Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Ψηφιακή Εγκληματολογία – Ανάκτηση Δεδομένων στο Σύστημα Αρχείων NTFS** **Digital Forensics – Data Retrieval on the NTFS File System**
Ονοματεπώνυμο Φοιτητή	**Δημήτριος Κατσούλης**
Πατρώνυμο	**Νικόλαος**
Αριθμός Μητρώου	**ΜΠΠΛ/ 15027**
Επιβλέπων	**Κωνσταντίνος Πατσάκης, Επίκουρος Καθηγητής**

Ημερομηνία Παράδοσης **Ιούνιος 2018**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)	(υπογραφή)	(υπογραφή)
Κωνσταντίνος Πατσάκης Επίκουρος Καθηγητής	Χρήστος Δουληγέρης Καθηγητής	Παναγιώτης Κοτζανικολάου Επίκουρος Καθηγητής

Πίνακας Περιεχομένων

Κεφάλαιο 3°: Πρακτική Ανάκτηση Δεδομένων

Ευχαριστίες

Ευχαριστώ τον καθηγητή μου Επίκουρο Καθηγητή Κο Πατσάκη Κωνσταντίνο για την ευκαιρία που μου έδωσε να συνεργαστώ μαζί του στην παρούσα εργασία και όλη την βοήθεια που μου παρείχε για την επιτυχή ολοκλήρωσή της.

Αφιερώνω την Μεταπτυχιακή Διατριβή μου στον Βασιλαρά Αλέξανδρο. Φίλο, συνάδελφο και συνοδοιπόρο στο αέναο ταξίδι της αυτό-βελτίωσης, για την ώθηση που αυτός μου έδωσε, χρόνια πριν, ώστε να το ξεκινήσω και όλη την πολύτιμη καθοδήγηση που μου προσφέρει ανελλιπώς καθ' όλη τη διάρκειά του.

Περίληψη

Η ιστορική επαλήθευση του Νόμου του Moore, πενήντα τρία χρόνια μετά την αρχική του διατύπωση το 1965, έχει πλέον αδιαμφισβήτητα επιτευχθεί με τη σύγχρονη ψηφιακή τεχνολογία. Συσκευές με εξαιρετικά τεχνικά χαρακτηριστικά και επιδόσεις είναι πλέον οικονομικά προσιτές σε τέτοιο βαθμό ώστε να θεωρούνται άρρηκτα συνυφασμένες με τον σύγχρονο τρόπο ζωής. Οι δυνατότητες που προσφέρουν αυτές οι συσκευές αναρίθμητες και μεταξύ αυτών, αναπόφευκτη η χρήση τους και για εγκληματικούς σκοπούς. Ως αυτού ο τομέας της Ψηφιακής Εγκληματολογίας γεννήθηκε και εξελίσσεται ραγδαία τα τελευταία χρόνια με σκοπό, μεταξύ άλλων, τη διερεύνηση ψηφιακών μέσων, την ανάλυση ψηφιακών ιχνών, την ανακάλυψη πολύτιμων αποδεικτικών στοιχείων και την καθοριστική συμβολή στην καταπολέμηση της εγκληματικότητας και την επιβολή του Νόμου. Καθώς η τεχνολογία εξελίσσεται το ίδιο κάνουν παράλληλα τόσο οι κακόβουλες εγκληματικές μέθοδοι όσο και η επιστήμη της Ψηφιακής Εγκληματολογίας. Μία διαμάχη νομιμότητας έχει ξεκινήσει και στη βάση αυτής, απαραίτητος πυλώνας για την επιτυχή έκβασή της, βρίσκεται η γνώση. Γνώση τόσο των θεμελιωδών αρχών της Ψηφιακής Εγκληματολογίας όσο και του πολυποίκιλου αντικειμένου της. Οι θεμελιώδεις αρχές της είναι ρητά καθορισμένες και πρέπει να τηρούνται απαρέγκλιτα, ενώ το εύρος του αντικειμένου της είναι τεράστιο με πολλά διαφορετικά επίπεδα. Τα Συστήματα Αρχείων υπολογιστικών συστημάτων αποτελούν ένα από τα βαθύτερα αυτά επίπεδα που μπορούν να αναλυθούν επιστημονικά στον τομέα της Ψηφιακής Εγκληματολογίας. Κυρίαρχος του τομέα, ανάμεσα στα πολλά, το Σύστημα Αρχείων NTFS, ένα Σύστημα σύγχρονο, αποδοτικό και σταθερό με πολυσύνθετες δομές ασφάλειας και εφεδρείας. Χαρακτηριστικά γνωρίσματα που όχι μόνο το καθιέρωσαν ως ένα από τα πλέον εμπορικά και διαχρονικά Συστήματα Αρχείων αλλά που το κατέστησαν επίσης και το ευρύτερα χρησιμοποιούμενο σε παγκόσμια κλίμακα. Στοιχεία που αναντίρρητα προσδίδουν στο NTFS μία θέση βαρύνουσας σπουδαιότητας στον τομέα της Ψηφιακής Εγκληματολογίας και ως αυτού καθιστούν την ουσιαστική και βαθιά γνώση του επιτακτική για την τελική επικράτηση της ασφάλειας και της Νομιμότητας.

Abstract

Historical verification of Moore's Law, fifty three years after its initial wording in 1965, has indubitably been accomplished by modern digital technology. Devices with outstanding specifications and performance are now affordable to such an extent that they are considered indissolubly interlinked with modern era lifestyle. The capabilities these devices can offer are countless and amongst them, inevitable is their use in criminal activities. On account of this misuse the field of Digital Forensics was born and is rapidly evolving over the last years with goal, amongst others, the examination of digital media, the analysis of digital trails, the discovery of valuable evidence and the critical contribution to crime fighting and Law preservation. As technology is evolving so are malicious crime methods and the Digital Forensic Science. A legality conflict has begun and in the essence of this conflict, indispensable pillar for its successful outcome, there lays knowledge. Knowledge for both Digital Forensics' fundamental principles and its miscellaneous scope. The basic principles of Digital Forensics are explicitly defined and must be applied diligently, while its scope's magnitude is huge with many different layers. Computational Systems' File Systems comprise one of these deeper layers that can be analyzed scientifically in the field of Digital forensics. Salient File System on the field, among many, is the NTFS File System, a modern, efficient and robust System with complex safety and redundancy infrastructures. Distinctive features that not only did they establish NTFS as one of the most commercial and long lasting File Systems but in addition rendered it the most widely used worldwide. Aspects that undeniably transfuse NTFS a high importance role in the field of Digital Forensics and thus a substantive and deep knowledge of them is considered imperative for the final prevalence of security and Legality.

Εισαγωγή

Η τελευταία δεκαετία υπήρξε συγκλονιστική για την παγκόσμια κοινωνία επί της θεματικής των εν γένει πληροφοριακών δομών, της ψηφιακής ασφάλειας και της ιδιωτικότητας. Το 2009 ο στρατιωτικός αναλυτής πληροφοριών Bradley Manning κατάφερε να υποκλέψει 250.000 χιλιάδες απόρρητα αρχεία από την στρατιωτική βάση όπου υπηρετούσε στο Ιράκ, μεγάλος αριθμός εκ των οποίων απεικόνιζε για πρώτη φορά άοπλους πολίτες και άμαχο πληθυσμό να εκτελείται από αμερικάνους στρατιώτες. Ο τρόπος που το κατάφερε; «Παιδιάστικα εύκολος» όπως ο ίδιος αναφέρει σε συνομιλία του. Μέσω ενός απλού ψηφιακού δίσκου CD μουσικής γνωστής αμερικανίδας τραγουδίστριας. Σβήνοντας τη μουσική και αντιγράφοντας στο CD συμπιεσμένα αρχεία, σιγοτραγουδώντας τις επιτυχίες της καλλιτέχνιδας ενώ παράλληλα λάμβανε μέρος στη μεγαλύτερη διαρροή πληροφοριών στην ιστορία της Αμερικής μέχρι εκείνη τη στιγμή.

Τον Ιούνιο του 2013 ο Edward Snowden συντάραξε τον κόσμο με την δημοσίευση, μέσω της αγγλικής εφημερίδας «The Guardian», χιλιάδων διαβαθμισμένων αρχείων που ο ίδιος είχε υποκλέψει από την Εθνική Υπηρεσία Πληροφοριών (NSA) των Ηνωμένων Πολιτειών Αμερικής κατά την περίοδο στην οποία υπηρετούσε εκεί ως Ειδικός Πράκτορας. Τα υποκλαπέντα αρχεία, τα οποία υπολογίζονται σε περίπου 1.500.000 εκατομμύριο, περιείχαν σοκαριστικές πληροφορίες. Μεταξύ άλλων ανέφεραν τον τρόπο που η Αμερικάνικη κυβέρνηση παρακολουθούσε τις τηλεφωνικές συνομιλίες εκατομμυρίων αμερικάνων πολιτών, την τοποθέτηση κοριών παρακολούθησης στα γραφεία της Ευρωπαϊκής Ένωσης στη Νέα Υόρκη, την Ουάσινγκτον και τις Βρυξέλλες, την ψηφιακή διείσδυση σε στρατιωτικούς υπολογιστές-εξυπηρετητές στην Κίνα κ.α. «Αποκαλύψεις που είχαν βαθύ αντίκτυπο σε πολιτικό, βιομηχανικό, οικονομικό επίπεδο και πυροδότησαν δημόσιους διαλόγους σχετικά με την έννοια της παρακολούθησης, τους μηχανισμούς προστασίας και την επιβολή της ιδιωτικότητας.» [Patsakis C., Charemis A., Papageorgiou A., Mermigas D, Pirounias S., The market's response toward privacy and mass surveillance: The Snowden aftermath. Computers & Security Vol. 73: 194-206 (2018)].

Το πλέον πρόσφατο σκάνδαλο Facebook – Cambridge Analytica το 2018 υπολογίζεται ότι επηρέασε περίπου 87 εκατομμύρια πολίτες (Πηγή: ακρόαση Mark Zuckerberg στο Αμερικάνικο Κογκρέσο 11-04-2018) ανά τον κόσμο, με την εταιρεία κοινωνικής δικτύωσης να διαμοιράζει παρατύπως δεδομένα των χρηστών στην προαναφερόμενη συμβουλευτική εταιρεία με απώτερο σκοπό τον επηρεασμό και τη διαμόρφωση πεποίθησης για τις κομβικές προεδρικές εκλογές των Ηνωμένων Πολιτειών την προηγούμενη χρονιά.

Αναρίθμητα άλλα συμβάντα όπως η παγκοσμίως πρώτη διαδικτυακή "επιδημία" με τον ιό «Melissa» δεκαετίες πίσω, το έτος 1999, οι σύγχρονες τεχνικές μόλυνσης υπολογιστικών συστημάτων με κακόβουλο λογισμικό (malware infection) και οι διαμοιρασμένες επιθέσεις άρνησης υπηρεσίας (DDoS attacks), η δημιουργία του ιστοχώρου WikiLeaks από τον Julian Assange το 2006, η τεχνολογία του Internet of Things (IoT), ο νέος γενικός Ευρωπαϊκός κανονισμός για την προστασία των προσωπικών δεδομένων (GDRP May 2018) καθώς και η ανάπτυξη του Σκοτεινού Ιστού (Dark Web) και των κρυπτονομισμάτων (crypto currency) η συνύπαρξη και συνδυασμός των οποίων διευκολύνει την περαίωση εγκληματικών ενεργειών και την απόκρυψη των ιχνών τους, οδηγούν σε ένα αδιαμφισβήτητο συμπέρασμα. Διανύουμε την εποχή της πληροφορίας και ως ένα πρωταρχικό αγαθό και βασική προϋπόθεση ευημερίας κάθε σύγχρονης οργανωμένης κοινωνικής δομής η ασφάλεια της θα πρέπει να αποτελεί κύρια προτεραιότητα. Ένας από τους προστατευτικούς μανδύες που προσφέρει αυτή την ασφάλεια απέναντι στη κακόβουλη διαχείριση και εκμετάλλευση ψηφιακών δεδομένων είναι ο κλάδος της Ψηφιακής Εγκληματολογίας.

Η Ψηφιακή Εγκληματολογία αποτελεί έναν σχετικά νέο τομέα στην Επιστήμη της Εγκληματολογίας. Το 2003 το American Society of Crime Laboratory Directors–Laboratory Accreditation Board (ASCLD–LAB) αναγνώρισε τα ψηφιακά αποδεικτικά μέσα ως πλήρως

ολοκληρωμένο και αποδεκτό κομμάτι της Εγκληματολογικής Επιστήμης. Η επίλυση ενός μέρους ποινικώς κολάσιμων υποθέσεων καθίσταται ιδιαίτερα δυσχερής, ή συχνά ακατόρθωτη, χωρίς τη χρήση ψηφιακής τεχνοτροπίας και μεθοδολογίας στη διαδικασία της έρευνας. Στη βάση της διαδικασίας εξέτασης εντοπίζονται οι καθαυτές ψηφιακές συσκευές είτε ως προϊόντα εγκλήματος είτε ως μέσα διάπραξής του. Η πολυπληθέστερη κατηγορία που απαντάται είναι οι Ηλεκτρονικοί Υπολογιστές χωρίς να αποκλείονται και έτερες ψηφιακές συσκευές, όπως συσκευές κινητής τηλεφωνίας, ψηφιακές κάμερες και ρολόγια, συσκευές κίνησης και εντοπισμού θέσης κ.α.

Η διαδικασία εξέτασης μίας ψηφιακής συσκευής εντοπίζεται ακριβέστερα στη διαδικασία εξέτασης του αποθηκευτικού μέσου το οποίο αυτή χρησιμοποιεί για να αποθηκεύει τα ψηφιακά δεδομένα που διαχειρίζεται. Αυτά ποικίλουν από σκληρούς δίσκους (HDD), δίσκους συμπαγούς κατάστασης (SDD), μνήμες Flash (NOR, NAND) κλπ.. Κάποια από τα προαναφερθέντα ψηφιακά αποθηκευτικά μέσα μπορούν να φέρουν λειτουργικό σύστημα και κάποια όχι. Αυτό που όλα όμως έχουν ως κοινό χαρακτηριστικό είναι ότι οπωσδήποτε φέρουν κάποιο Σύστημα Αρχείων με το οποίο είναι διαμορφωμένα έτσι ώστε να επιτυγχάνουν τους σκοπούς παραγωγής τους. Η ψηφιακή ανάλυση για κάθε μέσο διαφέρει ανάλογα με το Σύστημα Αρχείων το οποίο αυτό φέρει.

Κάθε Λειτουργικό Σύστημα χρησιμοποιεί αντίστοιχα και ξεχωριστά Συστήματα Αρχείων για να περαιώνει επιτυχώς την αποστολή του. Το πιο ευρέως διαδεδομένο Λειτουργικό Σύστημα για Ηλεκτρονικούς Υπολογιστές στον κόσμο αυτή τη στιγμή αποτελεί το Λειτουργικό Σύστημα των Windows 10 της εταιρείας Microsoft με 36,22% επί του συνόλου (Πηγή: statcounter GlobalStats - Operating System Market Share Worldwide - April 2018). Το βασικό προεπιλεγμένο Σύστημα Αρχείων που χρησιμοποιούν τα Windows στις νεότερες εκδόσεις τους (ήδη από το 2000 με τα Windows ME) είναι το NTFS (New Technology File System). Η δημοτικότητα του το καθιστά ένα από τα πιο συχνά απαντούμενα Συστήματα Αρχείων και η πολυπλοκότητά του ένα από τα πιο δυσχερή κατά την εξέτασή του.

Η παρούσα εργασία αναλύει μία μόνο εκ των πτυχών του Συστήματος Αρχείων NTFS. Περιορίζεται στην ανάκτηση δεδομένων από αυτό σε συνάρτηση με τους κανόνες και τις αρχές της Ψηφιακής Εγκληματολογίας. Η εν λόγω πτυχή αποτελεί και την πλέον σημαντική στον τομέα καθώς η πλειοψηφία των υποθέσεων που απαιτούν ψηφιακή εγκληματολογική διερεύνηση αφορούν στοιχεία πληροφορίας που έχουν διαγραφεί ή τουλάχιστον έγινε προσπάθεια αυτό να επιτευχθεί από έναν κακόβουλο χρήστη. Δεδομένης της εξέλιξης του κλάδου ένας ικανός αριθμός εταιρειών προσφέρουν εμπορικά εργαλεία που τείνουν να αυτοματοποιούν και να απλοποιούν την εν λόγω διαδικασία. Στην παρούσα εργασία αυτά τα εργαλεία δεν θα αναφερθούν καθώς οι διάφορες διαδικασίες ανάκτησης που παρουσιάζονται έχουν πραγματοποιηθεί μόνο με χειροκίνητο τρόπο ο οποίος και θα αναλυθεί.

Η εργασία χωρίζεται σε τρία μέρη. Στο πρώτο θα αναλυθούν οι αρχές και οι έννοιες της Ψηφιακής Εγκληματολογίας. Το δεύτερο μέρος αποτελεί μια γενική παρουσίαση του Συστήματος Αρχείων NTFS, της βασικής του διάρθρωσης, των χαρακτηριστικών και της δομής του. Επικεντρώνεται και αναλύει πιο διεξοδικά τα στοιχεία εκείνα τα οποία θα βοηθήσουν τον αναγνώστη να κατανοήσει το τρίτο μέρος το οποίο αποτελεί και το πρακτικό κομμάτι της εργασίας όπου γίνεται χειροκίνητη ανάκτηση δεδομένων με τη βοήθεια hex editor και την τελική παρουσίαση των αποτελεσμάτων της ψηφιακής εξέτασης.

Κεφάλαιο 1°: Ψηφιακή Εγκληματολογία

1.1 Τι είναι η Ψηφιακή Εγκληματολογία

Η Ψηφιακή Εγκληματολογία (Digital Forensics) αποτελεί έναν συνδυασμό πολλών διαφορετικών επιστημών, διακρίνεται από ένα ευμετάβλητο, συνεχώς εξελισσόμενο αντικείμενο εξέτασης και διέπεται από κανόνες οι οποίοι μπορεί να διαφέρουν ανά γεωγραφική ή ακόμα και χρονική μονάδα. Ένεκα των ανωτέρω δεν υπάρχει ένας επίσημος ορισμός για αυτήν. Σε μια αφαιρετική προσέγγιση θα μπορούσε να περιγραφεί ως ο κλάδος της επιστήμης της Εγκληματολογίας ο οποίος ασχολείται με την ανάκτηση και έρευνα δεδομένων που εντοπίζονται σε ψηφιακές συσκευές. Αυτή η περιγραφή καίτοι δίνει ένα πολύ ξεκάθαρο στίγμα για το τι είναι η Ψηφιακή Εγκληματολογία δεν παύει να αποκρύπτει κάποιες από τις βασικές πτυχές της.

