniques to hide secret data, and methods to distribute stego-media throughout cyberspace, thus

posing a serious challenge to scientists and professionals alike in the field of information security ?. Especially for the digital forensic community, Steganography has been a great challenge from the very beginning. Nonetheless, one has to be prudent and unbiased to recognize the good side of Steganography, such as digital copyrighting and watermarking.

It is well known that one of the most widely used benchmarks for evaluation of information systems security focuses on the three core goals— *Confidentiality*, *Integrity* and *Availability* of information. These three core goals are often collectively referred to as the “CIA of security”. While all the three core goals are equally important for the security of a system, depending on the nature of the information and the corresponding domain, one or more of these three core goals can weigh in more than the other(s). In a well designed and implemented file system, which is the primary focus of this paper, all the three core goals of security have to be met. However, it is because of the integrity component of a file system that all the files and folders have unique names and/or paths.

In this paper, we present and discuss *DupeFile*, a simple yet critical security vulnerability that exists in numerous file systems. More specifically, *DupeFile* is a file system integrity vulnerability. This vulnerability was first discovered on a FAT12 formatted disk on a Windows 98 Virtual Machine. Precisely, the vulnerability was encountered while recovering deleted files, in the aforementioned environment, using DiskEdit, a Hexeditor developed by Norton Utilities.¹ The vulnerability exists across Microsoft’s proprietary File Allocation Table (aka FAT) file system family, which includes FAT12, FAT16, and FAT32. It also exists on other Microsoft and Apple proprietary file systems, including NTFS, HFS+, HFS+ Journaled, etc.