YÖNETİM BİLİŞİM SİSTEMLERİ DERGİSİ
http://dergipark.ulakbim.gov.tr/ybs/

Yayın Geliş Tarihi: 30.05.2016
Yayına Kabul Tarihi: 15.08.2016
Online Yayın Tarihi: 05.10.2016

Cilt:1, Sayı:3, Yıl:2016, Sayfa 19-28
ISSN: 2148-3752

ADLİ BİLİŞİMDE SİLİNMİŞ DOSYALARIN KURTARILMASI ÜZERİNE KARŞILAŞTIRMALI YÖNTEMLER

Yusuf Güllüce

Recep Benzer

Hüseyin Çakır

Öz

Veri kurtarma çalışmalarında temel olarak iki yöntem uygulanır. Bunlardan birincisi dosya sistemlerinin analiz edilerek yapılan veri kurtarma çalışmaları, ikincisi ise dosyaların yapısal özelliklerinden yola çıkılarak yapılan çalışmalardır. Çalışmamızda öncelikle dosya sistemleri ve çalışmamızın konusu olan NTFS dosya sistemi hakkında bilgi verilmiştir. Daha sonra NTFS dosya sisteminde bir dosyanın oluşturulması ve silinmesi işlemlerinin arka planda ne şekilde gerçekleştiği ortaya konulmaya çalışılmıştır. Bu bağlamda veri kurtarma çalışmalarında uygulanan iki farklı yöntem, avantaj ve dezavantajları tartışıldıktan sonra örnek bir çalışma üzerinde uygulanmıştır.

Anahtar Kelimeler: Veri Kurtarma, NTFS, Dosya Sistemi, Üstbilgi, Dosya Formatı.

COMPARATIVE METHODS ON RECOVERING THE DELETED FILE AT DIGITAL FORENSICS

Abstract

Two different methods are used in data recovery studies. The first method is doing data recovery by analyzing the first file system. The second is done by considering file structures. In our study, firstly information about the file systems and NTFS file system that is the subject of our study are given. Then, we tried to explain how a file creation and deletion took place in the background of NTFS file system. In this context, two different data recovery methods are applied on a sample work after discussing the advantages and disadvantages.

Keywords: Data Recovery, NTFS, File System, Metadata, File Format.

1.GİRİŞ

Silinmiş verilerin kurtarılmasında temelde iki farklı yöntem uygulanır. Bunlardan birincisi dosya sisteminin analiz edilmesi ile dosya sistemi kayıtlarında silinmiş olarak işaretli olan kayıtların tespit edilmesi ve kurtarılması yöntemidir.

İkinci yöntemde dosya sistemi göz önünde bulundurulmaksızın doğrudan dosya türlerini tanımlayıcı ibarelerin diskin içerisinde taranması ve bu şekilde dosyaların kurtarılması yöntemidir. "File Carving" olarak nitelendirilen bu yöntemde örneğin "jpg" uzantılı bir resim dosyası için dosya başlangıç ve bitiş imzaları taranır ve diskte tespit edilen bu iki değer arasında kalan veriler "jpg" uzantılı resim dosyalarına ait olarak işaretlenir.

Dosya sistemi üzerinden veri kurtarma ile "File Carving" yöntemleri arasında önemli farklar bulunmaktadır. (Beek, 2011, s.3) Bu iki farklı yöntemin avantajları ve dezavantajları bulunmakla birlikte birinin diğerine mutlak üstünlüğü söz konusu değildir. Bu nedenle çalışmamızın içerisinde bu yöntemler açıklanacak ve karşılaştırmalar yapılacaktır.

Bunun öncesinde birinci yöntem kapsamında NTFS dosya sistemi ve ikinci yöntem kapsamında dosyalar ve imzaları ile ilgili açıklamalar yapılacaktır.

2.DOSYA SİSTEMLERİ

Verinin kalıcı olarak saklanabilmesi ve gerektiğinde ulaşılabilmesi hususları dosya sistemlerine duyulan ihtiyacın temel nedenleridir. (Carrier, 2005, s. 129) Bir diskte herhangi bir dosyanın nereye ve ne şekilde kaydedileceğini organize eden, daha sonra bu veriye ulaşılmak istendiğinde bu dosyayı bulup kullanıcıya getiren mekanizma "dosya sistemi" olarak adlandırılır.

Diskler temelde kapasitelerine göre veri depolanabilir 512 baytlık (yeni nesil disklerde 4096 bayt) sektörlerden oluşmaktadır. Bununla birlikte örneğin 2 TB kapasiteli bir diskte sıfırdan başlayıp yaklaşık dört milyara kadar devam eden sıralı sektörlerin veri depolama için kullanılabilir hale gelmesi için öncelikle disk üzerinde dosya sistemlerinin çalışabileceği bölümlemenin yapılması gerekir. Hard disklerde ilk sektörde bulunan MBR (Master Boot Record) kaydı disk içerisinde kaç bölüm olduğu, hangi sektörde başlayıp hangi sektörde bittiğini, bölüm içerisinde hangi dosya sisteminin bulunduğu gibi bilgileri tutar. MBR sistemi en fazla dört adet bölümleme yapılabilmesine olanak verir. Bu sınırlamanın kaldırılabilmesi için GPT bölümleme sistemi geliştirilmiştir. Her bölümün ilk sektöründe ise dosya sisteminin "boot" kaydı yani VBR (Volume Boot Record) bulunur. (Medeiros, 2008, s.10) USB bellek veya hafıza kartı gibi depolama birimlerinde ise normal şartlarda bölümleme yapısı bulunmaz ve doğrudan tek bölüm şeklinde çalışırlar.