Ένας αναλυτικότερος και πλήρης ορισμός, λοιπόν, ο οποίος απεικονίζει αρτιότερα κάθε μία από τις πλευρές αυτού του πολυσχιδούς κλάδου είναι ο κάτωθι: «Ψηφιακή Εγκληματολογία είναι η εφαρμογή της Επιστήμης των Υπολογιστών και ερευνητικών διαδικασιών για ένα νομικό σκοπό που περιλαμβάνει την ανάλυση ψηφιακών πειστηρίων (πληροφορία αποδεικτικής αξίας που αποθηκεύεται ή μεταδίδεται σε ψηφιακή μορφή) κατόπιν κατάλληλης αρμοδιότητας έρευνας, αλυσίδας επιτήρησης των αποδεικτικών μέσων, επαλήθευσης με την χρήση Μαθηματικών, χρήσης επικυρωμένων εργαλείων, επαναληψιμότητας, γραπτής αναφοράς και πιθανής παρουσίαση από τον ειδικό». [Ken Zatyko, Scientific Working Group on Digital Evidence - https://www.swgde.org/]

Ο όρος Εγκληματολογία Υπολογιστών (Computer Forensics) που χρησιμοποιούταν παλαιότερα για να περιγράψει τον κλάδο έχει αντικατασταθεί προκειμένου να αποτυπώσει ορθότερα το πολύ πιο ευρύ φάσμα δράσης του, καθώς η ανάπτυξη της τεχνολογίας έχει εντάξει πλέον πλήθος διαφορετικών συσκευών στο πεδίο έρευνας και εξέτασης της Ψηφιακής Εγκληματολογίας.

Αναλόγως της συσκευής καθώς και του γενικότερου αντικειμένου της εξέτασης ο κλάδος της Ψηφιακής Εγκληματολογίας δύναται να χωριστεί σε:

- Εγκληματολογία Υπολογιστών (Computer Forensics)
- Εγκληματολογία Κινητών Συσκευών (Mobile Device Forensics)
- Εγκληματολογία Δικτύων (Network Forensics)
- Εγκληματολογία Βάσεων Δεδομένων (Data Base Forensics)
- Εγκληματολογική Ανάλυση Δεδομένων (Forensic Data Analysis)

Διάφορες μεθοδολογίες έχουν αναπτυχθεί με την πάροδο των ετών για το πεδίο εξέτασης εκάστου εκ των προαναφερθέντων αλλά παρά τις διαφορές και τις ιδιαιτερότητες που παρουσιάζει κάθε μία, στη βάση τους όλες διέπονται από τις ίδιες βασικές αρχές της Ψηφιακής Εγκληματολογίας.

1.2 Βασικές Αρχές της Ψηφιακής Εγκληματολογίας

1.2.1 Περιβάλλον Ψηφιακής Εγκληματολογικής Ανάλυσης

Η βάση της Ψηφιακής Εγκληματολογίας είναι αυτή καθαυτή η ψηφιακή έρευνα που λαμβάνει χώρα επί των δεδομένων που εξετάζονται. Η ψηφιακή έρευνα αποτελεί μια διαδικασία κατά την οποία ο εξεταστής αναπτύσσει συγκεκριμένα υποθετικά σενάρια, βασισμένα στην ύπαρξη ψηφιακών δεδομένων που ήδη έχουν ανευρεθεί στο υπό εξέταση μέσο, και στη συνέχεια

χρησιμοποιώντας επιστημονικές μεθόδους προσπαθεί να ανακαλύψει και έτερα δεδομένα που επιβεβαιώνουν ή καταρρίπτουν τα σενάρια αυτά.

Η εγκληματολογική ψηφιακή έρευνα εισαγάγει επιπλέον τον όρο των νομικών προϋποθέσεων που πρέπει να απορρέουν από την εξεταζόμενη υπόθεση. Οι θεωρίες που αναπτύσσονται και δοκιμάζονται πρέπει να έχουν νομικούς συσχετισμούς και το τελικό αποτέλεσμα της εξέτασης θα πρέπει να καταλήγει στις δικαστικές αίθουσες όπου θα απαντάει νομικά ερωτήματα σχετικά με συμβάντα που έλαβαν χώρα και άπτονται κάποιας σχετικής νομολογίας. Με άλλα λόγια, η εγκληματολογική ψηφιακή έρευνα αποτελεί μία πιο αυστηρά ορισμένη και δομημένη μορφή ψηφιακής έρευνας. Αυτή η επαυξημένης σημαντικότητας δομή της είναι ακριβώς ο λόγος που το περιβάλλον εντός του οποίου πραγματοποιείται μια εγκληματολογική ψηφιακή έρευνα πρέπει να είναι σαφώς καθορισμένο και να τηρείται μία σειρά προϋποθέσεων που διασφαλίζουν την εγκυρότητα αυτής στις δικαστικές αίθουσες.

Η εγκυρότητα μιας εγκληματολογικής ψηφιακής έρευνας διασφαλίζεται εφόσον αυτή πραγματοποιήθηκε εντός ενός εγκληματολογικά ασφαλούς περιβάλλοντος. Εγκληματολογικά ασφαλές περιβάλλον εξέτασης είναι το περιβάλλον το οποίο βρίσκεται απόλυτα υπό τον έλεγχο του εξεταστή σε κάθε στιγμή, και εντός του οποίου τίποτα δε συμβαίνει εκτός και αν ο εξεταστής το επιτρέπει ή το προκαλεί ο ίδιος. Και όταν ο εξεταστής προκαλέσει ή επιτρέψει ένα γεγονός να συμβεί, μπορεί με ένα μεγάλο βαθμό σιγουριάς να δηλώσει ποιο ακριβώς θα είναι το αποτέλεσμα που πρόκειται να συμβεί.

Λέγοντας όλα τα παραπάνω πρέπει να ξεκαθαριστεί η λανθασμένη αντίληψη ότι το περιβάλλον της εγκληματολογικής ψηφιακής έρευνας είναι το εγκληματολογικό εργαστήριο του εξεταστή όπου μία έρευνα λαμβάνει χώρα με ασφάλεια και πως η ίδια η εγκληματολογική ψηφιακή έρευνα είναι η χρήση εμπορικών εγκληματολογικών εργαλείων για την ανάκτηση δεδομένων.

Στην πραγματικότητα, η έννοια «περιβάλλον» είναι πολύ πιο περίπλοκη από τον αυστηρό γεωγραφικό χώρο ενός γραφείου όπως και η έννοια «έρευνα» είναι πολύ πιο περίπλοκη από την απλή χρήση εγκληματολογικών λογισμικών όπως θα δούμε και σε επόμενο κεφάλαιο όταν και θα αναλυθούν οι περιπτώσεις όπου η πρακτική παρεκκλίνει από τους συνήθεις κανόνες.

Συνεπώς η έννοια «περιβάλλον» επεκτείνεται σε κάθε τοποθεσία όπου μία ψηφιακή συσκευή δύναται να εντοπιστεί σε κάθε στιγμή, όπως το σπίτι του θύματος, μία σκηνή εγκλήματος, μία τοποθεσία επίσημης έρευνας μετά δικαστικού λειτουργού, οχήματα μεταφοράς, χώρος αποθήκευσης ψηφιακών πειστηρίων κ.α.

Ομοίως η έννοια «έρευνα» επεκτείνεται στη γενικότερη ροή εργασιών και περιλαμβάνει κάθε δυνατή διαδικασία που μπορεί να χρησιμοποιηθεί κατά την διενέργειά της, όπως για παράδειγμα η απευθείας εξέταση το ψηφιακού μέσου από τον εξεταστή (live analysis), η πραγματοποίηση συγκεκριμένων διεργασιών επί ενός ψηφιακού αντιγράφου, η μεταχείριση και εξειδικευμένη ανάλυση ανακτημένων δεδομένων κ.α..

1.2.2 Ψηφιακό Πειστήριο - Αποδεικτικό Στοιχείο

Άλλη μία θεμελιώδη έννοια της Ψηφιακής Εγκληματολογίας αποτελεί η έννοια «πειστήριο». Προσπαθώντας κανείς να αναλύσει την προαναφερόμενη έννοια θα βρεθεί μπροστά σε ένα κυκεώνα προβλημάτων που θα καταλήξουν να δημιουργήσουν ένα περίπλοκο και μη παραγωγικό αποτέλεσμα. Οι χιλιάδες νόμοι και κανονισμοί που αφορούν τη νομική αποδεικτική διαδικασία και οι διαφορετικές νομολογίες που απαντώνται ανάμεσα σε κράτη ή συχνά και μεταξύ περιοχών εντός του ίδιου κράτους καθιστούν την προσπάθεια ανάλυσης της έννοιας «πειστήριο» ιδιαίτερα δυσχερή.

Για να κάνει τα πράγματα ακόμα χειρότερα, η διεθνώς θεσμοθετημένη ορολογία χρησιμοποιεί τον αγγλικό όρο «evidence» για να περιγράψει όμως τη διττή φύση της έννοιας «πειστήριο» δημιουργώντας επιπλέον παρανοήσεις στην πλήρη κατανόησή της.

Χρησιμοποιώντας την ελληνική γλώσσα και την ποικιλομορφία που αυτή μας προσφέρει μπορούμε να αποτυπώσουμε με διαύγεια τον όρο «evidence» διακρίνοντας τα δύο συνθετικά του μέρη, ήτοι τους όρους «πειστήριο» και «αποδεικτικό στοιχείο».

Πειστήριο μπορεί να θεωρηθεί οτιδήποτε είναι δυνητικά ικανό να αποδείξει ή να καταρρίψει έναν ισχυρισμό ή μια διεκδίκηση, ανεξαρτήτως του τύπου του, της φυσικής του υπόστασης, της άμεσης ή έμμεσης φύσης του.

Τα πειστήρια μπορούν να κατηγοριοποιηθούν με πολλούς τρόπους σε υλικά ή άυλα, συμβατικά ή μη συμβατικά, ενοχοποιητικά ή αθωωτικά κλπ. Κύριο χαρακτηριστικό της έννοιας «πειστήριο» είναι η οιονεί φύση του. Ο «πειστήριος» ηλεκτρονικός υπολογιστής που κατασχέθηκε κατά τη διάρκεια έρευνας για το αδίκημα της κατοχής υλικού παιδικής πορνογραφίας από τον φερόμενο ως δράστη μπορεί μετά το πέρας της εξέτασης να μην παρουσιάσει κανένα εργαστηριακό εύρημα και ως αυτού κάθε άλλο παρά «πειστήριος» να θεωρηθεί τελικά.

Το πειστήριο που μετά την εξερεύνηση του θα αποτελέσει μέσο επιβεβαίωσης ή κατάρριψης μιας θεωρίας με νομικές προεκτάσεις αποκαλείται αποδεικτικό στοιχείο. Ορισμένα πειστήρια αποτελούν εμφανώς αποδεικτικά στοιχεία άνευ οιασδήποτε περαιτέρω έρευνας, όπως για παράδειγμα η βολίδα του όπλου στο σώμα του θύματος και άλλα απαιτούν επίπονη ερευνητική διεργασία προς αναζήτηση της ακριβούς φύσης τους, όπως για παράδειγμα ένα ψηφιακό υπολογιστικό μέσο.

Αναλύοντας, όμως, τους ανωτέρω όρους υπό το πρίσμα των βασικών αρχών της Ψηφιακής Εγκληματολογίας θα πρέπει να αναλύσουμε τις περισσότερο έμπρακτες προεκτάσεις τους και να ξεφύγουμε από τις βαθιές νομικές ορολογίες. Με αυτό το σκεπτικό θεωρώ απαραίτητο να τονιστούν τα ακόλουθα.

Τίποτα δεν μπορεί να θεωρηθεί ως πειστήριο/αποδεικτικό στοιχείο μέχρι τη στιγμή που η ύπαρξή του θα αναγνωριστεί επισήμως σε μία νομική διαδικασία και θα γίνει αποδεκτό ότι πληροί όλες τις νομικές προϋποθέσεις για να χαρακτηριστεί ως τέτοιο. Ακόμα και στην περίπτωση αυτή, ένα πειστήριο/αποδεικτικό στοιχείο δεν έχει καμία αξία εάν δε μπορεί να αντέξει μία σφοδρή επίθεση στη δικαστική αίθουσα από έναν ικανό συνήγορο – εξωτερικό πραγματογνώμονα.

Με πιο πρακτικούς όρους:

- Ο καλύτερος τρόπος για να αποτρέψει κανείς ένα πειστήριο να παραμείνει πειστήριο και ενδεχομένως να μετατραπεί σε αποδεικτικό στοιχείο είναι να το εμποδίσει εξ' αρχής να γίνει αποδεκτό στην ακροαματική διαδικασία. Εφεσιβάλλοντας λοιπόν την αρχική του αναγνώριση και αποδοχή από το δικαστήριο "εξαφανίζει" το πειστήριο από τη δίκη.
- Αποτυγχάνοντας κανείς να αποτρέψει την ολική διαγραφή του πειστηρίου από το δικαστήριο, η επόμενη ενέργειά του είναι να το υποβαθμίσει σε βαθμό που οι αμφιβολίες αρχίζουν να δημιουργούνται όχι μόνο για το συγκεκριμένο πειστήριο αλλά και για την υπόθεση γενικά.
- Τέλος αν ακόμα και η επίθεση εναντίον του ίδιου του πειστηρίου αποτύχει, η επόμενη σειρά ενεργειών αφορά την υπονόμευση οτιδήποτε άλλου περιβάλλει το πειστήριο, ήτοι τον ίδιο τον εξεταστή καθώς και την διαδικασία εξέτασης.

Συνοψίζοντας είναι εμφανές βάσει των ανωτέρω πόσο "εύθραυστη" είναι κάθε έρευνα και τον κομβικό ρόλο που παίζουν για την επιτυχή της έκβαση όλα τα μέρη της διαδικασίας της. Ένα εγκληματολογικά ασφαλές περιβάλλον έρευνας δημιουργεί τις ιδανικές συνθήκες για τον χειρισμό και την εξέταση πειστηρίων/αποδεικτικών στοιχείων που αποτελούν τον πυρήνα της κάθε υπόθεσης. Δεδομένης της ουσιώδους φύσης τους τα πειστήρια δέχονται ποικίλες επιθέσεις και συνεπώς ο εξεταστής χρειάζεται επιπλέον τρόπους θωράκισης τους. Ένας από

αυτούς είναι η επόμενη βασική Αρχή στην οποία θα αναφερθούμε, η Απόκτηση Αντιγράφου - Ψηφιακή Επικύρωση.

1.2.3 Απόκτηση Αντιγράφου και Ψηφιακή Επικύρωση

Απόκτηση Εγκληματολογικού Αντιγράφου Ασφαλείας

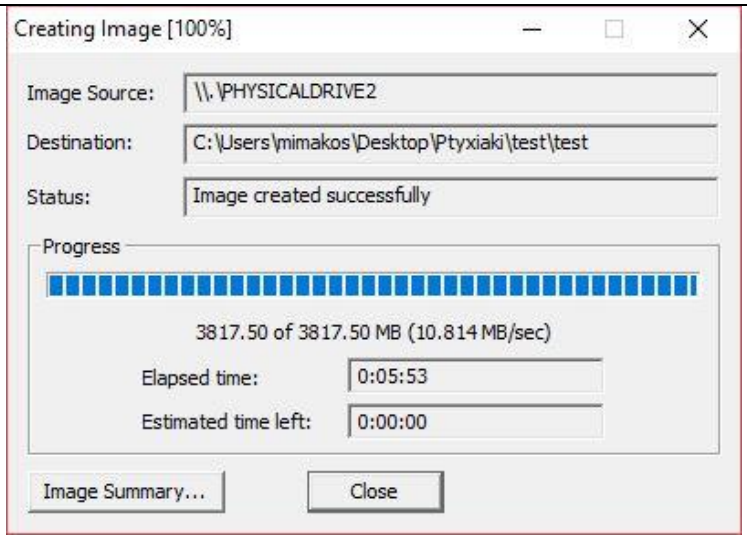
Ένας από τους βασικότερους κανόνες στην Ψηφιακή Εγκληματολογία είναι ότι, εκτός εξαιρετικών περιπτώσεων, ο εξεταστής δεν πρέπει ποτέ να διεξάγει μία εγκληματολογική ψηφιακή έρευνα χρησιμοποιώντας το αυθεντικό ψηφιακό μέσο. Αντιθέτως θα πρέπει πάντοτε να χρησιμοποιεί ένα εγκληματολογικό αντίγραφο ασφαλείας του (forensic backup - image).

Φυσικά θα υπάρξουν σπάνιες περιπτώσεις όπου ο εξεταστής θα είναι αναγκασμένος να διεξάγει έρευνα επί του αρχικού μέσου. Για παράδειγμα στην περίπτωση που τα χρονικά περιθώρια είναι τόσο ασφυκτικά όπου πρέπει να δράσει αμέσως ή σε εκείνες τις περιπτώσεις που η δημιουργία εγκληματολογικού αντιγράφου είναι αδύνατη λόγω λειτουργικού σφάλματος ή απαρχαιωμένου υλικού.

Μία ακόμα σημαντική παρατήρηση είναι ότι τα ψηφιακά πειστήρια αποτελούν απλά άλλον έναν από τους διάφορους τύπους πειστηρίων και επομένως απαιτούν τον ίδιο βαθμό ειδίκευσης και προσοχής στον χειρισμό τους, την ίδια προστασία απέναντι στην αλλοίωση τους, την ίδια αναλυτική περιγραφή για ασφαλή εντοπισμό τους και τα ίδια βήματα για την μακρά αποθήκευσή τους όπως όλα τα υπόλοιπα πειστήρια.

Συνεπώς μπορούμε να πούμε πως το στάδιο της απόκτησης ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας (Forensic Backup Acquire) αποτελεί ένα από τα σημαντικότερα στη διαδικασία της έρευνας καθώς κατά τη διάρκεια αυτού ο κίνδυνος αλλοίωσης του αρχικού πειστηρίου είναι ο μεγαλύτερος.

Η διαδικασία απόκτησης ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας περιλαμβάνει την αντιγραφή όλων των δεδομένων ενός ψηφιακού μέσου με έναν εγκληματολογικά ασφαλή τρόπο που αφήνει όλα τα αρχικά δεδομένα του μέσου άθικτα και απαράλλακτα. Όλα τα δεδομένα σημαίνει την ολότητα των περιεχομένων του μέσου, συμπεριλαμβανομένων όλων των δομών του Συστήματος Αρχείων, όλων των λογικών αρχείων, όλων των δεδομένων από τους διάφορους χώρους υπολείμματος (slack space), όλο το μη κατανεμημένο χώρο (unallocated space) και συνεπώς όλα τα διαγεγραμμένα αρχεία και όποιον τυχόν μη χρησιμοποιούμενο χώρο υπάρχει στο μέσο (unused space). Η απόκτηση όλων των ανωτέρω δεδομένων είναι που ξεχωρίζει την διαδικασία εγκληματολογικού αντιγράφου ασφαλείας η οποία απαιτεί ειδικό εγκληματολογικό λογισμικό με τις άπλες διαδικασίες αντιγράφων που παράγουν εμπορικά εργαλεία κατά τις οποίες αντιγράφονται μόνο τα λογικά αρχεία του μέσου.



Απόκτηση ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας από το εργαλείο FTK imager.

Υπάρχουν δύο είδη εγκληματολογικών αντιγράφων ασφαλείας (Forensic Backup) :

- το εγκληματικό αντίγραφο (Forensic copy) και
- τα εγκληματολογικά πειστήρια αρχεία (Forensic Evidence Files)

Το εγκληματικό αντίγραφο (Forensic copy) είναι ένα αντίγραφο του αρχικού μέσου που δημιουργείται σε ένα μέσο-στόχο ίδιας ή μεγαλύτερης χωρητικότητας, με τυχόν εναπομείναντα χώρο στο μέσο-στόχο να εγγράφεται με τη δεκαεξαδική τιμή 0x00 (κενό μέγεθος).

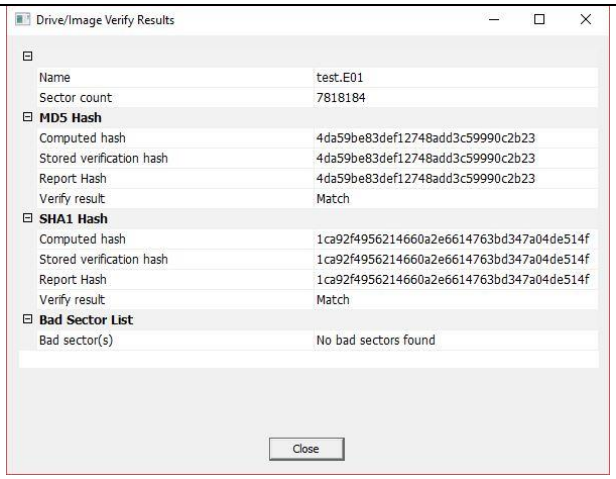
Τα εγκληματολογικά πειστήρια αρχεία (Forensic Evidence Files) είναι ένα ή περισσότερα αρχεία που περιέχουν ένα ακριβές bit προς bit αντίγραφο των δεδομένων που υπάρχουν στο αρχικό μέσο και συχνά αποκαλούνται εγκληματολογικές εικόνες (forensic images). Τα δεδομένα σε αυτά τα αρχεία αναλύονται είτε απευθείας από εγκληματολογικά λογισμικά (όπως μεταξύ άλλων το EnCase της Guidance Software ή το FTK της Access Data) ή μέσω επαναφοράς τους (Restore) σε έτερο μέσο δημιουργώντας κατά αυτόν τον τρόπο κάτι που μοιάζει με το εγκληματικό αντίγραφο (Forensic copy). Κάποια είδη εγκληματολογικών πειστήριων αρχείων προσφέρουν συμπίεση παράγοντας αρχεία πολύ μικρότερου μεγέθους σε σχέση με τα αρχικώς εξεταζόμενα μέσα, όπως επίσης προσθέτουν επιπλέον πληροφορία με τη μορφή επισυναπτόμενων επικεφαλίδων (headers) χωρίς φυσικά να αλλοιώνουν την αρχικά αποκτούμενη πληροφορία από το μέσο.

Έχοντας περαιώσει το στάδιο της απόκτησης ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας (Forensic Backup Acquire) ο εξεταστής έχει αποκτήσει έναν κλώνο του αρχικού μέσου για την περαιτέρω συνέχιση της έρευνας. Αυτό που μένει να γίνει τώρα είναι να διασφαλιστεί η ακεραιότητα του αρχικού πειστηρίου και με κάποιο τρόπο να αποδειχθεί ότι το αντίγραφό του είναι όντως ένας πιστός κλώνος που περιέχει ακριβώς την ίδια πληροφορία, γεγονός που θα επιτρέψει τη συνέχιση της έρευνας με ασφαλή τρόπο επί του αντιγράφου.

Ψηφιακή Επικύρωση – Αλγόριθμοι Hash

Η απόκτηση ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας (Forensic Backup Acquire) και η ψηφιακή επικύρωση του (Hashing) είναι αλληλένδετα σε τέτοιο βαθμό που καίτοι αποτελούν δύο διαφορετικά στάδια αποφάσισα να τα παρουσιάσω ως μία ενιαία θεμελιώδη Αρχή της Ψηφιακής Εγκληματολογίας.

Ψηφιακή επικύρωση δεδομένων ονομάζεται η διαδικασία επιβεβαίωσης ότι ένα αντίγραφο ενός αρχικού πειστηρίου είναι η ακριβής απεικόνιση του αρχικού πειστηρίου. Επιτυγχάνεται με τη χρήση μαθηματικών αλγορίθμων (Hashing algorithms) που εφαρμόζονται επί του αρχικού σετ δεδομένων, την παραγωγή μίας μοναδικής αλφαριθμητικής τιμής και τη σύγκριση αυτής με την τιμή-αποτέλεσμα που παρήγαγε ο ίδιος αλγόριθμος επί του αντιγράφου.



Ψηφιακή Επιβεβαίωση Αρχικού Μέσου – Εγκληματολογικού Αντιγράφου από το FTK imager.

Η διαδικασία επιβεβαίωσης θα πρέπει να εφαρμόζεται απαρέγκλιτα από τον εξεταστή καθώς είναι κομβικής σημασίας για την απόδειξη του έγκυρου συνδέσμου μεταξύ του αρχικού πειστηρίου και των ευρημάτων που τυχόν θα ανακαλύψει κατά την έρευνα του αντιγράφου. Έχοντας επικυρωθεί μαθηματικά η πιστότητα του χρησιμοποιούμενου αντιγράφου κανείς δε μπορεί να αμφισβητήσει τα αποτελέσματα της έρευνας και το γεγονός ότι αυτά τα ίδια ευρήματα θα μπορούσαν να ανευρεθούν στο αρχικό μέσο εάν αυτό αναλυόταν αντί του αντιγράφου.

Φυσικά η ψηφιακή επικύρωση έχει αμέτρητες άλλες χρήσεις στον τομέα της Ψηφιακής Εγκληματολογίας. Οι αλγόριθμοι hashing μπορούν να χρησιμοποιηθούν για την επικύρωση ενός μεμονωμένου αρχείου, ενός συνόλου αρχείων ή ακόμα και ενός string συμβολοσειράς.

Οι τιμές Hash διάφορων γνωστών αρχείων μπορούν να χρησιμοποιηθούν για να αποκλειστούν αυτά από την εξέταση, όπως αρχεία λειτουργικών συστημάτων και γνωστών εφαρμογών, μια διαδικασία γνωστή ως αρνητικό Hashing (Negative Hashing). Με αυτό τον τρόπο ο αριθμός των αρχείων προς εξέταση μπορεί να μειωθεί κατά πολύ αποκλείοντας αρχεία που ο εξεταστής μπορεί να παραβλέψει με ασφάλεια λόγω της εκ των προτέρων γνωστής αδιάφορης φύσης τους.

Αντίστοιχα οι τιμές Hash διάφορων γνωστών και αναζητούμενων αρχείων μπορούν να συγκεντρωθούν και να χρησιμοποιηθούν για την ανακάλυψη αρχείων ενδιαφέροντος για τη συγκεκριμένη υπόθεση, γνωστό ως θετικό Hashing (Positive Hashing).

Ολόκληρα σετ τιμών Hash μπορούν να δημιουργηθούν ώστε οι ενδιαφερόμενοι να δύνανται να ανταλλάσουν μόνο αυτά αντί των πραγματικών αρχείων. Αυτή η χρήση αποδεικνύεται ιδιαίτερα χρήσιμη στις περιπτώσεις εκείνες που η ανταλλαγή των πραγματικών αρχείων θα ήταν παράνομη ή και επικίνδυνη όπως για παράδειγμα στην ανταλλαγή αρχείων πορνογραφίας ανηλίκων μεταξύ Υπηρεσιών που μάχονται για την καταπολέμησή της ή αρχείων ιών και κακόβουλων λογισμικών για την αναβάθμιση βάσεων δεδομένων που τα περιγράφουν.

Υπάρχουν αρκετοί αλγόριθμοι ψηφιακής επιβεβαίωσης όπως οι CRC, MD5, SHA-1, SHA-256 και η ισχύς του καθενός εξαρτάται από την πολυπλοκότητα υλοποίησής του και τον αριθμό

των byte που χρησιμοποιεί για την έκφραση της τελικής αλφαριθμητικής ταυτότητας των δεδομένων που υπολόγισε.

Προβλήματα έχουν ανακύψει στη χρήση των αλγορίθμων ψηφιακής επιβεβαίωσης με την αποκάλυψη αδυναμιών σε αλγορίθμους που θεωρούνταν ασφαλείς πριν μερικά χρόνια όπως η εύρεση συγκρούσεων (collisions) από την ύπαρξη αρχείων με διαφορετικά ψηφιακά δεδομένα που παράγουν όμως την ίδια αλφαριθμητική ταυτότητα (Crypto2004 Conference, August 2004). Επίσης η αρχιτεκτονική των Δίσκων Συμπαγούς Κατάστασης (Solid State Disks) και τα ιδιαίτερα χαρακτηριστικά που υπάρχουν ενσωματωμένα στους ελεγκτές τους (controllers) όπως η λειτουργίες “garbage-collection” και “wear-leveling” καθιστούν τη διαδικασία της ψηφιακής επικύρωσης άχρηστη καθώς η ψηφιακή κατάσταση των δίσκων δύναται να αλλάζει κάθε στιγμή μεταβάλλοντας και την αλφαριθμητικής τους ταυτότητα.

Παρά τα όποια προβλήματα η διαδικασία της ψηφιακής επικύρωσης αποτελεί απαραίτητο στάδιο κατά τη διάρκεια μιας ψηφιακής εγκληματολογικής έρευνας και εξασφαλίζει την αρτιότητα του αποτελέσματος. Είναι άρρηκτα συνυφασμένη με τη διαδικασία απόκτησης ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας, την οποία ακολουθεί χρονικά. Πως όμως αποκτήθηκε το εγκληματολογικό αντίγραφο εξ’ αρχής προτού επιβεβαιωθεί ψηφιακά για την αρτιότητά του;

Νωρίτερα χρησιμοποιήθηκε η φράση «…αντιγραφή όλων των δεδομένων ενός ψηφιακού μέσου με έναν εγκληματολογικά ασφαλή τρόπο…» για να περιγραφεί η διαδικασία απόκτησης ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας. Ποιος ακριβώς όμως είναι αυτός ο ασφαλής τρόπος;

1.2.4 Φραγμός Εγγραφής

Η ψηφιακή εγκληματολογική ανάλυση ξεκινάει πολύ πριν ο ερευνητής αρχίσει να εξετάζει το μέσο στο εργαστήριο. Στην πραγματικότητα όπως αναφέρθηκε και προηγουμένως η διαδικασία της έρευνας ξεκινάει τη στιγμή που αυτή διατάσσεται, και συνεχίζει σε όλη τη φάση της συλλογής πειστηρίων, στο πρακτικό στάδιο της φυσικής εξέτασης μέχρι και στη μακρόχρονη αποθήκευση των πειστηρίων σύμφωνα με όλες τις νομικές διαδικασίες.

Συνεπώς ένας εξεταστής πρέπει πάντα να προσέχει τη σωστή διαχείριση, προστασία, καταγραφή και διατήρηση του αρχικού ψηφιακού μέσου. Όπως σημειώθηκε προηγουμένως, το στάδιο κατά το οποίο το πειστήριο βρίσκεται στο μεγαλύτερο κίνδυνο είναι κατά τη φυσική του διαχείριση στο εργαστήριο και κατά τη διαδικασία απόκτησης ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας.

Για να προστατέψει το μέσο ο εξεταστής πρέπει να λάβει ενεργά μέτρα για την αποτροπή οποιασδήποτε αλλαγής στα αρχικά δεδομένα του. Αυτό μπορεί να επιτευχθεί με τη χρήση τεχνικών φραγμού εγγραφής (Write Blocking Techniques) οι οποίες μπορούν να υλοποιηθούν με τρείς διαφορετικούς τρόπους:

- Μέσω φυσικών τροποποιήσεων του υλικού (hardware) που είναι σχεδιασμένες να μπλοκάρουν οποιαδήποτε εγγραφή στο μέσο.
- Με τη χρήση λογισμικού (software) που είναι σχεδιασμένο να αποτρέπει ή να αναχαιτίζει οποιαδήποτε εγγραφή στο μέσο.
- Μέσω υλικολογισμικού (firmware) το οποίο ελέγχει μία φυσική συσκευή και ενεργοποιεί την προστασία εγγραφής σε αυτή.

Οι τρείς διαφορετικές τεχνοτροπίες φραγμού εγγραφής πολύ συχνά μπορεί να προκαλούν σύγχυση οπότε η κατανόηση του τρόπου με τον οποίο λαμβάνουν χώρα οι εγγραφές σε ένα ψηφιακό μέσο είναι επιτακτική.

Αρχικά τα πρώτα Λειτουργικά Συστήματα (IBM PC DOS και Microsoft MS DOS) παρείχαν μία πολύ απλή και σταθερή διαδικασία εγγραφής. Οι περισσότερες εγγραφές στο μέσο συνέβαιναν μέσω κλήσεων στις διακοπές BIOS (BIOS Interrupt 13 ή Interrupt 13x). Μία διακοπή

BIOS είναι απλά μία κλήση σε μία προγραμματιστική ρουτίνα του συστήματος BIOS (Basic Input Output System). Έτσι οι περισσότερες εφαρμογές που ήθελαν να γράψουν δεδομένα στο ψηφιακό αποθηκευτικό μέσο, πραγματοποιούσαν μία κλήση στο Λειτουργικό Σύστημα που με τη σειρά του πραγματοποιούσε κλήση στις διακοπές BIOS για να κάνουν την εγγραφή. Οι διακοπές BIOS περιλάμβαναν κώδικα που περιείχε εντολές ATA για να χειριστούν τους ελεγκτές (Controllers) των δίσκων. Το ανωτέρω ονομάζεται Παραδοσιακό Μοντέλο Πρόσβασης Δίσκων.

Επιπλέον του Παραδοσιακού υπήρχε το Μοντέλο Άμεσης Πρόσβασης Δίσκων κατά το οποίο ορισμένες εφαρμογές μπορούσαν να παρακάμψουν τελείως το Λειτουργικό Σύστημα και να κάνουν απευθείας κλήσεις στις διακοπές BIOS για να κάνουν την εγγραφή. Η διαδικασία αυτή θεωρείτο ριψοκίνδυνη καθώς τα προγράμματα επιχειρούσαν εγγραφές αδιαφορώντας για έτερες διεργασίες που ίσως χρησιμοποιούσαν ταυτόχρονα τους ίδιους πόρους συστήματος.

Ο ερχομός του Λειτουργικού Συστήματος Windows άλλαξε τις διαδικασίες εγγραφής. Οι εφαρμογές τώρα πια απαγορεύεται να επικοινωνούν με το σύστημα BIOS καθώς αυτό αναγνωρίστηκε ως έλλειμμα ασφαλείας. Τα Windows πλέον απαιτούν κάθε αίτηση για πρόσβαση στο δίσκο και στο υλικό να λαμβάνει χώρα μέσω του Windows API (Application Programmer's Interface), το οποίο αποτελεί μία συλλογή προγραμματιστικών ρουτινών και συναρτήσεων που είναι ενσωματωμένη στο Λειτουργικό Σύστημα. Με αυτόν τον τρόπο επιτυγχάνεται σταθερότητα και αξιοπιστία.

Έχοντας ξεκαθαρίσει τα προαναφερθέντα είναι πλέον πολύ πιο εύκολο να αναλύσουμε τις διαφορετικές τεχνοτροπίες φραγμού εγγραφής που απαντώνται στον τομέα της Ψηφιακής Εγκληματολογίας.

Φραγμός Εγγραφής Υλικού (Hardware Write Blocking)

Η πρώτη μέθοδος προστασίας των μέσων επιτεύχθηκε μέσω του υλικού. Τροποποιήσεις επί του υλικού ή οι δυνατότητες του ίδιου του υλικού αποτρέπουν τις εγγραφές στο μέσο. Μερικά παραδείγματα είναι:

- Το διάφραγμα προστασίας εγγραφής στις παλιές δισκέτες 3.5 ιντσών
- Το πώμα προστασίας εγγραφής που φέρουν ορισμένα αφαιρούμενα αποθηκευτικά μέσα διασύνδεσης USB (USB Thumbdrives).
- Οι ψηφιακοί δίσκοι CD-R και DVD-R.

Συσκευή φραγμού εγγραφής υλικού.

Αν και ο φραγμός εγγραφής υλικού είναι αρκετά απλή και αποτελεσματική μέθοδος, δεν μπορεί να προσφέρει λύση στη πλειοψηφία των περιπτώσεων που απασχολούν ένα εξεταστή. Δεν υπάρχει κανένας τρόπος να εφαρμοστεί σε έναν σκληρό δίσκο καθώς δεν είναι μία ενσωματωμένη τεχνολογία επί αυτού.

Από τα πρώιμα κιόλας στάδια της εξέλιξης της Ψηφιακής Εγκληματολογίας έγινε εμφανής η ανάγκη εύρεσης νέων τεχνικών πέρα όσων παρείχαν τα ίδια τα ψηφιακά μέσα και έτσι αναπτύχθηκαν οι τεχνικές Φραγμού Εγγραφής Λογισμικού.

Φραγμός Εγγραφής Λογισμικού (Software Write Blocking)

Ο φραγμός έγγραφής λογισμικού εξελίχθηκε με το Λειτουργικό Σύστημα DOS. Αρχικά υλοποιήθηκε με απλές τροποποιήσεις σε αρχεία κλειδιά του Λειτουργικού Συστήματος ώστε το τελευταίο να αποτραπεί από το να εκτελεί εγγραφές στο μέσο.

Δεδομένης της διαδικασίας που χρησιμοποιούταν για τις εγγραφές, με τη χρήση των διακοπών BIOS Int13 και Int13x από το Λειτουργικό Σύστημα, γράφτηκαν προγράμματα τα οποία διέκοπταν αυτές τις κλήσεις και κατ' επέκταση απέτρεπαν τις εφαρμογές από το να προκαλούν εγγραφές στα μέσα. Φορτώνοντας ένα τέτοιο πρόγραμμα στη μνήμη ο εξεταστής μπορούσε να συνεχίσει τη διαδικασία απόκτησης ψηφιακού εγκληματολογικού αντιγράφου ασφαλείας από το αρχικό πειστήριο, εκτελώντας διαφορετικά εγκληματολογικά λογισμικά για αυτή τη χρήση, χωρίς να ενέχει ο κίνδυνος της εγγραφής δεδομένων στο αυθεντικό μέσο και επομένως της αλλοίωσης του. Παρόλο αυτά ο συγκεκριμένος τρόπος λειτουργούσε μόνο για τα προγράμματα που χρησιμοποιούσαν το Παραδοσιακό Μοντέλο Πρόσβασης Δίσκων.

Καθώς η ανάπτυξη λογισμικού αναπτυσσόταν άρχισαν να εμφανίζονται προγράμματα που χρησιμοποιούσαν το Μοντέλο Άμεσης Πρόσβασης Δίσκων που οδήγησε σε αποτυχίες κατά τη διαδικασία φραγμού εγγραφής καθώς τα λογισμικά δεν μπορούσαν να αποτρέψουν τέτοιου είδους εγγραφές που παρέκαμπταν το Λειτουργικό Σύστημα.

Τελικά νέα λογισμικά αναπτύχθηκαν που συμβάδισαν με τις εξελίξεις και κατάφεραν να αποτρέπουν εγγραφές για οποιοδήποτε από τα δύο μοντέλα πρόσβασης δίσκων του Λειτουργικού Συστήματος DOS αλλά και πάλι καταστάθηκαν απολύτως παρωχημένα με την έλευση του Λειτουργικού Συστήματος Windows. Το περιβάλλον των Windows (και ομοίως των λοιπών σύγχρονων Λειτουργικών Συστημάτων) είναι πολύ πιο περίπλοκο και δύσκολο στον έλεγχο του. Όπως σημειώθηκε ήδη, δεν υπάρχουν απευθείας κλήσεις στη BIOS ή στο υλικό μέσω των Windows οπότε όλες οι λύσεις φραγμών έγγραφής λογισμικού ήταν πλέον αναποτελεσματικές και δε μπορούσαν να προστατέψουν το μέσο από τροποποιήσεις και αλλοιώσεις. Μία νέα μέθοδος φραγμού εγγραφής έπρεπε να αναπτυχθεί.