Disk bölümlerini veri depolama açısından kullanılabilir hale getiren ve tüm sektörleri adreslenebilir hale getiren dosya sistemidir. Her bölüm için ayrı dosya sistemleri kurulabileceğinden bir diskte farklı bölümlerde farklı dosya sistemleri çalışabilir.

Dosya sistemlerinin çalışma yapıları farklılık gösterebilmektedir. Bununla birlikte genellikle tüm dosya sistemleri dosya üstbilgilerini ve dosya içeriklerini diskin farklı yerlerinde tutarlar. Bu durumu bir kitaba benzeterek açıklamak gerekirse, kitabın içindekiler kısmında bulunan konu başlıklarını dosya üstbilgileri olarak, kitabın ilerleyen sayfalarındaki konu içeriklerini ise dosya içerikleri olarak düşünebiliriz. Bu durumda dosya sistemlerinin nasıl çalıştıklarını anlamak için dosya üstbilgilerinin (metadata) nasıl tutulduğu ve sistemin dosya içeriklerinin kayıtlı olduğu yerleri nasıl adreslediğini anlamak gerekir.

Günümüzde NTFS, FAT 32, EXT2/3/4 ve HFS+ gibi dosya sistemleri yaygın olarak kullanılmakla birlikte en aşina olduğumuz dosya sistemi NTFS'dir. Çünkü güncel Windows

işletim sistemleri NTFS dosya sistemini kullanırlar. Bu çerçevede çalışmamızın kapsamını NTFS dosya sistemi ile sınırlandırılmıştır.

2.1.NTFS Dosya Sistemi (The New Technologies File System)

Windows NT sürümünden itibaren Windows işletim sistemleri için "default" dosya sistemi olan NTFS bugün dünyada en çok kullanılan dosya sistemidir. NTFS tüm yönetimsel araçları kullanıcı dosyaları gibi birer dosya şeklinde tutmaktadır. Bu dosyalar üst dosyalar olarak (metafile) isimlendirilmiştir. Spesifik sektörlerin dosya sistemi tarafından ayrıldığı ve kullanıcıların kullanımına kapalı olduğu diğer dosya sistemlerinden NTFS'i ayıran en önemli hususlardan biri her sektörün bir dosyaya tahsis edilmiş olmasıdır.

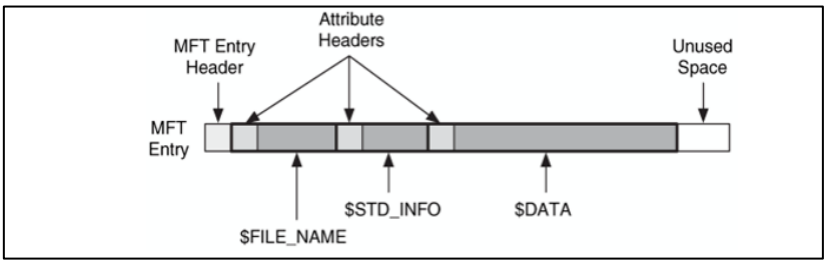
NTFS yönetimsel fonksiyonlarını yerine getiren üst dosyalar kısaca açıklanacaktır. (http://ntfs.com/ntfs-system-files.htm) Bu dosyalar içerisinde indeksleme görevini üstlenen MFT dosyasıdır.

2.1.1.MFT

Dosya ve klasör üstbilgileri MFT isimli bir dosya içerisinde tutmaktadır. MFT'nin hangi adreste bulunduğu ise dosya sisteminin bulunduğu bölümün ilk sektöründeki "boot" kaydında bulunmaktadır. MFT dosyası her dosya veya klasör için ayrılmış 1024 baytlık kayıtlardan meydana gelir. (Fellows, 2005, s. 3) İlk kayıt kendine yani $MFT dosyasına aittir. Her kayıt içerisinde dosyanın ismi, oluşturma, erişim tarihleri gibi üstbilgiler bulunur. Kritik önemdeki bu dosyanın genellikle ilk dört kaydın bir yedeği MFTMirr dosyasında tutulmaktadır.

Her MFT kaydı bölümlerden oluşur. MFT kayıtları içerisinde bulunabilecek bölümler (attributes) $AttrDef içerisinde tanımlanır. (Russon ve Fledel, 2004, s. 10) 1024 bayt boyutundaki bir MFT kaydının temel yapısı ve kayıt içerisindeki çeşitli bölümler aşağıda açıklanacaktır.

Şekil 1. MFT kaydının genel yapısı



MFT Header: MFT kaydının başlangıcında "FILE" veya sorunlu kayıtlar için "BAAD" değeri bulunur. Ayrıca bu bölümde, hataların tespiti amacıyla konulan "fixup" değerleri, MFT kaydının boyutu, kullanılan alan, dosyanın kullanımda olup olmadığı ile ilgili olarak ve kaydın bir dosyaya mı veya klasöre mi ait olduğuna dair işaretler yer almaktadır.