Φραγμός Εγγραφής Υλικολογισμικού (Firmware Write Blocking)

Όπως το ίδιο το όνομα υπαινίσσεται ο φραγμός εγγραφής υλικολογισμικού δεν εξαρτάται από το Λειτουργικό σύστημα, το σύστημα BIOS του υπολογιστή ή από προγράμματα που εκτελούνται στη μνήμη για να αποτρέπουν εγγραφές. Αντιθέτως ο μηχανισμός προστασίας είναι το υλικολογισμικό που είναι ενσωματωμένο επί μίας συγκεκριμένης συσκευής.

Οι συσκευές φραγμού εγγραφής υλικολογισμικού (write blockers) που παράγονται από την Guidance Software, LogiCube και πολλές άλλες είναι παραδείγματα τέτοιων συσκευών.

Συσκευή φραγμού εγγραφής υλικολογισμικού της εταιρείας TABLEAU.

Ο φραγμός εγγραφής υλικολογισμικού είναι αποτελεσματικός σε κάθε περιβάλλον και έτσι η πιθανότητα αποτυχίας είναι συγκριτικά πολύ μικρότερη σε σχέση με φραγμούς εγγραφής λογισμικού. Για αυτόν το λόγο είναι με διαφορά ο πιο δημοφιλής τρόπος προστασίας εγγραφής.

Οι συσκευές φραγμού εγγραφής υλικολογισμικού λειτουργούν σαν ένα φυσικό φράγμα μεταξύ του υπολογιστή/συσκευή δημιουργίας εγκληματολογικού αντιγράφου και του πειστήριου μέσου που κλωνοποιείται. Με όποιο τρόπο και αν επιχειρείται η εγγραφή, μέσω του Λειτουργικού Συστήματος και των διακοπών BIOS, απευθείας μέσω της BIOS, απευθείας στο δίσκο μέσο εντολών ATA ή μέσω του Windows API, η συσκευή δεν επιτρέπει την κλήση στον ελεγκτή του μέσου κι επομένως η εντολή εγγραφής αποτυγχάνει διατηρώντας το αρχικό πειστήριο αναλλοίωτο.

Φυσικά όπως και κάθε άλλο τεχνολογικό δημιούργημα οι συσκευές φραγμού εγγραφής υλικολογισμικού εμφανίζουν διάφορες αδυναμίες. Αρχικά κάποιες παλαιότερες συσκευές δεν μπορούν να υποστηρίξουν σύγχρονους δίσκους μεγάλης χωρητικότητας. Έπειτα, το Λειτουργικό Σύστημα πρέπει να μπορεί να αναγνωρίζει και να υποστηρίζει τη συσκευή φραγμού όπως και κάθε άλλη συσκευή η οποία συνδέεται σε αυτό, κάτι που δεν είναι πάντα εφικτό λόγω ασυμβατότητας τεχνολογιών. Τέλος ενδέχεται να υπάρχουν διάφορα προβλήματα ταχύτητας και συμβατότητας καλωδίων ειδικά με παλαιότερα εξεταζόμενα μέσα.

Οι κατασκευαστές αυτών των συσκευών εντοπίζουν τα προβλήματα και εκδίδουν αναβαθμίσεις των υλικολογισμικών τους για να τα διορθώνουν. Γεγονός όμως που ενέχει μεγάλους κινδύνους για τους εξεταστές ψηφιακών πειστηρίων.

Όπως ολόκληρος ο τομέας της πληροφορικής είναι διαρκώς εξελισσόμενος έτσι και η Ψηφιακή Εγκληματολογία βρίσκεται σε μια διαρκή διαδικασία ανάπτυξης. Και καθώς νέες τεχνολογίες αναπτύσσονται οι εταιρείες που δραστηριοποιούνται στο χώρο αναπτύσσονται μαζί τους.

Αναβάθμιση μίας συσκευής φραγμού εγγραφής μπορεί να προκύψει ως ανάγκη διόρθωσης κάποιων σφαλμάτων που έχουν εντοπιστεί στη λειτουργία του ή για την ενσωμάτωση κάποιας νέας τεχνολογίας. Ο εκάστοτε εξεταστής έχει καθήκον να είναι ενήμερος για κάθε τέτοια αλλαγή διότι η χρήση της συσκευής μπορεί να προκαλέσει αλλοίωση ενός πειστηρίου, είτε γιατί δεν έχει μία απαραίτητη ενημέρωση είτε γιατί έχει μία νέα η οποία όμως δε λειτουργεί όπως θα έπρεπε.

Αυτό μας επαναφέρει και πάλι στην πρώτη βασική αρχή της Ψηφιακής Εγκληματολογίας δίνοντας έμφαση στο γεγονός πως ολόκληρη η διαδικασία της ψηφιακής εγκληματολογικής έρευνας θα πρέπει να λαμβάνει χώρα σε ένα ασφαλές, αποστειρωμένο περιβάλλον όπου ο εξεταστής έχει πάντοτε τον πλήρη έλεγχο.

Παρατηρούμε λοιπόν πως όλες οι ανωτέρω προαναφερθείσες αρχές αποτελούν ουσιαστικά μία αλληλένδετη έννοια, η απαρέγκλιτη τήρηση των οποίων είναι απαραίτητη για τη διασφάλιση

της ποιοτικής ανάλυσης, της νομικής αποδοχής και της τελικής επιτυχής ολοκλήρωσης μίας ψηφιακής εγκληματολογικής έρευνας.

Για κάθε κανόνα όμως υπάρχει και μία εξαίρεση.

1.2.5 Αποκλίσεις από τις Συμβατικές Τακτικές

Συνδυάζοντας όλα τα ανωτέρω στάδια σε μία ομαλή διαδικασία εγκληματολογικής έρευνας είναι η ιδανική κατάσταση που εγγυάται νομική αποδοχή των αποτελεσμάτων της στις δικαστικές αίθουσες. Αυτό δυστυχώς δεν είναι πάντοτε εφικτό.

Υπάρχουν στιγμές που ακόμα και στην ηρεμία του εγκληματολογικού εργαστηρίου ο εξεταστής θα χρειαστεί να παρεκκλίνει από τις συμβατικές διαδικασίες της Ψηφιακής Εγκληματολογίας. Και σίγουρα θα υπάρξουν στιγμές έκτος εργαστηρίου όπου στην επιτόπια έρευνα ο εξεταστής θα χρειαστεί να αποκλίνει παρασάγγας από κάθε συνήθης τακτική.

Γενικά ένα τέτοιο γεγονός είναι αναμενόμενο λόγω της ιδιαίτερης φύσης της εξέτασης και ως ένα βαθμό αποδεκτό, εφόσον τηρούνται κάποιοι βασικοί κανόνες. Αυτοί οι κανόνες ως στόχο έχουν τη διατήρηση της αξιοπιστίας της έρευνας και των αποτελεσμάτων της και είναι οι ακόλουθοι:

- Τα γεγονότα και οι ιδιαίτερες περιστάσεις καταγράφονται πλήρως και αναφέρονται στον ιεραρχικά προϊστάμενο της έρευνας.
- Οι διαδικασίες που χρησιμοποιούνται είναι λογικές και δικαιολογούνται πλήρως από τις ιδιαίτερες περιστάσεις.
- Οι διαδικασίες που χρησιμοποιούνται είναι όσο το δυνατόν πιο διακριτικές γίνεται υπό τις ιδιαίτερες περιστάσεις.
- Το κέρδος από τη χρήση μη συνήθων τεχνικών υπερέχει των κινδύνων για καταστροφή δεδομένων.

Ιδού μερικά παραδείγματα διαδικασιών που αποκλίνουν από τις συνήθεις τεχνικες:

- Η δημιουργία ενός εγκληματολογικού αντιγράφου δεδομένων στο πεδίο της επιτόπιας έρευνας από ένα υπολογιστή σε λειτουργία.
- Η εγκληματολογική ανάκτηση (capture) της μνήμης RAM από ένα υπολογιστή σε λειτουργία.
- Η χειροκίνητη επιθεώρηση μίας φορητής συσκευής και η αλλαγή των ρυθμίσεών της.
- Η δημιουργία ενός λογικού αντιγράφου (logical backup) δεδομένων από πειστήρια μέσα.
- Η απευθείας έρευνα ψηφιακών συσκευών και μέσων άμεσα στο σημείο.

Κάποια από τα ανωτέρω λαμβάνουν χώρα τόσο συχνά που μπορούν να θεωρηθούν ως συμβατικές πρακτικές στον χώρο των ειδικών της Ψηφιακής Εγκληματολογίας. Για παράδειγμα η εγκληματολογική ανάκτηση της μνήμης RAM (Memory Dump) από ένα υπολογιστή σε λειτουργία θεωρείται πλέον ότι αποτελεί ή θα έπρεπε να αποτελεί ρουτίνα στη διαδικασία της ανάλυσης καθώς πλέον τα υπολογιστικά συστήματα φέρουν τόσο μεγάλα μεγέθη μνήμης που η απώλεια των πληροφοριών τους κρίνεται μη αποδεκτή.

Ομοίως η διαδικασία επιλογής (triage) των απολύτως αναγκαίων υπολογιστικών συστημάτων που πρέπει να κατασχεθούν σε μία έρευνα θεωρείται ευρέως μία λογική και διακριτική τεχνική. Αν σκεφτεί κανείς τα μεγέθη των αποθηκευτικών μέσων που φέρουν πλέον οι σύγχρονες ψηφιακές συσκευές όπως ηλεκτρονικοί υπολογιστές, δίσκοι NAS (Network Attached Storage) ή ακόμα και συσκευές κινητής τηλεφωνίας κατανοεί ότι μία προσεκτικότερη

αρχική κατάσχεση θα βοηθήσει τη μελλοντική εργαστηριακή έρευνα και θα μειώσει το χρόνο απόδοσης της δικαιοσύνης.

Φυσικά την πλέον σημαντική θέση των λόγων απόκλισης από τις συμβατικές τεχνικές κατέχει η επιτόπια έρευνα σε ένα live υπολογιστικό σύστημα για την ανακάλυψη κρυπτογράφησης ή ύπαρξης προγραμμάτων καταστροφής δεδομένων και η λήψη των απαραίτητων μέτρων για τη διατήρηση των δεδομένων. Λόγος επιτακτικός εφόσον η εναλλακτική επιλογή είναι η πλήρης απώλεια τους.

Όσο για τις συσκευές κινητής τηλεφωνίας ή άλλες φορητές συσκευές σίγουρα αποτελεί απολύτως αιτιολογημένη πράξη η αλλαγή των ρυθμίσεων τους σε κατάσταση πτήσης (Airplane Mode) έτσι ώστε να απομονώνονται από δίκτυα και να ελαχιστοποιείται ό κίνδυνος απομακρυσμένης κακόβουλης διαχείρισης των δεδομένων τους.

Συνοψίζοντας θα πρέπει να τονιστεί η εξαιρετική σημασία της άρτιας εκπαίδευσης και τεχνογνωσίας από τους επαγγελματίες του κλάδου της Ψηφιακής Εγκληματολογίας. Η βαθιά γνώση των βασικών συμβατικών πρακτικών παρέχει τη δυνατότητα της σωστής απόφασης για το πότε αυτές θα πρέπει να εγκαταλειφθούν.

Και η ανάγκη λήψης τέτοιων αποφάσεων είναι απολύτως συνυφασμένη με τον πυρήνα της ίδιας της ψηφιακής εγκληματολογικής έρευνας. Είναι η ευμετάβλητη φύση της έρευνας και το απροσδόκητο που τη διακρίνει από τους λοιπούς κλάδους της Εγκληματολογίας. Και όταν ένας τομέας χαρακτηρίζεται από μία τόσο ασταθή δυναμική συνεπάγεται άμεσα και μεγάλους κινδύνους. Ο εξεταστής είναι αυτός που θα πρέπει στο τέλος της έρευνας να δικαιολογήσει την κάθε του ενέργεια και να πιστοποιήσει ότι όλα τα αποτελέσματα είναι αδιαμφισβήτητα.

Κι όπως ισχύει γενικότερα στην επιστήμη, οι κίνδυνοι αντιμετωπίζονται με τη γνώση. Κάθε είδους γνώση έχει συγκεκριμένη δομή και επίπεδα, για αυτό λοιπόν θα πρέπει κανείς να ξεκινάει πάντα από τη βάση της. Κι επειδή η παρούσα ασχολείται με τη θεματική της «Ψηφιακής Εγκληματολογίας» θα κάνουμε προσπαθήσουμε να εντρυφήσουμε στη βάση αυτής συνεχίζοντας στο επόμενο κεφάλαιο με την ανάλυση του Συστήματος Αρχείων και την Ανάκτηση Δεδομένων.

Κεφάλαιο 2°: Το Σύστημα Αρχείων NTFS

2.1 Τί είναι ένα Σύστημα Αρχείων;

Για να μπορέσει να απαντηθεί η παραπάνω ερώτηση θα πρέπει πρώτα να παρουσιαστούν τα βασικά μέρη και έννοιες ενός ψηφιακού αποθηκευτικού μέσου επί του οποίου ενυπάρχει ένα Σύστημα Αρχείων.

Ψηφιακά Αποθηκευτικά Μέσα

Σε κάθε ψηφιακή συσκευή αποθήκευσης μνήμης τα δεδομένα εγγράφονται με τη μορφή ενός μαγνητικού πεδίου ή μιας ηλεκτρικής κένωσης που αντιπροσωπεύουν μία τιμή «ΝΑΙ» ή «ΟΧΙ», την οποία γνωρίζουμε ως δυαδικό ψηφίο ή bit.

Οι παραδοσιακοί σκληροί δίσκοι απαρτίζονται από έναν αριθμό περιστρεφόμενων δίσκων φτιαγμένων από συμπαγές αλουμίνιο με διάφορες μορφές μαγνητικών υποστρωμάτων. Αυτά τα στρώματα υλικού αποτελούν το πεδίο επί του οποίου εγγράφονται τα δεδομένα με τη βοήθεια κεφαλών και τη χρήση ηλεκτρικού ρεύματος.

Από το 2007 με την έλευση στην αγορά των δίσκων συμπαγούς κατάστασης (SDD) αυτή η αρχιτεκτονική άλλαξε. Αυτού του είδους οι δίσκοι δεν περιέχουν κινητά μέρη αλλά αποτελούνται

από τσιπ μη πτητικής μνήμης τα οποία μπορούν να διατηρούν δεδομένα ακόμα και όταν δεν διατρέχονται από ρεύμα.

Αυτές οι διαφορετικές προσεγγίσεις στη εν γένει λειτουργία τους δεν επηρεάζουν κανένα άλλο τομέα των δίσκων που σχετίζεται με τις δομές αποθήκευσης δεδομένων. Και τα δύο είδη ακολουθούν το ίδιο θεωρητικό μοντέλο για την καταχώριση, διαχείριση και αποθήκευση πληροφορίας και στη θεωρητική τους βάση χρησιμοποιούν τις ίδιες κατασκευαστικές δομές.

Τομείς (Sectors)

Προκειμένου οι συσκευές αποθήκευσης δεδομένων να γίνουν χρηστικές κάποιου είδους τάξης χρειάζεται να εφαρμοστεί επί αυτών, η οποία αναφέρεται ως Λογική Δομή Δίσκου (Logical Disk Structure). Η Λογική Δομή Δίσκου κατανέμει περιοχές του μέσου σε ισομεγέθη καθορισμένα μπλοκ χώρου που ονομάζονται τομείς. Ένας τομέας είναι:

- Καθορισμένου σταθερού μεγέθους
- Η μικρότερη μονάδα που μπορεί να εγγραφεί στον δίσκο (writeable)
- Η μικρότερη μονάδα που μπορεί να διευθυνσιοδοτηθεί στον δίσκο (addressable)
- Η μικρότερη μονάδα που μπορεί να κατανεμηθεί στον δίσκο (allocable)

Διευθυνσιοδότηση Λογικών Μπλοκ (Logical Block Addressing)

Η διευθυνσιοδότηση λογικών μπλοκ αποτελεί την βασική μέθοδο διευθυνσιοδότησης στα αποθηκευτικά μέσα. Πραγματοποιείται από τον ελεγκτή του δίσκου και το σύστημα BIOS και συμπεριφέρεται στους τομείς του δίσκου σαν να ήταν τοποθετημένοι γραμμικά αριθμώντας τους με διαδοχικούς αριθμούς. Οι αριθμοί αυτοί δεν χρειάζεται να έχουν σχέση με την πραγματική φυσική θέση του κάθε τομέα απλά απλοποιούν τη διαδικασία εγγραφής και ανάκτησης δεδομένων.

Με τα δύο αυτά δομικά στοιχεία επιτυγχάνεται μία πρώτη χαρτογράφηση του δίσκου. Σε συνέχεια, και βάση των προηγούμενων, για να μπορέσουν να εγγραφούν δεδομένα σε ένα ψηφιακό αποθηκευτικό μέσο υπάρχουν τρεις διεργασίες που πρέπει να συμβούν:

- Χαμηλού επιπέδου διαμόρφωση (Low-Level Formatting), διαδικασία που λαμβάνει χώρα κατά την κατασκευή του μέσου στο εργοστάσιο και κατά την οποία ο αποθηκευτικός χώρος του μέσου χωρίζεται σε διακριτά κομμάτια που ονομάζονται τομείς.
- Διαμέριση (Partitioning), διαδικασία που πραγματοποιείται από τον χρήστη κατά την οποία ο αποθηκευτικός χώρος του μέσου διαιρείται λογικά σε ενιαία μέρη που χρησιμοποιούνται για να δημιουργήσουν πολλαπλούς τόμους (Volumes) σε αυτό.
- Υψηλού επιπέδου διαμόρφωση (High-Level Formatting), διαδικασία που πραγματοποιείται από τον χρήστη και κατά την οποία οι τόμοι αποκτούν ένα επιλεγμένο Σύστημα Αρχείων υπεύθυνο για τη διαχείριση των δεδομένων τους.