$FILE_NAME: Bu bölümde dosyanın hangi klasörde bulunduğu, dosya oluşturma, değiştirme, erişim ve MFT kaydı değişim tarihleri, dosyaya ayrılmış alan, gerçek dosya boyutu, işaretler (gizli, sistemi arşiv vs.) ve dosya ismi gibi bilgiler bulunur.

$STANDARD_INFORMATION: Bu bölümde ise yine oluşturma, dosya değişim, MFT değişim, dosya erişim tarihleri, sınıf ID, sahip ID gibi bilgiler tutulur. Windows dosya özellikleri kısmındaki tarihler bu bölümden gelmektedir. (Cho, 2013, s.4)

$DATA: Bu bölümde ise dosya içeriği bulunmaktadır. (Alazab, Venkatraman ve Watters, 2009, s.4) Dosya içeriği MFT kaydına ayrılmış 1024 baytlık alandan kalan kısma

sığarsa dosya içeriği için ayrıca bir alan ayrılmamaktadır. Dolayısıyla küçük dosyaların içeriği MFT kaydı içerisinde tutulmaktadır. Ancak daha büyük boyutlu dosyalar için bu dosya içeriklerinin hangi adreslerde tutulduğuna ilişkin listeler $DATA bölümüne kaydedilmektedir.

Logfile: Dosya üstbilgilerindeki değişiklikleri arşivler.

Volume: Bölüm etiketi, dosya sistem versiyonu gibi bölüm ile ilgili bilgileri barındırır.

AttrDef: MFT kayıtlarındaki dosya özelliklerini tutan tablodur. NTFS dosya sistemi dosya adı, dosya erişim yetkisi veya dosya içeriği gibi her şeyi dosya özelliği olarak görür. AttrDef dosyasında da bu özellikler, isimleri ve ID numaraları bulunmaktadır.

Bitmap: Bu dosya içerisinde bulunan her bir sektörün bir dosya tarafından kullanılıp kullanılmadığını ifade eder. (Bui, Enyeart ve Luong, 2003, s. 7)

NTFS dosya sisteminde bu üst dosyalar dışında $BadClus, $Secure, $UpCase gibi başkaca sistemsel dosyalar da bulunmaktadır.

Dosya sistemleri nasıl ki dosyalara ait verilerin diskte nerelere kaydedileceğine karar veriyorsa aynı şekilde silinmiş bir dosyanın diskte kayıtlı verilerine ne olacağına da karar vermektedir. Yani her dosya sistemi veri kaydetme ve veri silme prensipleri açısından farklı yöntemler izleyebilir. Çalışmamızın kapsamı NTFS Dosya sistemi ile sınırlı olduğu için aşağıda NTFS dosya sisteminde dosya oluşturma ve dosya silme işlemlerinin arka planda nasıl gerçekleştiğini inceleyeceğiz.

2.2.NTFS Dosya Sisteminde Bir Dosya Oluşturmak

1. MFT dosyası içerindeki kullanımda olmayan ilk kayıt $BITMAP taranarak tespit edilir ve burası yeni oluşturulacak kayıt için temizlenir. $BITMAP'te ilgili yer 1'e çevrilir.
2. $FILE_NAME ve $STANDARD_INFORMATION bölümleri MFT kaydında oluşturulur.
3. Dosya oluşturma ve diğer tarihler ayarlanır.
4. Dosyanın kullanımda olduğuna dair (in-use) işaret konur.
5. $BITMAP dosyasından disk üzerinde dosyanın boyutuna göre en uygun yer tespit edilerek bu sektörlere dosya verisi kaydedilir. Dosya verisi MFT kaydı içerisine sığacak boyutta ise MFT kaydı içerisine kaydedilir.
6. MFT dosyasının beşinci sırasında bulunan "root" klasörünün özelliklerinden dosyanın oluşturulacağı klasör tespit edilir. Klasörün MFT kaydına gidilerek buraya yeni oluşturulan dosya için indeks kaydı oluşturulur.
7. Dosya ile ilgili yapılan işlemlerin kayıtları $LogFile dosyasında arşivlenir.

Kısacası yeni bir dosya olduğunda disk üzerinde üç ana değişiklik olur. Bunlardan birincisi dosya ile ilgili MFT kaydı oluşturulur. İkincisi $BITMAP'te dosya içeriğinin kaydedildiği sektörler 1'e çevrilir. Üçüncüsü dosyanın oluşturulduğu klasörün MFT kaydı yeni dosya eklenerek güncellenir.

NTFS dosya sisteminde bir dosyanın silinmesi ise genel anlamda dosya oluşturma sürecinin geriye yürümesinden ibarettir. Aşağıda dosya silinme işleminin arka planda nasıl gerçekleştiği açıklanmıştır.

2.3.NTFS Dosya Sisteminde Bir Dosya Silmek

1. MFT dosyasında silinecek dosyanın kaydı tespit edilerek bu dosyanın "in-use" işareti 0'a çevrilir. Bu durumda silinen dosyanın MFT kaydı, üzerinde yeni bir dosya için oluşturulacak kayda kadar diskte muhafaza edilir.
2. Dosya içeriğinin kayıtlı olduğu sektörlere ait kullanım haritası $BITMAP dosyasından 0'a çevrilir. Bu durumda dosya içerikleri, kayıtlı oldukları sektörlere başka veriler yazılana kadar diskte muhafaza edilir. (Mahant ve Meshram, 2012, s.5)

3. Silinen dosyanın bulunduğu klasöre ait MFT kaydı güncellenerek silinen dosyanın klasör altındaki indeks kaydı çıkarılır. Diğer dosyaların indeks sıralaması yeniden oluşturulur.
4. Dosya sisteminde yapılan değişiklikler $LogFile içerisinde arşivlenir.

3. DOSYALAR VE DOSYA İMZALARI

Bilgi depolamak amaçlı olarak oluşturulan ve bilgisayar programları tarafından kullanılabilen, değiştirilebilen ve tekrar kaydedilebilen birimler dosya olarak nitelendirilir.

Veri kurtarma ilgi alanına giren dosyalar genellikle kullanıcılara ait, resim, video, ses, ofis, veritabanı vb. dosyalardır.

Disk üzerinde dosya yapıları incelendiğinde dosya içeriklerinin genellikle belirli bir hexadecimal değerle başladığı görülmektedir. Dosya içeriği başlangıcında bulunan spesifik değerler dosya imzası (file signature) veya dosya başlığı (file header) olarak isimlendirilir. Kullanılan bazı dosya formatları ve dosya imzaları Tablo 1.'de verilmiştir.

Her dosya formatı için dosya imzası olmayabilir. Örneğin QQ messenger veya ICQ 98 geçmiş dosyaları diskte "binary" formatta tutulur ve herhangi bir imza bilgisi bulunmaz. (Gubanov, 2012, s.11) Ayrıca bazı dosya formatları için başlık değeri ile birlikte ayrıca dosya içeriği sonunda "file footer" yani dosya içeriğinin sonlandığını belirten bir değer de bulunabilir.

Tablo 1. Kullanılan bazı dosya formatları ve dosya imzaları

Dosya Uzantısı	**Açıklama**	**Düz Metin**	**Hexadecimal**
jpg jpeg	JPEG raw, JFIF veya Exif dosya formatı		FF D8 FF DB
PDF	Adobe Portable Document Format	%PDF	25 50 44 46
mp3	MP3	ID3	49 44 33
Doc, xls, ppt, msg	Microsoft Office belgeleri		D0 CF 11 E0 A1 B1 1A E1
avi	Audio Video Interleave video format	RIFF.... AVI.	52 49 46 46 nn nn nn nn 41 56 49 20

4.SİLİNEN DOSYALAR

Son kullanıcılar için dijital ortamdaki "silinmiş dosya" bir dosyanın tamamen kaybolması ve yok olması anlamına gelir. Oysa NTFS dosya sisteminde kullanıcı tarafından silinen bir dosyaya ait veriyi disk üzerinden silinmez. Ancak bu verilerin bulunduğu alanları kullanılabilir olarak işaretlenerek daha sonra diske kaydedilecek dosyalar için müsait duruma getirilir. Bu hususla ilgili detaylı açıklamalar giriş bölümünde yapılmıştır. Şu halde veri kurtarma açısından silinmiş dosya veya daha geniş anlamda silinmiş veri kavramını detaylandırmak ve dört sınıfa ayırmak gerekir (Tablo 2).

1. Dosya sistemi tarafından silinmiş olarak işaretlenmiş ve dosya içeriği üzerine henüz veri yazılmamış dosyalar. Bu tür dosyalar hem üstbilgileri hem dosya içerikleri korunduğu için eksiksiz bir şekilde kurtarılabilir.
2. Dosya sistemi yani MFT kaydı yeni oluşturulacak kayıtlar için müsait olmasına rağmen henüz üzerine yazılmamış ancak dosya içeriğinin üzerine veri yazılmış olduğu durumlar. Burada dosya üstbilgilerine ulaşılabilir. Ancak üstbilgilerden yola çıkılarak kurtarılacak veri başka bir dosyaya aittir. Çünkü silinmiş dosyaya ait sektörler üzerine daha sonra farklı bir dosyaya ait veriler kaydedilmiştir.

3. Dosya sistemi kaydı silinmiş olmasına rağmen dosya içeriğinin üzerine henüz veri yazılmamış olduğu durumlardır. Bu tür dosyaların içeriklerine dosya sistemi kaydı kaybolduğu için doğrudan erişmek mümkün olmayacaktır. Çünkü dosya içeriğinin diskte hangi sektörlerde bulunduğu dosya sistemi tarafından tutulmaktadır. Bu nedenle bu tür dosyaların verileri diskte tarama yapılarak tespit edilebilir.
4. Hem dosya sistem kaydının hem dosya içeriğinin üzerine yazılmış olduğu durumlarda dosyaya ait herhangi bir verinin kurtarılması söz konusu değildir.

Tablo 2. Silinmiş dosya kavramları

#	MFT Kaydı	Dosya İçeriği	Açıklama
Birinci Tür Silinmiş Dosya	Mevcut	Mevcut	Dosya silinmeden önceki en son hali ile kurtarılabilir
İkinci Tür Silinmiş Dosya	Mevcut	Üzerine Yazılmış	Dosya içeriği kurtarılamaz. Ancak dosya üst bilgileri (isim vs.) kurtarılabilir.
Üçüncü Tür Silinmiş Dosya	Üzerinde Yazılmış	Mevcut	Dosya sistem kaydı silinmiş olduğu için dosya imzaları taratılarak kurtarılabilir.
Dördüncü Tür Silinmiş Dosya	Üzerinde Yazılmış	Üzerinde Yazılmış	Dosya üstbilgileri veya dosya içeriği kurtarılamaz.