Σύστημα Αρχείων

Συνεπώς Σύστημα Αρχείων ονομάζεται το σύνολο των διαδικασιών που οργανώνουν και διαχειρίζονται τα δεδομένα ενός μέσου σε μία ιεραρχία καταλόγων, υποκαταλόγων και αρχείων και το οποίο εγκαθίσταται στο αποθηκευτικό ψηφιακό μέσο κατά τη διαδικασία της υψηλού επιπέδου διαμόρφωσης (High-Level Formatting). Οι λειτουργίες που επιτελεί είναι:

- Παρακολουθεί τον κατανεμημένο και τον ελεύθερο χώρο

- Διατηρεί ονόματα αρχείων, καταλόγων και τη δομή αυτών
- Καταγράφει που ακριβώς είναι αποθηκευμένο κάθε αρχείο στο μέσο

Υπάρχουν διάφορα Συστήματα αρχείων, για ένα πλήθος λειτουργικών συστημάτων, όπως τα FAT, NTFS, ExFAT, Ext3, Ext4, NFS, JFSS2 κ.α. καθένα με τη δική του ιδιαίτερη δομή και δυνατότητες. Η επιλογή του κατάλληλου εξαρτάται από πολλαπλούς παράγοντες ανά περίσταση όπως:

- Είδος του αποθηκευτικού μέσου στο οποίο θα χρησιμοποιηθεί (σκληρός δίσκος, αφαιρούμενο αποθηκευτικό μέσο διασύνδεσης τύπου «USB» κλπ)
- Υποστηριζόμενο μέγεθος αρχείων
- Βελτιστοποίηση στραμμένη προς την απόδοση
- Βελτιστοποίηση στραμμένη προς την χωρητικότητα (προσφέρει συμπίεση αρχείων)
- Βελτιστοποίηση στραμμένη προς την ασφάλεια (προσφέρει κρυπτογράφηση)
- Υποστήριξη ανάκτησης αρχείων μετά από κατάρρευση συστήματος (καταγραφή ημερολογίου-journaling) και άλλα.

Η κύρια λειτουργία του Συστήματος Αρχείων ως διαχειριστή των αρχείων στο μέσο, το καθιστά ένα από τα πολυτιμότερα συστατικά μέρη σε μία ψηφιακή εγκληματολογική εξέταση καθώς η φύση και η σημαντικότητα των πληροφοριών που προσφέρει είναι εξαιρετικής σημασίας και καθορίζει αναλόγως τη μεθοδολογία της εξέτασης που θα χρησιμοποιηθεί.

2.2 Το Σύστημα Αρχείων NTFS

2.2.1 Σύντομη Ιστορική Αναδρομή

Το Σύστημα Αρχείων NTFS αποτελεί το προεπιλεγμένο Σύστημα Αρχείων που χρησιμοποιείται από τα Λειτουργικά Συστήματα της Microsoft για σταθερούς δίσκους καθιστώντας το, το πιο ευρέως χρησιμοποιούμενο Σύστημα Αρχείων παγκοσμίως.

Η Microsoft χρειαζόταν ένα Σύστημα Αρχείων που θα ήταν πιο αξιόπιστο, πιο αποδοτικό και με περισσότερα διαχειριστικά εργαλεία έτσι ώστε να παραμείνει ανταγωνιστική, ειδικότερα στην αγορά των υπολογιστών-εξυπηρετητών (Servers). Επιπλέον η ραγδαία αύξηση του μεγέθους των αποθηκευτικών μέσων είχε ήδη ξεπεράσει κατά πολύ της δυνατότητες του Συστήματος Αρχείων FAT, το οποίο και ανάγκαζε τη Microsoft να σχεδιάσει ένα σύστημα από το μηδέν.

Τα επιπρόσθετα χαρακτηριστικά και η σταθερότητα του γρήγορα κατέστησαν το Σύστημα Αρχείων NTFS το προεπιλεγμένο Σύστημα Αρχείων για όλα τα Λειτουργικά Συστήματα της Microsoft μετά τα Windows ME. Το NTFS είναι επίσης ένα επεκτάσιμο Σύστημα Αρχείων και μπορεί εύκολα να τροποποιηθεί ώστε να αποκριθεί στην εκθετική αύξηση των μεγεθών των σύγχρονων αποθηκευτικών μέσων.

Το NTFS αναπτύχθηκε τη δεκαετία του 1990 ως αντικαταστάτης του FAT. Έχει δεχθεί τέσσερις αναβαθμίσεις από την αρχική του έκδοση, καθεμία εκ των οποίων συνόδευαν νέες κυκλοφορίες Λειτουργικών Συστημάτων της εταιρείας. Τα ιστορικά στάδια που ακολούθησε είναι τα ακόλουθα:

NTFS v1.0

Το Σύστημα Αρχείων NTFS κυκλοφόρησε για πρώτη φορά με το Λειτουργικό Σύστημα Windows NT 3.1 τον Ιούλιο του 1993.

NTFS v1.1

Η έκδοση 1.1 κυκλοφόρησε με το Λειτουργικό Σύστημα Windows NT 3.51 τον Μάιο του 1995. Εισήγαγε καινοτόμες λειτουργικές δομές όπως η συμπίεση αρχείων, η εναλλακτική ροή δεδομένων (Alternate Data Streams) και χρησιμοποιούσε λίστες ελέγχου εισόδου (Access Control Lists) για αυξημένη διαχειριστικό έλεγχο.

NTFS v1.2

Η έκδοση 1.2 θεωρείται ως η πρώτη σημαντική έκδοση του NTFS. Κυκλοφόρησε με το Λειτουργικό Σύστημα Windows NT 4.0 τον Ιούλιο του 1996. Υποστήριζε περιγραφείς ασφαλείας (Security Descriptors) για επαυξημένη ασφάλεια. Ο οδηγός (Driver) της παρούσας έκδοσης εντός του λειτουργικού συστήματος φέρει την έκδοση 4.0, γεγονός που οδήγησε εσφαλμένα στην αναφορά του Συστήματος Αρχείων ως NTFS 4.0.

NTFS v3.0

Η δεύτερη σημαντική έκδοση του NTFS είναι η 3.0 κυκλοφόρησε με το Λειτουργικό Σύστημα Windows 2000 τον Φεβρουάριο του 2000. Αυτή η έκδοση αναβάθμισε σημαντικά της διαχειριστικές δυνατότητες μέσω της ποσόστωσης δίσκου (Disk Quotas), της κρυπτογράφησης και των σημείων αναφοράς Reparse Points. Προσέθεσε ανθεκτικότητα μέσω της καταγραφής ημερολογίου συναλλαγών (Journaling) στο επίπεδο του ίδιου του Συστήματος Αρχείων. Ο οδηγός (Driver) της παρούσας έκδοσης είναι ο NTFS.sys 5.0.

NTFS v3.1

Υπήρξε μία μικρή αλλαγή στο Σύστημα Αρχείων, NTFS v3.1, που κυκλοφόρησε με το Λειτουργικό Σύστημα Windows XP τον Αύγουστο του 2001. Αυτή η έκδοση προσέθεσε επιπλέον πληροφορία στο Σύστημα Αρχείων (αριθμός εγγραφής εκάστοτε αρχείου στη ίδια την εγγραφή του στον πίνακα MFT) που διευκόλυναν την ανάκτηση κατεστραμμένων δεδομένων. Ο οδηγός (Driver) της παρούσας έκδοσης είναι ο NTFS.sys 5.1.

Με την κυκλοφορία των Λειτουργικών Συστημάτων Windows Vista και Windows 7 η δομή του NTFS έμεινε απαράλλακτη. Μερικές αλλαγές που προστέθηκαν είναι ποια δεδομένα του δίσκου επιβιώνουν της διενέργειας πλήρους διαμόρφωσης (Full Format) και η προσθήκη της δυνατότητας ανακατανομής μεγέθους διαμερίσματος (Partition Resizing). Παρόλαυτα, αυτές ήταν επιπλέον δυνατότητες που δόθηκαν στο Λειτουργικό Σύστημα μέσω αναβάθμισης των οδηγών του Συστήματος Αρχείων κι όχι της γενικής του δομής. Τα Windows Vista φέρουν τον οδηγό NTFS.sys 6.0 και τα Windows 7 τον NTFS.sys 6.1.

Η κυκλοφορία των Windows 8 τον Οκτώβριο του 2012 έφερε μερικές μικρές αλλαγές. Η έκδοση παραμένει η NTFS 3.1 (όπως και για τα μεταγενέστερα Windows 10) αλλά για τόμους οι οποίοι διαμορφώνονται με το αναλυόμενο Σύστημα Αρχείων λόγω αναβάθμισης του προηγούμενου Λειτουργικού Συστήματος (Windows 7). Σε δίσκους που παράγονται με προεγκαταστημένο το λειτουργικό σύστημα Windows 8 παρατηρείται ή αναγραφή του αριθμητικού 3.1.80. Μία ακόμα σημαντική αλλαγή είναι το γεγονός ότι δίσκοι με το νέο Λειτουργικό Σύστημα Windows 8 δεν παρουσιάζουν αναχρονιστική συμβατότητα (backward compatibility) με τη σύμβαση ονοματοδοσίας αρχείων DOS 8.3. Ο οδηγός (Driver) της έκδοσης NTFS για τα Windows 8 είναι ο NTFS.sys 6.2.

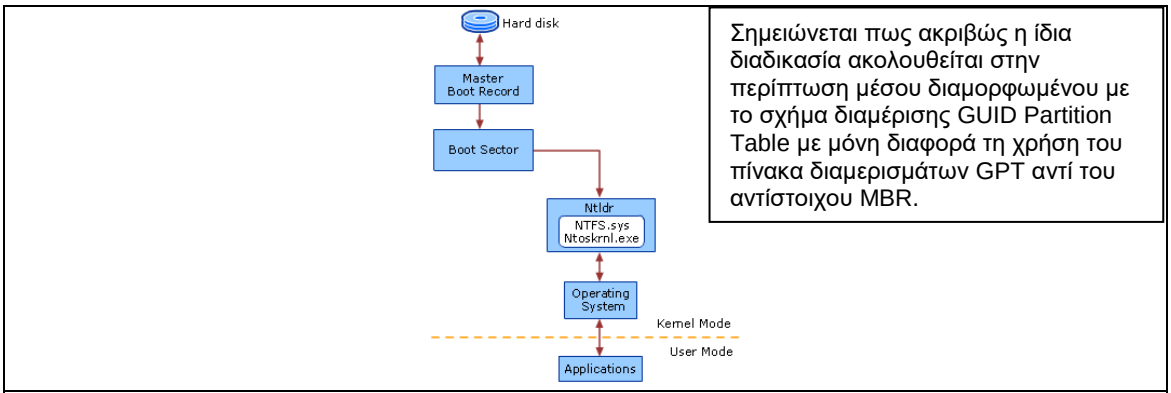
Ένας από τους κυριότερους λόγους που ο εξεταστής θα πρέπει να γνωρίζει τις διαφορετικές εκδόσεις κατά την εργαστηριακή εξέταση είναι οι κομβικές διαφορές που παρουσιάζονται μεταξύ

τους. Σημεία κλειδιά για την έρευνα αποτελούν το μέγεθος και το περιεχόμενο της επικεφαλίδας μίας εγγραφής αρχείου που διαφοροποιούνται σημαντικά και θα αναλυθούν παρακάτω.

2.2.2 Ιδιαίτερα Χαρακτηριστικά του Συστήματος Αρχείων NTFS

Το Σύστημα Αρχείων NTFS αποτέλεσε μία επαναστατική προσέγγιση στην ιεράρχηση δομών δεδομένων. Απομακρύνθηκε από εδραιωμένες τεχνικές προγραμματισμού για την αποθήκευση πληροφορίας, όπως η επίπεδη γραμμική αναζήτηση και ο διαχωρισμός του Συστήματος Αρχείων σε Περιοχή Συστήματος και Δεδομένων. Εισήγαγε πρωτοποριακές διαχειριστικές δυνατότητες συμβάλλοντας στη δημιουργία ενός σταθερού, ασφαλούς συστήματος προσανατολισμένου στην αποδοτικότητα και την ασφάλεια.

Η βασική αρχιτεκτονική εκκίνησης ενός συστήματος που χρησιμοποιεί το Σύστημα Αρχείων NTFS παρουσιάζεται στην παρακάτω εικόνα. Στον πρώτο φυσικό τομέα του μέσου εντοπίζεται το Master Boot Record το οποίο περιέχει κώδικα εκκίνησης, αναγνωριστικά/υπογραφές δίσκων και τον πίνακα με τα πρωταρχικά διαμερίσματα. Η διαδικασία εκκίνησης περνάει στον τομέα εκκίνησης (Boot Sector) του διαμερίσματος που περιέχει το Λειτουργικό Σύστημα σύμφωνα με την ανάλυση του πίνακα διαμερισμάτων, την εύρεση του ενεργού διαμερίσματος και την εντολή JUMP που δίνεται από το Master Boot Record. Ο κώδικας του Boot Sector του ενεργού διαμερίσματος φορτώνεται στη μνήμη RAM και παρέχει δείκτες (pointers) για τα αρχεία συστήματος NTFS.sys Ntoskrnl.exe τα οποία με τη σειρά τους φορτώνονται στη μνήμη. Η προσπέλαση τους φορτώνει το Λειτουργικό Σύστημα μετά από τους απαραίτητους ελέγχους και η διαδικασία εκκίνησης ολοκληρώνεται.



Βασική Αρχιτεκτονική εκκίνησης ενός συστήματος διαμορφωμένου με NTFS

Σημαντικότερες αναφορές στην πρωτοποριακή δομή του NTFS αποτελούν τα αραιά αρχεία (sparse files), η συμπίεση σε επίπεδο συστήματος (file compression) και η ποσόστωση δίσκου (Disk Quotas), λειτουργίες που εκτελούνται αυτομάτως από το Σύστημα Αρχείων και συμβάλλουν στη μεγάλη αποδοτικότητα αποθήκευσης.

Αραιά αρχεία (Sparse Files)

Τα αραιά αρχεία (sparse files) παρέχουν μία μέθοδο για εξοικονόμηση χώρου στο δίσκο για αρχεία τα οποία περιέχουν τόσο ουσιαστική πληροφορία όσο και μεγάλα κομμάτια δεδομένων που αποτελούνται από την δεκαεξαδική τιμή 0x00 (κενή τιμή). Στην περίπτωση που ένα αρχείο χαρακτηριστεί ως αραιό, τότε το NTFS κατανέμει για την αποθήκευσή του μόνο τους απαραίτητους τομείς (ή ομάδες τομέων που ονομάζονται clusters) που απαιτεί η ενεργή πληροφορία του όπως την καθορίζει κάθε φορά το ανάλογο πρόγραμμα που χρησιμοποιείται.

Αυτό έχει ως άμεσο αποτέλεσμα την εξοικονόμηση χώρου αλλά και ως έμμεσο την αύξηση της πολυπλοκότητας στη διαδικασία ανάκτησης του αρχείου κατά τη διάρκεια μίας ψηφιακής εγκληματολογικής έρευνας στο μέσο.

Συμπίεση Αρχείων (File Compression)

Το NTFS χρησιμοποιεί τον αλγόριθμο συμπίεσης LZNT1 για τη συμπίεση των αρχείων. Τα αρχεία συμπιέζονται σε κομμάτια των 16 clusters. Ο αλγόριθμος συμπίεσης είναι σχεδιασμένος να υποστηρίζει μεγέθη cluster μέχρι 4KB που είναι και η προεπιλεγμένη τιμή για τόμους διαμορφωμένους με το εν λόγω Σύστημα Αρχείων. Όταν το μέγεθος των cluster έχει ρυθμιστεί σε μεγαλύτερη τιμή, η συμπίεση δεν είναι διαθέσιμη ως λειτουργία. Ο χώρος που καταλάμβαναν τα δεδομένα που πλέον έχουν συμπιεστεί αντιμετωπίζονται σαν ανενεργές σελίδες αραιών αρχείων και δεν γράφονται στο δίσκο. Επομένως η εργαστηριακή ανάκτηση τέτοιων δεδομένων είναι και πάλι μια πρόκληση για τον εξεταστή.

Ποσόστωση Δίσκου (Disk Quotas)

Μία ακόμη νέα λειτουργία που παρουσίασε το NTFS ήταν η ποσόστωση δίσκου. Με τη χρήση της δίνεται η δυνατότητα στον διαχειριστή του συστήματος να ορίσει έναν ανώτατο χώρο αποθήκευσης που οι χρήστες δύνανται να χρησιμοποιήσουν. Ο διαχειριστής έχει επίσης τη δυνατότητα να ελέγχει τι χώρο έχει καταλάβει ο κάθε χρήστης και να ρυθμίσει την έκδοση προειδοποιητικών μηνυμάτων όταν αυτός πλησιάζει στο ανώτατο όριο που του έχει εκχωρηθεί. Η δυνατότητα αυτή αντικατοπτρίζει τις βλέψεις της Microsoft για επέκτασή της στην αγορά των λειτουργικών συστημάτων για υπολογιστές-εξυπηρετητες και της βελτιστοποίησης που επιδίωξε να επιτύχει με το Σύστημα Αρχείων NTFS σε περιβάλλοντα δικτύων.

Επιπροσθέτως, στο πλαίσιο της αυξημένης πολιτικής ασφάλειας το NTFS προσέφερε νέες δυνατότητες όπως καταγραφή ημερολογίου (Journaling), κρυπτογράφηση σε επίπεδο συστήματος (Encryption), περιγραφείς ασφαλείας (Security Descriptors) και λίστες ελέγχου προσπέλασης (Access Control List), χαρακτηριστικά τα οποία προσφέρουν μεγάλο εύρος διαχειριστικών ενεργειών σε αντίθεση με προγενέστερα Συστήματα Αρχείων.

Καταγραφή ημερολογίου (Journaling)

Το NTFS είναι ένα καταγραφικό σύστημα και χρησιμοποιεί το αρχείο μεταδεδομένων $LogFile για να καταγράφει αλλαγές που λαμβάνουν χώρα στον τόμο. Με αυτόν τον τρόπο το σύστημα μπορεί να διατηρεί μία υγιή κατάσταση σε περίπτωση κατάρρευσης συστήματος καθώς όσες ενέργειες και συναλλαγές διεκόπησαν, μπορούν να επανέλθουν στην πρότερη κατάστασή τους η οποία είχε καταγραφεί. Ακόμα το USN Journal είναι ένα χαρακτηριστικό διαχείρισης που καταγράφει τροποποιήσεις σε αρχεία και καταλόγους του τόμου καθώς επίσης και διάφορα άλλα χαρακτηριστικά τους. Περιέχεται στο αρχείο μεταδεδομένων $Extend και προσδίδει ένα επιπλέον επίπεδο ασφάλειας με την καταγραφή κάθε είδους ενέργειας επί στοιχείων του συστήματος. Η πληροφόρηση που παρέχει στον εξεταστή είναι πολύτιμη καθώς αποδεικνύει αλληλεπίδραση του χρήστη με αρχεία.

Κρυπτογράφηση (Encryption)

Το NTFS χρησιμοποιεί το Σύστημα Κρυπτογράφησης Αρχείου (EFS) το οποίο παρέχει ισχυρή κρυπτογράφηση η οποία παράλληλα είναι απαρατήρητη στον χρήστη καθως λαμβάνει χώρα στο επίπεδο του Συστήματος Αρχείων. Το EFS λειτουργεί σε συνδυασμό με την υπηρεσία EFS, τη διεπαφή προγραμματισμού εφαρμογών CryptoAPI της Microsoft και την βιβλιοθήκη

EFS File System Run-Time Library (FSRTL). Το EFS χρησιμοποιεί ένα συνδυασμό τεχνικών κρυπτογράφησης. Το αρχείο αρχικά κρυπτογραφείται με ένα συμμετρικό κλειδί (File Encryption Key - FEK) που προσφέρει ταχύτητα για την κρυπτογράφηση μεγάλου όγκου δεδομένων. Το κλειδί FEK κατόπιν κρυπτογραφείται με το δημόσιο κλειδί που σχετίζεται με το χρήστη που κατέχει το αρχείο και αποθηκεύεται σε ένα χαρακτηριστικό εναλλακτικής ροής δεδομένων (ADS) του. Για να αποκρυπτογραφήσει το αρχείο, το NTFS χρησιμοποιεί πρώτα το ιδιωτικό κλειδί του χρήστη για να αποκρυπτογραφήσει το κλειδί FEK και κατόπιν το τελευταίο για να αποκρυπτογραφήσει το ίδιο το περιεχόμενο του αρχείου. Η δυνατότητα κρυπτογράφησης που προσφέρει ενδογενώς το NTFS είναι αμοιβαίως αλληλοαποκλειόμενη με τη δυνατότητα συμπίεσης που επίσης προσφέρει. Αν και το σύστημα EFS δεν είναι από τα πιο επαρκή που υπάρχουν στον τομέα της κρυπτογραφίας δεν παύει να αποτελεί ένα ενσωματωμένο εργαλείο του Συστήματος Αρχείων και συνεπώς να προσθέτει επιπλέον επίπεδα στην ασφάλειά του.