5.VERİ KURTARMA

5.1.Birinci Yöntem: Dosya Sistemi Üzerinden Veri Kurtarma

Yukarıdaki yapılan açıklamalardan da anlaşılacağı üzere NTFS dosya sisteminde bir dosyanın mevcut veya silinmiş olması MFT kayıtlarındaki "in-use" işaretinin "0" veya "1" olmasından anlaşılabilmektedir. Dolayısıyla MFT dosyası üzerinde yapılacak bir analiz sonrasında MFT kayıtlarının ilgili bölümlerinde (22. ve 23. bayt) "in-use" işaretinin "0" olduğu kayıtların silinmiş dosya veya klasörlere ait olduğu ifade edilebilir. Veri kurtarma yazılımları basit bir anlatımla bu şekilde çalışmaktadır.

Dosya sistemi üzerinden veri kurtarma yapmanın önemli avantajları bulunmaktadır. Bunlardan birisi kullanıcılar açısından önemli olan her ne kadar dosya içerikleri olsa da, binlerce veya on binlerce dosyanın kurtarılması söz konusu olduğunda her dosyanın kendi klasöründe ve kendi ismiyle kurtarılması büyük önem arz etmektedir. Ayrıca silinmiş dosyalar arasına mevcut dosyalar karışmamaktadır. İşte dosya sistemi kayıtları ulaşılabilir olduğu için dosyalar bir anlamda nitelikleri ile birlikte kurtarılabilmektedir.

Dosya sistemi üzerinden veri kurtarma yapmanın bir diğer avantajı ise disk üzerinde parçalı olarak kaydedilmiş olan dosyaların kurtarılmasında problem yaşanmamasıdır. Yine bu kayıtlar dosya sisteminde tutulduğu için dosyaların eksik veya bozuk şekilde kurtarılması söz konusu olmayacaktır.

Dosya sisteminin analiz edilmesi ve bu şekilde silinmiş verilerin kurtarılması diskin tümünü taranmasına göre çok daha hızlı gerçekleştiğinden ciddi anlamda bir zaman tasarrufu sağlayacaktır. Ayrıca MFT kaydı mevcut olan ancak dosya içeriğinin üzerine yazılmış dosyaların kurtarılamaması ile ilgili olarak yöntemsel bir hatanın olmadığı bilinir.

Tüm avantajlara karşın dosya sistemi MFT kayıtlarına göre tahsis edilmeyen silinmiş dosyalar (Tabloda üçüncü sırada belirtilen tür) dosya sistemi taranarak kurtarılacak veriler arasında bulunmayacaktır.

5.2.İkinci Yöntem: Dosya İmzaları Üzerinden Veri Kurtarma

Özellikle kullanıcı dosyaları, ham formatları ile incelendiğinde dosya başlangıçlarında spesifik bazı değerlerin bulunduğu görülecektir. Bu değerlerin dosya imzası olduğu daha öncede belirtilmişti.

Veri kurtarma yöntemlerinden biri diskin tamamı taratılarak, kurtarılmak istenen dosya formatlarının imzaları aratılır. Tespit edilen değerden sonra –belirlenecek algoritmaya göre- ya bir sonraki dosya imzasına kadar yada belirli sayıdaki sektör sayısı kadar veri kopyalanır. (Pal, Sencar ve Memon, 2008, s. 1) Daha sonra bu veri jenerik bir isimle dosya haline getirilir.

Bu yöntemde mevcut veya silinmiş dosya ayrımı yapmak mümkün değildir. Ayrıca dosyalar, özellikle büyük boyutlu dosyalar diskte farklı alanlara parçalı olarak kaydedildiği için bu dosyaların tek parça halinde kurtarılması mümkün değildir. Hatta bazı durumlarda dosyaların ilk parçadan sonraki parçalarında tanımlayıcı imza bilgisi olmadığından parçalı dosyaların sadece ilk parçası kurtarılabilecektir.

Bu yöntemle kurtarılacak dosyalar jenerik isimlendirilir ve orijinal klasöründen bağımsız olarak kurtarılır. Az sayıda dosyalar için bu durum önemli olmasa bile nicelik olarak çok sayıdaki dosyalar için istenen dosyalara kolayca ulaşılabilmesi için problem oluşturabilir.

İlk yöntemden farklı olarak diskin tamamı yani tüm sektörleri taranacağı için işlemler daha uzun sürecektir.

Bununla birlikte MFT kaydı silinmiş ancak dosya içeriği üzerine yazılmamış dosyaların kurtarılabilmesi için bu yöntemin de uygulanması gerekmektedir.

Bu nedenle birinci yöntem uygulanarak kurtarılması istenen dosyalara ulaşmak mümkün olmamışsa ikinci yöntem yani dosya imzaları üzerinden tarama yapılarak veri kurtarma çalışması yapmak gerekmektedir.