Λίστες Ελέγχου Προσπέλασης (Access Control Lists)

Στο NTFS κάθε αρχείο ή κατάλογος έχει ένα συγκεκριμένο περιγραφέα ασφαλείας (Security Descriptor) που καθορίζει τον κάτοχο του και περιέχει δύο λίστες ελέγχου προσπέλασης (Access Control Lists). Η πρώτη λίστα ονομάζεται επιλεκτική (DACL) και καθορίζει τους ακριβής τύπους αλληλεπίδρασης (Ανάγνωση, Εγγραφή, Διαγραφή κλπ) που επιτρέπονται για κάθε χρήστη ή γκρουπ χρηστών επί του αρχείου. Η δεύτερη λίστα ονομάζεται λίστα συστήματος (SACL) και καθορίζει ποιες αλληλεπιδράσεις με το αρχείο ή τον κατάλογο πρέπει να ελέγχονται και αν θα πρέπει να καταγράφονται. Η δυνατότητα που προσφέρει το συγκεκριμένο χαρακτηριστικό καθιστά το Σύστημα Αρχείων NTFS ένα απολύτως ασφαλές περιβάλλον για διαχείριση οργανισμών με πολλαπλούς χρήστες. Οι δυνητικές συνέπειες που μπορεί να έχει σε μία ψηφιακή εγκληματολογική έρευνα είναι προς όφελος του εξεταστή ο οποίος μπορεί ενδεχομένως να λάβει πολλαπλές πληροφορίες από τις επιπλέον δομές που του προσφέρει το Σύστημα.

Εξίσου βασικό χαρακτηριστικό του NTFS είναι ο τρόπος ομαδοποίησης των δεδομένων που εφαρμόζει. Παρατηρώντας το Σύστημα Αρχείων NTFS στο βαθύτερο επίπεδο της ανάλυσης των byte φαίνεται ότι μεγάλος όγκος πληροφορίας είναι τοποθετημένος μαζί χωρίς ουσιαστικό διαχωρισμό. Στην πραγματικότητα είναι οργανωμένος σε ομάδες που ονομάζονται δομές δεδομένων, όπως για παράδειγμα οι εγγραφές αρχείων, οι επικεφαλίδες χαρακτηριστικών κ.α. Οι περισσότερες τιμές που χρησιμοποιούνται σε αυτό το επίπεδο είναι ακέραιοι αριθμοί οι οποίοι δύναται να έχουν τόσο αρνητική όσο και θετική τιμή (signed integers), γεγονός που αυξάνει την αποδοτικότητα προσθέτει όμως πολυπλοκότητα και κινδύνους στην ανάλυση.

Σημαντικό σημείο στην εργαστηριακή έρευνα αποτελεί το γεγονός ότι επειδή τα Windows και κατ' επέκταση και το Σύστημα Αρχείων NTFS έχουν σχεδιαστεί για χρήση σε επεξεργαστές αρχιτεκτονικής βασισμένης στην Intel (Intel based processors), χρησιμοποιούν τη διάταξη τύπου "Little Endian" για τα byte. Συνεπώς το λιγότερο σημαντικό byte μίας τιμής δεδομένων εγγράφεται πρώτο κατά την αποθήκευση της ομάδας των bytes στο μέσο.

Τέλος βασικότερο χαρακτηριστικό του NTFS που το ξεχωρίζει από άλλα Συστήματα Αρχείων είναι η προσέγγιση του σχετικά με τα δεδομένα. Ένας τόμος διαμορφωμένος με NTFS δε διαχωρίζει τον χώρο του σε κατηλλειμένη περιοχή Συστήματος (System Reserved Area) και περιοχή δεδομένων (Data Area). Για το NTFS όλα τα δεδομένα αποθηκεύονται σε αρχεία, συμπεριλαμβανομένου και των δεδομένων του Συστήματος. Για το NTFS τα πάντα είναι αρχεία. Τα αρχεία δεδομένων συστήματος αποκαλούνται αρχεία μεταδεδομένων (Metadata Files). Ο ορισμός του όρου "Μεταδεδομένα" σημαίνει πολύ απλά δεδομένα για άλλα δεδομένα. Τα αρχεία μεταδεδομένων αποτελούν ουσιαστικά τη δομική βάση του Συστήματος Αρχείων NTFS. Λόγω της ιδιαίτερης σημασίας τους αλλά και των κινδύνων που ενέχει η τροποποίησή τους τα αρχεία Μεταδεδομένων είναι κρυφά αρχεία και δεν είναι προσβάσιμα από τους χρήστες σε ένα "ζωντανό" σύστημα.

Τα δύο σημαντικότερα αρχεία μεταδεδομένων σε ένα Σύστημα Αρχείων NTFS είναι τα αρχεία $Boot και $MFT που αποτελούν τη βασική υποδομή όλου του συστήματος.

2.2.3 Βασική Υποδομή του Συστήματος Αρχείων NTFS

Το Σύστημα Αρχείων NTFS αντιμετωπίζει ολόκληρο τον τόμο ως έναν ενιαίο χώρο αποθήκευσης δεδομένων χωρίς καμία περεταίρω διαμέριση ή διαχωρισμό. Ένεκα αυτού είναι καίριας σημασίας η γενική κατασκευαστική του δομή έτσι ώστε να αποφεύγονται σφάλματα κατά την εγγραφή και διαχείριση των δεδομένων.

Την "ραχοκοκαλιά" αυτής της δομής αποτελούν δύο αρχεία μεταδεδομένων, η ανάλυση και ο συνδυασμός των οποίων χαρτογραφεί πλήρως τον τόμο και παρέχει με απόλυτη λεπτομέρεια τη θέση οποιουδήποτε αρχείου στο μέσο. Η ύπαρξη τους, δε, είναι τόσο κρίσιμη για το Σύστημα που και τα δύο έχουν αντίγραφο ασφαλείας σε έτερη θέση στο δίσκο.

Εγγραφή Εκκίνησης Τόμου (Volume Boot Record)

Όπως και σε όλα τα άλλα Συστήματα Αρχείων της Microsoft, ο πρώτος τομέας ενός τόμου διαμορφωμένου με το NTFS περιέχει την Εγγραφή Εκκίνησης Τόμου (VBR). Συχνά αποκαλείται και τομέας εκκίνησης (Boot Sector) λόγω της απαραίτητης θέσης που κατέχει στην ομώνυμη διαδικασία. Το VBR είναι ένα αρχείο μεταδεδομένων με την ονομασία $Boot και στην πραγματικότητα οι πρώτοι δεκαέξι τομείς του τόμου κατανέμονται σε αυτό.

Ο κύριος σκοπός που επιτελεί είναι ότι παρέχει κρίσιμες παραμέτρους για τον τόμο και το Σύστημα Αρχείων, δείκτες σε συστατικά στοιχεία του Συστήματος και κώδικα εκκίνησης της συσκευής (Bootstrap code).

Οι κύριες καταχωρίσεις στο αρχείο $Boot αναφέρουν το μέγεθος των cluster, το μέγεθος του τόμου, την τοποθεσία του πίνακα $MFT και του αντιγράφου ασφαλείας του. Το αρχείο $Boot κατέχει τόσο σημαντική θέση στη δομή του Συστήματος Αρχείων NTFS που ένα δεύτερο αντίγραφο ασφαλείας του εντοπίζεται στο τέλος του τόμου.

Πολλές από της πληροφορίες που περιέχονται εντός του αρχείου $Boot είναι κληρονομικές (legacy) και δεν είναι απαραίτητες από το Σύστημα για να εκκινήσει τον τόμο (mount). Επιπλέον, πολλά σημεία της δομής του έχουν αφεθεί κενά κατά τον αρχικό προγραμματισμό του, προκειμένου να συμβάλουν στην επεκτασιμότητα του Συστήματος Αρχείων στο μέλλον. Η αξία του για μια εγκληματολογική έρευνα είναι πρώτιστης σημασίας καθώς παρέχει τη γενική διάταξη του τόμου, τα απαραίτητα δεδομένα για την ορθή πλοήγηση σε αυτόν ενώ επίσης είναι το μοναδικό αρχείο το οποίο υποδεικνύει τη θέση του πίνακα $MFT, απαραίτητου επίσης στοιχείου για την εξέταση.

Στον παρακάτω πίνακα παρατίθεται η δομή δεδομένων των πρώτων 512 bytes του αρχείου $Boot, ήτοι ο πρώτος λογικός τομέας του τόμου.

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Όνομα	Περιγραφή
00	3	Εντολή Jump	Εντολή Jump στον κώδικα εκκίνησης
03	8	Αναγνωριστικό ID	ASCII "NTFS"
0B	2	Bytes ανά τομέα	Ο συνδυασμός αυτών των δύο τιμών παρέχει το μέγεθος των Clusters
0D	1	Τομείς ανά Cluster	
0E	2	Κατηλλειμένοι τομείς	0x00
10	5	Δεν χρησιμοποιείται από το NTFS	Πρέπει να είναι 0x00

15	1	Περιγραφέας Μέσου	Πάντα 0xF8
16	2	Δεν χρησιμοποιείται από το NTFS	Πρέπει να είναι 0x00
18	4	Δεν χρησιμοποιείται από το NTFS	Αναχρονιστική πληροφορία BPB (Bios Parameter Block)
1C	4	Κρυφοί τομείς	Τομείς πριν από την αφετηρία του τόμου
20	4	Δεν χρησιμοποιείται από το NTFS	Πρέπει να είναι 0x00
24	4	Δεν χρησιμοποιείται από το NTFS	Πάντα 0x80 00 80 00
28	8	Συνολικοί τομείς Τόμου	Μέγεθος τόμου
30	8	Αφετηρία περιοχής $MFT	Η τιμή αφορά τον λογικό αριθμό cluster (LCN), ήτοι από την αρχή του παρόντος τόμου και όχι όλου του δίσκου
38	8	Αφετηρία περιοχής $MFTMirr	
40	1	Μέγεθος έκαστης καταχώρησης $MFT	•Θετική τιμή= αριθμός Cluster •Αρνητική τιμή= αριθμός byte (δύναμη του 2 υψωμένη στην απόλυτη τιμή του αριθμού)
44	1	Clusters ανά ενταμιευτή καταλόγου (index buffer)	Μέγεθος index buffer
48	8	Σειριακός αριθμός τόμου	
50	4	Άθροισμα Ελέγχου (Checksum)	Checksum για τον τομέα εκκίνησης
54	426	Κώδικας εκκίνησης	Οδηγίες εκκίνησης
1FE	2	Υπογραφή τομέα εκκίνησης	0x55 0xAA
Δομή Δεδομένων του Τομέα Εκκίνησης			

Στην ακόλουθη εικόνα απεικονίζεται ο πρώτος τομέας ενός αρχείου $Boot:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	EB	52	90	4E	54	46	53	20	20	20	20	00	02	08	00	00	ëR NTFS
00000010	00	00	00	00	00	F8	00	00	3F	00	FF	00	80	00	01	00	ø ? ÿ €
00000020	00	00	00	00	80	00	80	00	FF	2F	05	00	00	00	00	00	€ € ÿ/
00000030	55	37	00	00	00	00	00	00	02	00	00	00	00	00	00	00	U7
00000040	F6	00	00	00	01	00	00	00	36	26	36	FC	35	36	FC	80	ö 6&6ü56ü€
00000050	00	00	00	00	FA	33	C0	8E	D0	BC	00	7C	FB	68	C0	07	ú3ÀŽÐ¼ |ûhÀ
00000060	1F	1E	68	66	00	CB	88	16	0E	00	66	81	3E	03	00	4E	hf Ë^ f > N
00000070	54	46	53	75	15	B4	41	BB	AA	55	CD	13	72	0C	81	FB	TFSu ´A»ªUÍ r û
00000080	55	AA	75	06	F7	C1	01	00	75	03	E9	DD	00	1E	83	EC	Uªu ÷Á u éÝ ƒì
00000090	18	68	1A	00	B4	48	8A	16	0E	00	8B	F4	16	1F	CD	13	h ´HŠ ‹ô Í
000000A0	9F	83	C4	18	9E	58	1F	72	E1	3B	06	0B	00	75	DB	A3	ŸƒÄ žX rá; uÛ£
000000B0	0F	00	C1	2E	0F	00	04	1E	5A	33	DB	B9	00	20	2B	C8	Á. Z3Û¹ +È
000000C0	66	FF	06	11	00	03	16	0F	00	8E	C2	FF	06	16	00	E8	fÿ ŽÂÿ è
000000D0	4B	00	2B	C8	77	EF	B8	00	BB	CD	1A	66	23	C0	75	2D	K +Èwï¸ »Í f#Àu-
000000E0	66	81	FB	54	43	50	41	75	24	81	F9	02	01	72	1E	16	f ûTCPAu$ ù r
000000F0	68	07	BB	16	68	52	11	16	68	09	00	66	53	66	53	66	h » hR h fSfSf
00000100	55	16	16	16	68	B8	01	66	61	0E	07	CD	1A	33	C0	BF	U h¸ fa Í 3À¿
00000110	0A	13	B9	F6	0C	FC	F3	AA	E9	FE	01	90	90	66	60	1E	¹ö üó ªéþ f`
00000120	06	66	A1	11	00	66	03	06	1C	00	1E	66	68	00	00	00	f¡ f fh
00000130	00	66	50	06	53	68	01	00	68	10	00	B4	42	8A	16	0E	fP Sh h ´BŠ
00000140	00	16	1F	8B	F4	CD	13	66	59	5B	5A	66	59	66	59	1F	‹ôÍ fY[ZfYfY
00000150	0F	82	16	00	66	FF	06	11	00	03	16	0F	00	8E	C2	FF	, fÿ ŽÂÿ
00000160	0E	16	00	75	BC	07	1F	66	61	C3	A1	F6	01	E8	09	00	u¼ faÃ¡ö è
00000170	A1	FA	01	E8	03	00	F4	EB	FD	8B	F0	AC	3C	00	74	09	¡ú è ôëý‹ð¬< t
00000180	B4	0E	BB	07	00	CD	10	EB	F2	C3	0D	0A	41	20	64	69	´ » Í ëòÃ A di
00000190	73	6B	20	72	65	61	64	20	65	72	72	6F	72	20	6F	63	sk read error oc
000001A0	63	75	72	72	65	64	00	0D	0A	42	4F	4F	54	4D	47	52	curred BOOTMGR
000001B0	20	69	73	20	63	6F	6D	70	72	65	73	73	65	64	00	0D	is compressed
000001C0	0A	50	72	65	73	73	20	43	74	72	6C	2B	41	6C	74	2B	Press Ctrl+Alt+
000001D0	44	65	6C	20	74	6F	20	72	65	73	74	61	72	74	0D	0A	Del to restart
000001E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000001F0	00	00	00	00	00	00	8A	01	A7	01	BF	01	00	00	55	AA	Š § ¿ Uª

Τομέας Εκκίνησης (Boot Sector)

Τα δεδομένα του μεταφράζονται ως εξής (Little Endian):

- ID – NTFS
- Bytes ανά τομέα – 0x200 = 512
- Τομείς ανά Cluster – 0x08 = 8
- Κρυφοί τομείς – 0x01 00 08 =65.664
- Ολικοί τομείς στον τόμο – 0x05 2F FF = 339.967
- Αφετηρία χώρου $MFT – 0x37 55 = 14.165
- Αφετηρία χώρου $MFTMirr – 0x02 =2
- Μέγεθος καταχώρησης $MFT – 0xF6 (signed integer) = -10, ήτοι 2^{10} =1024 Bytes
- Σειριακός αριθμός τόμου = 36 26 36 FC 35 36 FC 80
- Υπογραφή τομέα εκκίνησης = 0x55 0xAA

Κύριος Πίνακας Αρχείων (Master File Table)

Το πιο σημαντικό αρχείο μεταδεδομένων στο NTFS είναι το αρχείο $MFT το οποίο αποτελεί τον Κύριο Πίνακα Αρχείων για το Σύστημα. Χρησιμοποιείται για να αποθηκεύει όλες τις πληροφορίες για κάθε αρχείο που βρίσκεται στον τόμο, όπως ημεροχρονολογίες και ώρες δημιουργίας, άδειες προσπέλασης, όνομα, περιεχόμενο και να εντοπίζει την τοποθεσία του σε αυτόν. Επιτυγχάνει τη λειτουργία του αποθηκεύοντας τουλάχιστον μία καταχώριση για κάθε αρχείο, η οποία ονομάζεται εγγραφή αρχείου (File Record) και αποκτά έναν μοναδικό αριθμό αναγνώρισης. Αυτή η αφαιρετική προσέγγιση επιτρέπει την εύκολη προσθήκη νέων χαρακτηριστικών και αυξάνει ακόμα περισσότερο την ήδη μεγάλη επεκτασιμότητα του Συστήματος σε συνδυασμό και με έτερες δομές δεδομένων που το απαρτίζουν και έχουν αφεθεί κενές για μελλοντική χρήση.

Το $MFT αποθηκεύει στην πρώτη θέση καταχώρισης του, μία εγγραφή για τον ίδιο του τον εαυτό, η οποία περιγράφει μεταξύ άλλων το μέγεθός του και την τοποθεσία του στον τόμο. Συνεπώς το ίδιο το $MFT χρειάζεται να αυτό-αναλυθεί ούτως ώστε να γνωρίζει τη θέση του και την πλήρη έκτασή του όπως κάθε άλλο αρχείο. Κάθε αρχείο ή κατάλογος, συμπεριλαμβανομένου και του ριζικού καταλόγου (Root Directory), έχουν τουλάχιστον μία εγγραφή στον πίνακα $MFT η οποία περιγράφει τη θέση τους και το μέγεθός τους κατά τον ίδιο τρόπο.

Το $MFT είναι τόσο σημαντικό για το Σύστημα που ένα αντίγραφο ασφαλείας των τεσσάρων πρώτων εγγραφών του αποθηκεύονται στο αρχείο μεταδεδομένων $MFTMirr, το οποίο αποθηκεύεται σε έτερο μέρος του τόμου και ταυτόχρονα αποτελεί τη δεύτερη εγγραφή του πίνακα $MFT. Αν η πρώτη καταχώριση στον πίνακα $MFT είναι αλλοιωμένη (corrupted) από οποιαδήποτε αιτία, τότε το NTFS διαβάζει τη δεύτερη για να βρει το $MFTMirr, του οποίου η πρώτη εγγραφή είναι ίδια με την πρώτη του $MFT. Με αυτόν τον τρόπο ο πίνακας ανακατασκευάζεται παρέχοντας μία δομή ασφάλειας στο Σύστημα. Οι τοποθεσίες και των δύο αρχείων εντός του τόμου, βρίσκονται αποθηκευμένες στον τομέα εκκίνησης.

Συνήθως , το 12% του τόμου είναι εκ των προτέρων κατηλλειμένο για το $MFT. Το $MFT υποστηρίζεται από αλγορίθμους ελαχιστοποίησης κατακερματισμού, παρόλαυτα ειδικά για τόμους που περιέχουν το Λειτουργικό Σύστημα της συσκευής, είναι σύνηθες ο πίνακας $MFT να κατακερματίζεται σε μεγάλο βαθμό. Άμεσο αντίκτυπο αυτού του φαινομένου στην εγκληματολογική ανάλυση είναι το γεγονός της ανεύρεσης προηγούμενων εγγραφών του $MFT στον μη κατανεμημένο χώρο μετά από μια διαδικασία διαμόρφωσης δίσκου (Format) και κατ' επέκταση η ανάκτηση αρχείων που θεωρούνταν διαγεγραμμένα.

Οι πρώτες είκοσι έξι εγγραφές στον πίνακα $MFT περιέχουν αρχεία μεταδεδομένων απαραίτητα για τη λειτουργία του Συστήματος και είναι πάντοτε ίδιες για κάθε τόμο διαμορφωμένο με το Σύστημα αρχείων NTFS. Αυτές παρατίθενται στον κάτωθι πίνακα:

Αρχείο Μεταδεδομένων	Ονομασία	Εγγραφή	Λειτουργία Αρχείου
Master File Table	$MFT	0	Περιέχει μία εγγραφή για κάθε αρχείο στον τόμο
MFT Mirror	$MFTMirr	1	Περιέχει ένα αντίγραφο ασφαλείας των 4 πρώτων εγγραφών του $MFT
Log File	$LogFile	2	Χρησιμοποιείται για επαναφορά σε περίπτωση αλλοιωμένου Συστήματος
Volume	$Volume	3	Πληροφορίες τόμου

			(ετικέτα, έκδοση Συστήματος, flags)
Attribute Definitions	$AtrrDef	4	Απαριθμεί τα ονόματα των Attributes, αναγνωριστικά και περιγραφή τους
Root Index	.	5	Ριζικός κατάλογος
Cluster Bitmap	$Bitmap	6	Αναπαράσταση κατανεμημένων και μη clusters
Boot Sector	$Boot	7	Τομέας εκκίνησης και κώδικας Bootstrap
Bad Cluster File	$BadClus	8	Αραιό αρχείο που περιέχει τα εσφαλμένα cluster του τόμου
Security file	$Secure	9	Ευρετήριο των ρυθμίσεων ασφαλείας που μπορούν να εφαρμοστούν στα αρχεία του Συστήματος
Upper Case Table	$UpCase	10	Μετατρέπει τα μικρά γράμματα στους αντίστοιχους κεφαλαίους χαρακτήρες Unicode
Extended Attributes	$Extend	11	Κατάλογος που περιέχει προαιρετικές επεκτάσεις
		12-15	Κατειλημμένες για μελλοντική χρήση
		16-23	Μη χρησιμοποιούμενες
Quota	$Quota	24	Καταγράφει τα δικαιώματα χώρου των χρηστών στον τόμο
Object Identifier	$ObjId	25	Ευρετήριο όλων των μοναδικών αναγνωριστικών που έχουν αποδοθεί στα αρχεία του τόμου
Reparse	$Reparse	26	Ευρετήριο όλων των σημείων Reparse στον τόμο
Αρχεία Μεταδεδομένων στον Πίνακα $MFT			

Ο χώρος για αποθήκευση αρχείων χρηστών αρχίζει μετά την 26η καταχώριση του $MFT και συνεχίζει μέχρις ότου κάθε αρχείο ή κατάλογος στον τόμο έχει απεικονιστεί με τουλάχιστον μία εγγραφή στον πίνακα. Για το $MFT τα αρχεία και οι κατάλογοι, υποκατάλογοι, αρχεία μεταδεδομένων συστήματος αντιμετωπίζονται όλα με τον ίδιο ακριβώς τρόπο και θεωρούνται όλα αρχεία. Οι δομές αποθήκευσής τους δε διαφέρουν παρά μόνο σε ένα μόλις bit-σημαία (flag) το οποίο υποδεικνύει τη φύση της εγγραφής.

Το περιεχόμενο λοιπόν του $MFT αρχείου είναι ένα πλήθος εγγραφών που αποτελούν την οργανωτική υποδομή του Συστήματος Αρχείων NTFS και αποτελούν απαραίτητη πληροφορία

για την ψηφιακή εγκληματολογική εξέταση του μέσου. Ας δούμε, λοιπόν, πως αναλύονται περαιτέρω οι εγγραφές αρχείων του $MFT που θα αποτελέσουν και βάση για την επιτυχή διαδικασία ανάκτησης διαγεγραμμένων αρχείων.

2.2.4 Εγγραφές Αρχείων (File Records)

Το $MFT αποθηκεύει κάθε αρχείο σε ατομικές καταχωρίσεις που ονομάζονται εγγραφές αρχείων. Κάθε εγγραφή φέρει το δικό της μοναδικό αριθμό εγγραφής (ID), ο οποίος χρησιμοποιείται ως αναγνωριστικό για το αρχείο το οποίο η εγγραφή περιγράφει. Καθώς προστίθενται αρχεία στον τόμο, μία καταχώριση για το καθένα προστίθεται στο $MFT με διαδοχική σειρά.

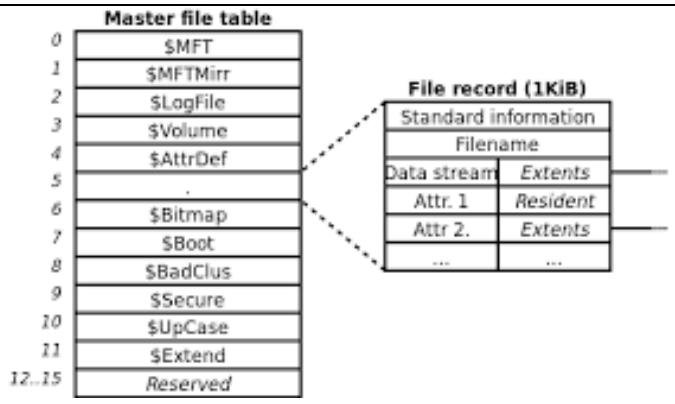
Εάν ένα αρχείο διαγραφεί, η εγγραφή που χρησιμοποιούταν για να περιγράψει αυτό το αρχείο καθίσταται ανενεργή αλλά δε διαγράφεται. Ο αλγόριθμος αποτροπής κατακερματισμού που χρησιμοποιεί όμως το $MFT θα ξαναχρησιμοποιήσει ανενεργές εγγραφές, σε περίπτωση αποθήκευσης νέου αρχείου στον τόμο, πρωτού δημιουργήσει καινούριες. Αυτό το χαρακτηριστικό έχει ισχυρό αντίκτυπο στην εγκληματολογική έρευνα καθώς οι “διαγεγραμμένες” καταχωρίσεις επαναγράφονται πολύ γρήγορα δημιουργώντας δυσχέρεια στην ανάκτηση δεδομένων.

Στην παρούσα έκδοση του Συστήματος Αρχείων NTFS το προεπιλεγμένο μέγεθος για κάθε εγγραφή αρχείου είναι 1024 bytes. Αυτή η τιμή μπορεί να επαληθευθεί καθώς δίνεται από τον τομέα εκκίνησης και ενδέχεται να τροποποιηθεί με μελλοντικές εκδόσεις.

Οι εγγραφές αρχείων χρησιμοποιούν μία πληθώρα δομών δεδομένων για να περιγράψουν ένα αρχείο. Διαχωρίζονται σε ξεχωριστά μπλοκ δεδομένων που περιέχουν πληροφορίες τόσο για την ίδια την εγγραφή όσο και για το αρχείο το οποίο η εγγραφή προσδιορίζει. Οι πληροφορίες για την ίδια την εγγραφή αποθηκεύονται σε μια δομή που αποκαλείται Επικεφαλίδα Εγγραφής Αρχείου (File Record Header). Πληροφορίες για το αρχείο περιέχονται σε διακριτά μπλοκ που ονομάζονται Χαρακτηριστικά του αρχείου (File Attributes). Κάθε χαρακτηριστικό αποθηκεύει ένα συγκεκριμένο είδος πληροφορίας για το αρχείο.

Η σήμανση της αφετηρίας μίας εγγραφής αρχείου πραγματοποιείται με την υπογραφή (signature) “FILE” σε χαρακτήρες ASCII. Το τέλος της σημαίνεται με την δεκαεξαδική τιμή 0XFF FF FF FF.

Η διαγραμματική απεικόνιση του $MFT με μία αναλυόμενη εγγραφή αρχείου παρουσιάζεται παρακάτω:



Ανάλυση εγγραφής αρχείου εντός του $MFT

Στο παρών διάγραμμα αναλύεται η εγγραφή αρχείου με το νούμερο πέντε, η οποία τυγχάνει η εγγραφή που περιγράφει τον ριζικό κατάλογο. Παρατηρείται ότι ακόμα και ένα τόσο κομβικής σημασίας αρχείο μεταδεδομένων όπως το root αρχείο του Συστήματος περιγράφεται με τον ίδιο τρόπο όπως κάθε άλλο αρχείο στον τόμο, καταλαμβάνοντας του ίδιου μεγέθους καταχώριση στο $MFT και αναλυόμενο στα ίδια διακριτά μπλοκ πληροφορίας.

Βασική δομή για την περαιτέρω άρτια ανάλυση μίας εγγραφής αρχείου αποτελεί η επικεφαλίδα της, η οποία αποκρύφτηκε από το ανωτέρω διάγραμμα. Ας την εξετάσουμε αναλυτικότερα.

Επικεφαλίδα Εγγραφής Αρχείου (File Record Header)

Κάθε εγγραφή αρχείου ξεκινάει με μία επικεφαλίδα, που παρέχει πληροφορίες για την ίδια την εγγραφή. Μία από τις σημαντικές διαφοροποιήσεις μεταξύ των αναθεωρήσεων του NTFS αποτελεί το μέγεθος της επικεφαλίδας. Στις εκδόσεις NTFS v1.x η επικεφαλίδα φέρει μήκος 48 bytes. Στις εκδόσεις NTFS v3.x το μήκος αυξήθηκε σε 56 bytes. Η επιπλέον πληροφορία που εισήχθη είναι ο αναγνωριστικός αριθμός $MFT της ίδιας της εγγραφής που προσφέρει ανυπολόγιστη βοήθεια στην ανάλυση ανακτημένων εγγραφών. Επειδή οι εκδόσεις NTFS v3.x ουσιαστικά σημαίνουν κάθε Λειτουργικό Σύστημα από τα Windows 2000 και έπειτα, είναι σχεδόν βέβαιο ότι ο σύγχρονος εξεταστής θα αντιμετωπίσει μόνο τη νέα μορφή επικεφαλίδας κατά τις εγκληματολογικές έρευνες που θα διενεργήσει.

Διάταξη Fix-up (Fix-up Array)

Πριν συνεχίσουμε την ανάλυση της επικεφαλίδας θα πρέπει να αναφερθούμε σε μια δομή ελέγχου που εισήγαγε το NTFS για να επιτύχει αυξημένη αξιοπιστία. Η διάταξη Fix-up είναι ένα χαρακτηριστικό που ανιχνεύει την αλλοίωση ή αποτυχία μονών τομέων.

Για να το επιτύχει αυτό μία τιμή ελέγχου δημιουργείται που έχει μήκος δύο byte. Η τιμή ελέγχου γράφεται στα τελευταία δύο byte κάθε τομέα της εγγραφής αρχείου και είναι η πρώτη καταχώριση στη διάταξη. Οι τιμές που θα βρισκόντουσαν κανονικά στο τέλος κάθε τομέα αλλά αντικαταστάθηκαν από την τιμή ελέγχου, αποτελούν κατά σειρά τις άλλες δύο καταχωρίσεις στη διάταξη.

Το Σύστημα, πριν από κάθε χρήση της εγγραφής, συγκρίνει τις τιμές στο τέλος κάθε τομέα και αν αυτές δεν ταιριάζουν σηματοδοτείται κάποιο σφάλμα στην εγγραφή. Η υπογραφή της ενδέχεται να τροποποιηθεί από “FILE” σε “BAAD” και να αγνοηθεί από το Σύστημα. Σε κάποια Λειτουργικά Συστήματα (όπως για παράδειγμα στα Windows 8) η εγγραφή θα διαγραφεί.

Η διάταξη Fix-up είναι μία τεχνική ελέγχου που το Σύστημα Αρχείων NTFS χρησιμοποιεί και σε άλλες δομές δεδομένων εκτός του $MFT, αλλά είναι χαρακτηριστικό μόνο των εσωτερικών αυτών δομών του και δεν εφαρμόζεται σε τομείς που περιέχουν δεδομένα αρχείων.

Σημαντικό γεγονός στην εγκληματολογική ανάλυση ενός μέσου παίζει το γεγονός ότι κατά την περιήγηση του χρήστη, το NTFS αυτομάτως φροντίζει να ανταλλάσει την τιμή ελέγχου με τις αρχικές τιμές κάθε τομέα ώστε να γίνεται σωστή μετάφραση της εγγραφής. Στη χειροκίνητη ανάλυση αυτή η εργασία θα πρέπει να πραγματοποιηθεί από τον εξεταστή προκειμένου να αποφευχθούν λάθη κατά την ερμηνεία της πληροφορίας.

Η δομή δεδομένων μίας επικεφαλίδας εγγραφής αρχείου παρουσιάζεται στον παρακάτω πίνακα και εν συνεχεία αναλύονται τα σημαντικότερα στοιχεία της:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Δεδομένα	Περιγραφή
00	4	Υπογραφή	“FILE” (ή BAAD αν εντοπιστεί σφάλμα στη διάταξη Fix-up)

04	2	Θέση διάταξης Fix-up	Αριθμός bytes από την αρχή της επικεφαλίδας
06	2	Αριθμός καταχωρίσεων στη διάταξη Fix-up	Αριθμός καταχωρίσεων μεγέθους 2 byte
08	8	Ακολουθία αρχείου $LogFile	Ο αριθμός της τελευταίας καταχώρισης συναλλαγής στο αρχείο $LogFile
10	2	Άθροισμα διαδοχής	Άθροισμα των φορών που η εγγραφή έχει διαγραφεί
12	2	Άθροισμα συνδέσμων Hard Link	Άθροισμα των χαρακτηριστικών ονόματος
14	2	Θέση πρώτου χαρακτηριστικού	Αριθμός bytes από την αρχή της επικεφαλίδας
16	2	Σημαίες (Flags) κατάστασης χρήσης •Bit 0 = Κατάσταση χρήσης •Bit 1 = Κατάσταση καταλόγου	•0x00 = διαγραμμένο αρχείο •0x01 = ενεργό αρχείο •0x02 = διαγραμμένος κατάλογος •0x03 = ενεργός κατάλογος
18	4	Λογικό μέγεθος της εγγραφής αρχείου (Logical)	Byte που χρησιμοποιεί στην πραγματικότητα η εγγραφή
1C	4	Υλικό μέγεθος της εγγραφής αρχείου (Physical)	Ολικά byte που καταλαμβάνει η εγγραφή
20	8	Αναφορά στην βασική εγγραφή	Χρησιμοποιείται μόνο όταν η εγγραφή είναι μεγαλύτερη από μία καταχώρηση στο $MFT
28	2	Αριθμός επόμενου χαρακτηριστικού	Άθροισμα των χαρακτηριστικών της εγγραφής (συμπεριλαμβάνει και τα διαγραμμένα χαρακτηριστικά)
2A	~	Διάταξη Fix-up και λοιπά χαρακτηριστικών	NTFS 3.0
2C	4	Αριθμός εγγραφής αρχείου στο $MFT	NTFS 3.1 +
30	~	Διάταξη Fix-up και λοιπά χαρακτηριστικών	NTFS 3.1 +
Δομή δεδομένων επικεφαλίδας μίας εγγραφής αρχείου			

Ακολουθία αρχείου $LogFile ($LogFile Sequence Number)

Αποτελεί τον αριθμό-αναγνωριστικό για την καταχώρηση που έλαβε χώρα στο αρχείο $LogFile για την τελευταία συναλλαγή του αρχείου με το Σύστημα Αρχείων NTFS. Το Σύστημα

διατηρεί αυτό το αρχείο προκειμένου να επιτυγχάνεται ανάκτηση δεδομένων σε περίπτωση αποτυχίας οποιασδήποτε συναλλαγής. Το ημερολόγιο που διατηρεί το αρχείο επιτρέπει στην μη επιτυχή συναλλαγή να αναιρεθεί, αποκαθιστώντας με αυτόν τον τρόπο την υγιή κατάσταση του Συστήματος.

Άθροισμα Διαδοχής (Sequence Count)

Δείχνει τον αριθμό των φορών που μία εγγραφή αρχείου έχει χρησιμοποιηθεί. Όταν το αρχείο $MFT δημιουργηθεί αρχικά, όλες οι εγγραφές έχουν το συγκεκριμένο αριθμό ορισμένο ως 1. Κατόπιν το άθροισμα διαδοχής αυξάνεται κάθε φορά που η εγγραφή χαρακτηρίζεται ως ανενεργή, με άλλα λόγια όταν το αρχείο (ή ο κατάλογος) διαγράφεται. Στην απίθανη περίπτωση που η εγγραφή επαναχρησιμοποιηθεί πάνω από 65.535 φορές τότε ο αριθμός εξαιτίας του μήκους του (2 byte, ήτοι 0xFFFF) ξεκινάει ξανά από το 1. Το συγκεκριμένο χαρακτηριστικό κατέχει έναν από τους σημαντικότερους ρόλους σε μία ψηφιακή εγκληματολογική εξέταση αφού ανακατασκευάζει τη δομή των φακέλων όπως θα αναλυθεί αργότερα.

Αναφορά στην βασική εγγραφή (File Reference to Base Record)

Κάποιες φορές, μία μόνο εγγραφή στο $MFT δεν είναι αρκετή για να χωρέσει όλη την πληροφορία για ένα αρχείο, συνήθως όταν το αρχείο κατακερματιστεί αρκετά. Σε αυτές τις περιπτώσεις υπάρχει μόνο μία εγγραφή αρχείου που χρησιμοποιείται για να περιγράψει το αρχείο, η οποία ονομάζεται εγγραφή βάσης ή βασική. Οι υπόλοιπες εγγραφές που αποθηκεύουν τις επιπλέον πληροφορίες ονομάζονται εγγραφές επέκτασης (extension records). Η εγγραφή επέκτασης φέρει τον αριθμό εγγραφής $MFT της εγγραφής βάσης στην επικεφαλίδα της. Αν αυτός ο αριθμός είναι 0 τότε εξετάζεται η εγγραφή βάσης.