6.ÖRNEK UYGULAMA

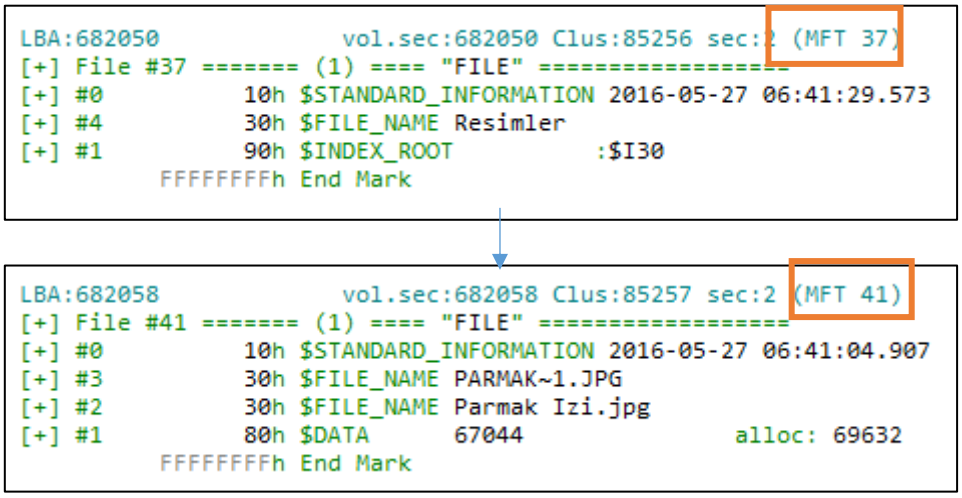
Çalışmamız için Windows 7 işletim sistemi yüklü standart bir bilgisayarın hard diskinden 1 GB bölüm ayrılarak NTFS dosya sistemi ile biçimlendirilmiştir. Tarafımızca “Test” olarak adlandırılan ve veri kurtarma yöntemlerin uygulanacağı bu alan biçimlendirme öncesinde “wipe” işlemine tabi tutularak tamamen temizlenmiştir. İlk olarak bölüm içerisinde “Resimler” isimli bir klasör oluşturulmuş ve klasör içerisine “bayrak.jpg”, “doga.jpg”, “Hard Disk.jpg” ve “Parmak Izi.jpg” isimli dört adet dosya oluşturulmuştur. MFT kayıtları incelendiğinde bir adet klasör ve dört adet dosya için 37 ile 41 arası toplamda beş adet MFT kaydı oluştuğu tespit edilmiştir.

Dosyalardan “doga.jpg”, “Hard Disk.jpg” silinmiş ve “Test” isimli disk bölümü Encase v.6.19 isimli adli bilişim yazılımı ile açıldığında bu dosyaların kurtarılabilir durumda oldukları tespit edilmiştir. MFT kayıtları incelendiğinde silinen dosyalara ait MFT kayıtlarda “in-use” işaretlerinin 0’a dönüştürüldüğü gözlemlenmiştir.

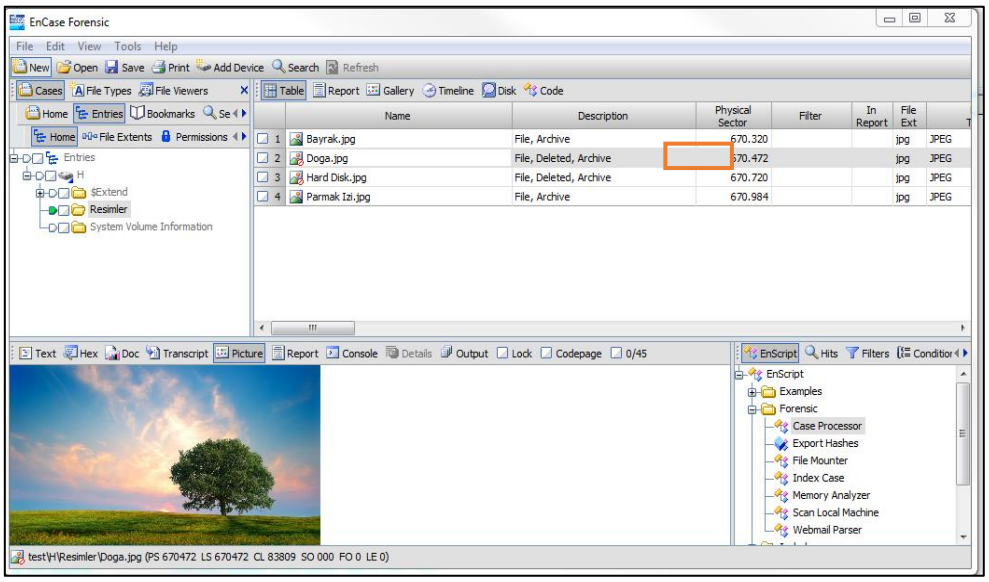
“Resimler” isimli klasör içerisine “Metin Belgesi.txt” isimli bir dosya oluşturulup MFT kayıtları tekrar incelendiğinde, daha öncesinde “Doga.jpg” isimli dosyaya ait olan 39 numaralı MFT kaydının “Metin Belgesi.txt” dosyasına tahsis edilmiş olduğu görülmüştür.

“Test” isimli bölüm, “Metin Belgesi.txt” dosyası oluşturulmadan önce ve oluşturulduktan sonra adli bilişim alanında yaygın olarak kullanılan temel yazılımlardan biri olan EnCase yazılımı ile açıldığında Şekil 3.’deki sonuçlarla karşılaşılmıştır.

Şekil-2. Oluşturulan Dosya ve Klasörlere ait MFT Kayıtları



Şekil 3. "Metin Belgesi.txt" dosyası oluşturulmadan önceki durum

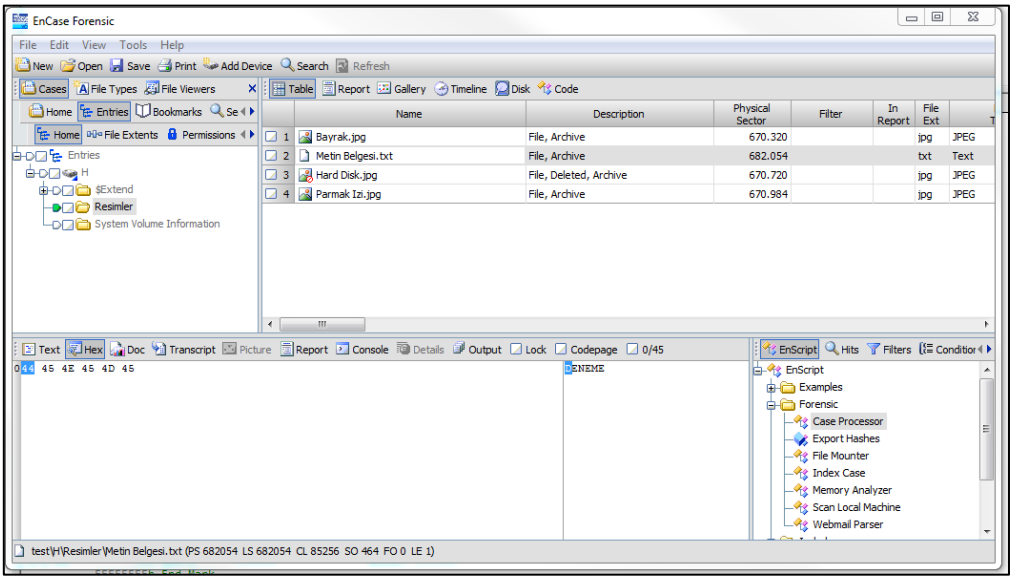


Adli bilişim yazılımları veya veri kurtarma yazılımları ilk olarak MFT dosyasını analiz ederek yani birinci yöntem olarak açıkladığımız sistemle dosya ve klasörleri tespit etmektedir. Burada silinen dosyanın MFT kaydı ilk müsait kayıt olduğu için dosya sistemi tarafından silinerek yeni dosyaya tahsis edilmiştir. Bunun neticesine beklediğimiz şekilde "Doga.jpg" dosyasının program tarafından tespit edilemediği görülmüştür. Şekil 3'ten anlaşılacağı üzere "Doga.jpg" dosya içeriğinin "670472." sektörde başladığı bilinmektedir. Söz konusu sektör kontrol edildiğinde resim dosyasına ait içerik bilgilerinin diskte kayıtlı olduğu yani dosya içeriği üzerine herhangi bir verinin yazılmamış olduğu anlaşılmıştır.

MFT kaydı yani üstbilgileri, üzerine yeni dosyalara ait verilerin yazılması sonucu tamamen silinen dosyaların kurtarılabilmesi ikinci yöntem olarak sunduğumuz dosya imzasının taratılması ile mümkündür.

Encase programında bu işlem "File Finder" menüsü ile yapılabilmektedir. "Test" isimli bölüm üzerinde "jpg" uzantılı dosyalar ile ilgili "File Finder" işlemi çalıştırıldığında tahsis edilmemiş alanlardan (Unallocated Clusters) bir adet resim dosyası tespit edilmiştir. Bu resim incelendiğinde silinen Doga.jpg isimli dosya olduğu tespit edilmiştir.

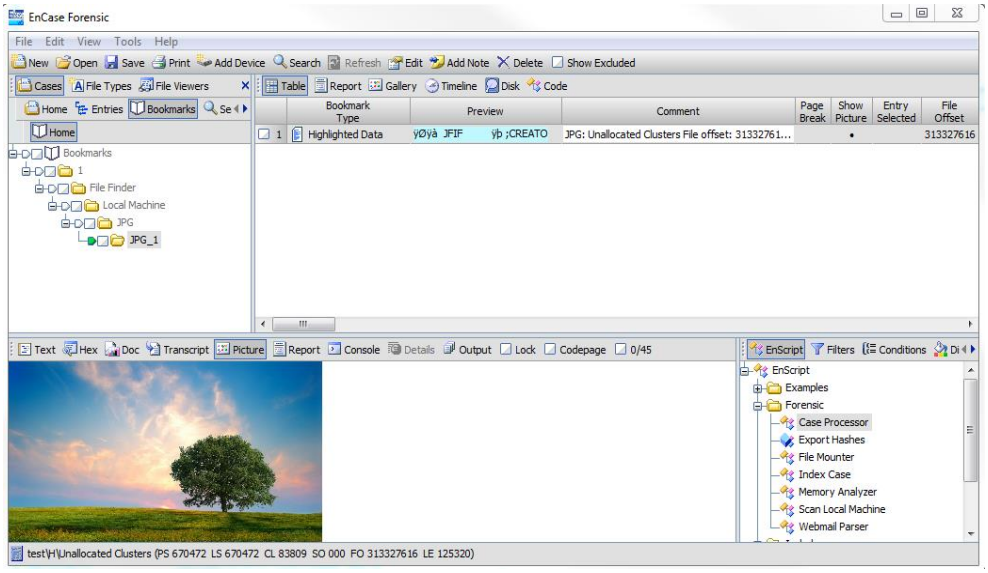
Şekil 4. "Metin Belgesi.txt" dosyası oluşturulmadan sonraki durum



Şekil 5. 670472. Sektörün bir bölümü

```
LBA:670472              vol.sec:670472 Clus:83809 sec:0
14761000: FF D8 FF E0 00 10 4A 46  49 46 00 01 01 00 00 01  ÿØÿà..JFIF......
14761010: 00 01 00 00 FF FE 00 3B  43 52 45 41 54 4F 52 3A  ....ÿş.;CREATOR:
14761020: 20 67 64 2D 6A 70 65 67  20 76 31 2E 30 20 28 75   gd-jpeg v1.0 (u
14761030: 73 69 6E 67 20 49 4A 47  20 4A 50 45 47 20 76 36  sing IJG JPEG v6
14761040: 32 29 2C 20 71 75 61 6C  69 74 79 20 3D 20 39 35  2), quality = 95
14761050: 0A FF DB 00 43 00 06 04  05 06 05 04 06 06 05 06  .ÿÛ.C...........
14761060: 07 07 06 08 0A 10 0A 0A  09 09 0A 14 0E 0F 0C 10  ........  ......
14761070: 17 14 18 18 17 14 16 16  1A 1D 25 1F 1A 1B 23 1C  ..........%...#.
14761080: 16 16 20 2C 20 23 26 27  29 2A 29 19 1F 2D 30 2D  .. , #&')*)..-0-
14761090: 28 30 25 28 29 28 FF DB  00 43 01 07 07 07 0A 08  (0%()(ÿÛ.C......
147610a0: 0A 13 0A 0A 13 28 1A 16  1A 28 28 28 28 28 28 28  .....(...(((((((
147610b0: 28 28 28 28 28 28 28 28  28 28 28 28 28 28 28 28  ((((((((((((((((
147610c0: 28 28 28 28 28 28 28 28  28 28 28 28 28 28 28 28  ((((((((((((((((
147610d0: 28 28 28 28 28 28 28 28  28 28 28 FF C0 00 11 08  (((((((((((ÿÀ...
```

Şekil 6. "File Finder" ile yani ikinci yöntem ile kurtarılan resim dosyası



7.SONUÇ

Çalışmamızda NTFS dosya sistemi üzerinden bir dosyanın oluşturulması ve silinmesi ile birlikte arka planda ne tür işlemlerin gerçekleştiği ve bu bağlamda dosya sistemi analiz edilerek nasıl veri kurtarılabileceğine dair açıklamalar yapılmıştır. Dosya sistemi üzerinden yapılan veri kurtarma çalışmalarına ek olarak dosya imzaları üzerinden veri kurtarma yöntemi ortaya konulmaya çalışılmıştır. Her iki yöntemin avantajları ve dezavantajları sıralanmıştır. Buna göre dosya sistemi analiz edilerek veri kurtarma çalışması yürütmek amaca yönelik olarak daha hızlı ve efektif gözükse de, disk üzerinde silinmiş olan tüm dosyaların kurtarılabilmesi için dosya imzaları üzerinden tarama yapılarak veri kurtarma yapmanın gerektiği anlaşılmıştır.

KAYNAKÇA

Cho, G.S. (2013). A computer forensic method for detecting timestamp forgery in NTFS, Computers & Security 34:36-46.

Fellows, G.F. (2005). The joys of complexity and the deleted file Digital Investigation. 2:89-93.

Carrier, B. (2005). File System Forensic Analysis, 2005 Addison Wesley Professional.

Pal, A. Sencar, H.T., Memon, N. (2008). Detecting file fragmentation point using sequential hypothesis testing digital investigation. 2-13.

Mahant, S.H., Meshram, B.B. (2012). NTFS Deleted Files Recovery: Forensics View, IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol. 2, No.3, June 2012.

Gubanov, Y. (2012). Retrieving Digital Evidence: Methods, Techniques and Issues ForensicFocus, July 2012

Beek, C. (2011). Introduction to File Carving, 2011, McAfee

Bui, S. Enyeart, M. Luong, J. (2003). Issues in Computer Forensics, COEN 150 Dr. Holliday May 22, 2003. http://ntfs.com/ntfs-system-files.htm (LSoft Technologies Inc.).

Medeiros, J. (2008). NTFS Forensics: A Programmers View of Raw Filesystem Data Extraction. 2008 Grayscale Research

Russon, R. Fledel, Y. (2016). NTFS Documentation. http://ftp.kolibrios.org/users/Asper/docs/NTFS/ntfsdoc.html.

Alazab, M., Venkatraman, S., Watters, P. (2009). Effective Digıial Forensic Analysis Of The Ntfs Disk Image, Special Issue on ICIT 2009 Conference - Applied Computing.