Η σημαντικότερη παρατήρηση που πρέπει να γίνει για τις εγγραφές αρχείων είναι ότι τα δεδομένα τους αφορούν μόνο το παρόν αρχείο στο οποίο αναφέρονται. Όταν μία εγγραφή αρχείου επαναχρησιμοποιείται όλες οι μη χρησιμοποιούμενες περιοχές της επαναγράφονται με την δεκαεξαδική τιμή 0x00 (κενή τιμή). Αυτό είναι ιδιαίτερα σημαντικό σε μία εγκληματολογική εξέταση αφού οποιοδήποτε χώρο υπολείμματος (Slack Space) στην εγγραφή απαραιτήτως συνδέεται με το τρέχων αρχείο και συνεπώς και τον τρέχοντα χρήστη. Πρακτικό παράδειγμα θα παρουσιαστεί στο οικείο μέρος της παρούσης.

Επόμενη δομή που ακολουθεί την επικεφαλίδα μίας εγγραφής αρχείου είναι τα χαρακτηριστικά αυτής τα οποία και αποθηκεύουν όλη την ουσιαστική πληροφορία για κάθε αρχείο.

2.2.5 Χαρακτηριστικά Αρχείων (File Attributes)

Κάθε εγγραφή αρχείου στο $MFT αποθηκεύει δεδομένα για το εκάστοτε αρχείο χρησιμοποιώντας μία γραμμική αποθήκη διαδοχικών δομών πληροφορίας που ονομάζονται χαρακτηριστικά και ακολουθούν αμέσως μετά το τέλος της επικεφαλίδας. Τα χαρακτηριστικά έχουν τη δική τους δομή που αποτελείται από επικεφαλίδες και περιεχόμενο.

Οι επικεφαλίδες και το περιεχόμενο του κάθε χαρακτηριστικού είναι διακριτά μπλοκ δεδομένων και πρέπει να ερμηνεύονται ατομικά για να παρέχουν σωστή πληροφόρηση για το αρχείο. Τα μεγέθη του περιεχομένου των χαρακτηριστικών ποικίλλουν και μερικές φορές ενδέχεται να μην βρίσκονται εντός της εγγραφής του οπότε και καταλαμβάνουν χώρο σε διαφορετικό σημείο του τόμου. Τα χαρακτηριστικά είναι πολλών διαφορετικών τύπων και το καθένα παρέχει διαφορετική μορφή πληροφορίας για το αρχείο.

Τύποι Χαρακτηριστικών (Attribute Types)

Τα χαρακτηριστικά προσφέρουν μεγάλη ποικιλία τύπων και περιέχουν στο περιεχόμενό τους πληροφορία για διαφορετικές πτυχές του αρχείου. Μία εγγραφή αρχείου θα περιέχει αρκετούς διαφορετικούς τύπους χαρακτηριστικών ανάλογα με τη φύση του αρχείου που περιγράφει. Στον παρακάτω πίνακα παρουσιάζονται όλοι οι δυνατοί τύποι χαρακτηριστικών που μία εγγραφή μπορεί να περιλαμβάνει και η περιγραφή τους:

Αναγνωριστικός κωδικός χαρακτηριστικού	Ονομασία χαρακτηριστικού	Περιγραφή
10 00 00 00	$Standard_Information	Περιέχει εξουσιοδοτήσεις αρχείου, ημεροχρονολογίες, διαχειριστικές πληροφορίες
20 00 00 00	$Attribute_List	Τοποθεσίες όλων των χαρακτηριστικών που δε χωρούν σε μία εγγραφή αρχείου
30 00 00 00	$File_Name	Ονομασία αρχείου
40 00 00 00	$Volume_Version	Έκδοση Τόμου (NTFS v1.x μόνο)
40 00 00 00	$Object_ID	Περιέχει μοναδικά αναγνωριστικά GUID για το αρχείο (NTFS v3.x μόνο)
50 00 00 00	$Security_Descriptor	Έλεγχος προσπέλασης και ιδιότητες ασφαλείας του αρχείου (αρχεία μεταδεδομένων μόνο για το NTFS v3.x)
60 00 00 00	$Volume_Name	Χρησιμοποιούνται μόνο στο αρχείο μεταδεδομένων $Volume και περιέχουν το όνομα του τόμου και την έκδοση του NTFS
70 00 00 00	$Volume_Information	
80 00 00 00	$Data	Τα πραγματικά δεδομένα του αρχείου ή δείκτες προς αυτά
90 00 00 00	$Index_Root	Ο γονικός κόμβος ενός ταξινομημένου δέντρου που απαριθμεί τα αρχεία-παιδιά ενός καταλόγου
A0 00 00 00	$Index_Allocation	Δείχνει τις τοποθεσίες των ενταμιευτών ευρετηρίου (Index Buffers) ενός μεγάλου καταλόγου
B0 00 00 00	$Bitmap	Παρακολουθεί την κατάσταση κατανομής μίας τοποθεσίας ή μίας οντότητας, αναλόγως του τύπου εγγραφής στην οποία εντοπίζεται
C0 00 00 00	$Symbolic_Link	Πληροφορίες για συνδέσμους Soft Link (NTFS v1.2 μόνο)
C0 00 00 00	$Reparse_Point	Παρόμοιο με ένα σύνδεσμο Soft Link (NTFS v3.x μόνο)
D0 00 00 00	$EA_Information	Προσφέρει συμβατότητα με το Σύστημα Αρχείων HPFS
E0 00 00 00	$EA	Προσφέρει συμβατότητα με το Σύστημα Αρχείων HPFS
00 01 00 00	$Logged_Utility_Stream	Περιέχει πληροφορίες και κλειδιά για κρυπτογραφημένα

		χαρακτηριστικά (NTFS v3.x μόνο)
Τύποι χαρακτηριστικών		

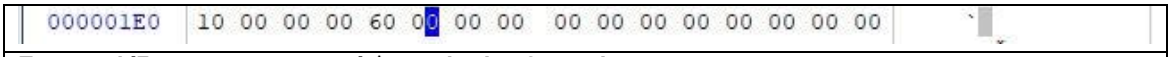
Κάθε τύπος χαρακτηριστικού έχει μία μοναδική δομή δεδομένων για το περιεχόμενό του και τα περισσότερα μπορούν να έχουν μη τοπικό (Non-Resident Content) περιεχόμενο. Η επικεφαλίδα για όλα τα χαρακτηριστικά είναι η ίδια.

Επικεφαλίδα Χαρακτηριστικού (Attribute Header)

Όλα τα χαρακτηριστικά ξεκινούν με μία επικεφαλίδα μήκους 16 byte η οποία περιέχει πληροφορίες για το ίδιο το χαρακτηριστικό. Αυτές συμπεριλαμβάνουν τον αναγνωριστικό κωδικό του χαρακτηριστικού, το μήκος του χαρακτηριστικού, αν το περιεχόμενό του είναι τοπικό ή όχι και πληροφορίες για το όνομα ροής (Stream Name) εάν υπάρχει. Η δομή της παρουσιάζεται στον κάτωθι πίνακα:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Ονομασία	Περιγραφή
00	4	Κωδικός χαρακτηριστικού	Ποικίλλει
04	4	Μήκος του χαρακτηριστικού	Μήκος σε bytes
08	1	Σημαία μη τοπικού περιεχομένου (Content Non-resident Flag)	•0x00 = τοπικό περιεχόμενο (Resident) •0x01 = μη τοπικό περιεχόμενο (Non-resident)
09	1	Μήκος της ονομασίας ροής (Stream Name)	Αριθμός χαρακτήρων Unicode
0A	2	Θέση της ονομασίας ροής (Stream Name)	Σε αριθμό bytes από την αρχή του χαρακτηριστικού
0C	2	Σημαίες χαρακτηριστικού (Flags)	•0x0001=συμπιεσμένο •0x4000=κρυπτογραφημένο •0x8000=αραιό
0E	2	Αναγνωριστικό χαρακτηριστικού	Αύξων αριθμός χαρακτηριστικού που προστέθηκε στην εγγραφή με διαδοχική σειρά
Δομή δεδομένων επικεφαλίδας χαρακτηριστικού			

Στην παρακάτω εικόνα παρατίθεται μία επικεφαλίδα χαρακτηριστικού και η λεπτομερής ανάλυσή της:



Επικεφαλίδα χαρακτηριστικού $Standard_Information

- Κωδικός χαρακτηριστικού – 0x10 00 00 00 = $Standard_Information
- Μήκος χαρακτηριστικού – 0x60 (Little Endian) = 96 bytes
- Σημαία μη τοπικού περιεχομένου – 0x00 = περιεχόμενο τοπικό
- Μήκος της ονομασίας ροής – 0x00 = δεν υπάρχει Name Stream
- Σημαίες χαρακτηριστικού – 0x00 = δεν έχουν τεθεί

- Αναγνωριστικό χαρακτηριστικού – 0x00 =πρώτο χαρακτηριστικό που προστέθηκε στην εγγραφή αρχείου

Περιεχόμενο Χαρακτηριστικού (Attribute Content)

Τα περιεχόμενα κάθε χαρακτηριστικού φέρουν διαφόρων μορφών πληροφορία για το αρχείο, από τα πραγματικά του δεδομένα μέχρι μεταδεδομένα συστήματος. Συνυπολογίζοντας την κάθε διαφορετική πληροφόρηση που προέρχεται από το σύνολο των χρησιμοποιούμενων χαρακτηριστικών σε μία εγγραφή, το Σύστημα αποκτά πλήρη εικόνα για το αρχείο και ξέρει ακριβώς πώς να το διαχειριστεί.

Στην παρούσα έκδοση του NTFS κάθε εγγραφή αρχείου στο $MFT έχει μήκος 1024 bytes και συνεπώς είναι περιορισμένη στο μέγεθος της πληροφορίας που μπορεί να αποθηκεύσει. Αν υπάρχουν πολλά δεδομένα που πρέπει να καταγραφούν για το αρχείο, τότε ορισμένα μέρη αυτών δύναται να μεταφερθούν σε άλλη περιοχή του δίσκου. Με αυτόν τον τρόπο εισάγονται οι έννοιες τοπικό και μη τοπικό χαρακτηριστικό.

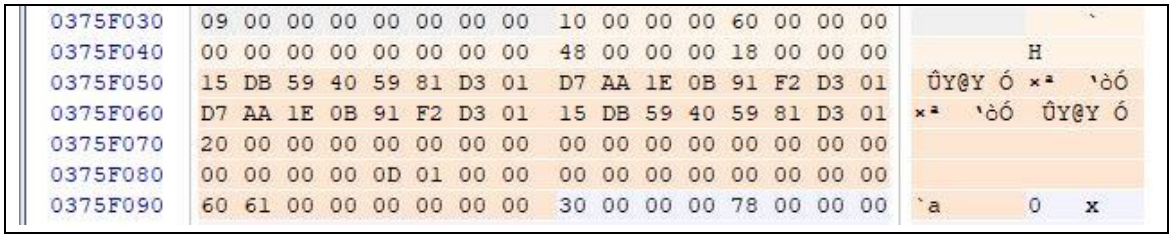
Τοπικό Χαρακτηριστικό (Resident Attribute)

Ένα χαρακτηριστικό ονομάζεται τοπικό όταν το περιεχόμενο του περιέχεται εξ' ολοκλήρου στην εγγραφή αρχείου του $MFT. Στην πραγματικότητα η ορθότερη ονομασία θα ήταν χαρακτηριστικό τοπικού περιεχομένου αλλά έχει επικρατήσει να ονομάζεται τοπικό χαρακτηριστικό. Η διαφοροποίηση είναι ότι οι επικεφαλίδες κάθε χαρακτηριστικού πρέπει οπωσδήποτε να περιλαμβάνονται στην εγγραφή του $MFT αλλά το περιεχόμενό τους όχι απαραιτήτως.

Εάν το περιεχόμενο ενός χαρακτηριστικού είναι τοπικό, τότε υπάρχει μία δομή δεδομένων τοπικής επικεφαλίδας αμέσως μετά τη γενική επικεφαλίδα του χαρακτηριστικού. Η τοπική επικεφαλίδα περιέχει το μέγεθος των δεδομένων του τοπικού περιεχομένου και τη θέση τους σε συνάρτηση με την αρχή του χαρακτηριστικού. Ο παρακάτω πίνακας δίνει τη δομή της τοπικής επικεφαλίδας:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Ονομασία	Περιγραφή
00	16	Επικεφαλίδα χαρακτηριστικού	Βλ. παραπάνω
10	4	Μέγεθος του περιεχομένου	Μέγεθος σε byte
14	2	Θέση του περιεχομένου	Συναρτήσει της αρχής του χαρακτηριστικού, μετρημένη σε byte
Δομή δεδομένων τοπικής επικεφαλίδας χαρακτηριστικού με τοπικό περιεχόμενο			

Στην ακόλουθη εικόνα παρουσιάζεται (με σήμανση πορτοκαλί χρώματος) ένα τοπικό χαρακτηριστικό με τις δομές δεδομένων που απαρτίζουν το πρώτο χαρακτηριστικό του:



Τοπικό χαρακτηριστικό

Επικεφαλίδα Χαρακτηριστικού

- Κωδικός χαρακτηριστικού – 0x10 00 00 00 = $Standard_Information
- Μήκος χαρακτηριστικού – 0x60 (Little Endian) = 96 bytes
- Σημαία μη τοπικού περιεχομένου – 0x00 = περιεχόμενο τοπικό
- Μήκος της ονομασίας ροής – 0x00 = δεν υπάρχει Name Stream
- Σημαίες χαρακτηριστικού – 0x00 = δεν έχουν τεθεί
- Αναγνωριστικό χαρακτηριστικού – 0x00 =πρώτο χαρακτηριστικό που προστέθηκε στην εγγραφή αρχείου

Τοπική Επικεφαλίδα

- Μέγεθος περιεχομένου – 0x48 = 72 bytes
- Θέση περιεχομένου – 0x18 = 24 bytes (από την αρχή του χαρακτηριστικού)

Το περιεχόμενο του ανωτέρω χαρακτηριστικού έχει σημανθεί με έντονο πορτοκαλί χρώμα.

Μη Τοπικό Χαρακτηριστικό (Non-resident Attribute)

Είναι δυνατόν το περιεχόμενο ενός χαρακτηριστικού να μεγαλώσει σε τέτοιο βαθμό ωστέ να είναι πολύ μεγάλο για να χωρέσει μέσα στους στενούς περιορισμούς που επιβάλει το μέγεθος της εγγραφής αρχείων. Σε αυτές τις περιπτώσεις το περιεχόμενο του χαρακτηριστικού αποθηκεύεται σε άλλο σημείο του τόμου και αποκαλείται ως μη τοπικό χαρακτηριστικό. Πρέπει να σημειωθεί ότι η επικεφαλίδα του χαρακτηριστικού παραμένει στην εγγραφή αρχείου οπότε μόνο το περιεχόμενό της μετατρέπεται σε μη τοπικό.

Η ανάλυση των μη τοπικών χαρακτηριστικών θα συνεχιστεί με μεγαλύτερη σαφήνεια στο κεφάλαιο αναφοράς στο χαρακτηριστικό $Data το οποίο είναι και το χαρακτηριστικό που εμφανίζει συχνότερα μη τοπικό περιεχόμενο.

Με την επεξήγηση των ανωτέρων δομών φαίνεται ήδη πόσο αυστηρά οργανωμένο είναι το Σύστημα Αρχείων NTFS. Η πολυπλοκότητά του οδηγεί σε ένα σταθερό, ασφαλές Σύστημα το οποίο είναι θετικό για τον χρήστη, αρνητικό όμως για τον εγκληματολόγο εξεταστή καθώς αυξάνει το βαθμό δυσκολίας της εξέτασης.

Η πλήρης και ενδελεχής έρευνα επί ενός ψηφιακού μέσου απαιτεί την άριστη γνώση και χειρισμό όλων των χαρακτηριστικών που περιλαμβάνει μία εγγραφή προκειμένου να αναλυθεί σωστά το αρχείο. Επειδή θεματική της παρούσας είναι η ανάκτηση δεδομένων, στη συνέχεια επικεντρωνόμαστε στα τρία πιο σημαντικά χαρακτηριστικά μιας εγγραφής που μας προσφέρουν τις απολύτως αναγκαίες πληροφορίες για την ανάκτηση του αρχείου που περιγράφουν, τα χαρακτηριστικά $Standard_Information, $File_Name και $Data. Αυτά τα τρία χαρακτηριστικά είναι και τα μόνα χαρακτηριστικά που μία εγγραφή αρχείου πρέπει απαραίτητα να περιέχει προκειμένου να αποτελεί έγκυρη καταχώριση στον πίνακα $MFT.

2.2.6 Το Χαρακτηριστικό $STANDARD_INFORMATION

Κάθε αρχείο φέρει υποχρεωτικά το χαρακτηριστικό $Standard_Information. Η κύρια πληροφορία που προσφέρει το εν λόγω χαρακτηριστικό είναι οι ημεροχρονολογίες και ώρες που αφορούν το αρχείο. Υπάρχουν ακόμα πληροφορίες ασφάλειας και δικαιωμάτων χρήσης. Το περιεχόμενο του χαρακτηριστικού πρέπει πάντοτε να είναι τοπικό εντός της εγγραφής $MFT.

Με την αυστηρή έννοια του όρου “ανάκτηση” θα μπορούσε κανείς να πει ότι το χαρακτηριστικό αυτό δεν είναι απαραίτητο για να ανακτηθεί ένα διαγεγραμμένο αρχείο από το μέσο. Οι απαραίτητες πληροφορίες για αυτήν την ενέργεια δίνονται στα επόμενα δύο χαρακτηριστικά που πρόκειται να αναλυθούν και πέραν αυτών κανένα άλλο δεν είναι απαραίτητο. Η φύση όμως μίας ψηφιακής εγκληματολογικής έρευνας, όπως προαναφέρθηκε στο 1^{ο} κεφάλαιο, είναι να παράσχει όλα εκείνα τα στοιχεία που θα στηρίξουν νομικά ολόκληρη την υπόθεση.

Το παρών χαρακτηριστικό, λοιπόν, αποτελεί ένα σημαντικό στοιχείο στην έρευνα καθώς η ερμηνεία των χρονικών σφραγίδων του αρχείου μπορεί να ανακατασκευάσει ένα πλήρες χρονολόγιο της αλληλεπίδρασης του χρήστη με το αρχείο. Η δομή δεδομένων του παρουσιάζεται στον επόμενο πίνακα:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Ονομασία	Περιγραφή
00	4	Κωδικός χαρακτηριστικού	0x10 00 00 00
04	4	Μήκος του χαρακτηριστικού	Μήκος σε bytes
08	1	Σημαία μη τοπικού περιεχομένου (Content Non-resident Flag)	•0x00 = τοπικό περιεχόμενο (Resident) •0x01 = μη τοπικό περιεχόμενο (Non-resident)
09	1	Μήκος της ονομασίας ροής (Stream Name)	Αριθμός χαρακτήρων Unicode
0A	2	Θέση της ονομασίας ροής (Stream Name)	Σε αριθμό bytes από την αρχή του χαρακτηριστικού
0C	2	Σημαίες αρχείου (Flags)	•0x0001=συμπιεσμένο •0x4000=κρυπτογραφημένο •0x8000=αραιό
0E	2	Αναγνωριστικό χαρακτηριστικού	Αύξων αριθμός χαρακτηριστικού που προστέθηκε στην εγγραφή με διαδοχική σειρά
10	4	Μέγεθος του περιεχομένου	Μέγεθος σε byte
14	2	Θέση του περιεχομένου	Συναρτήσει της αρχής του χαρακτηριστικού, μετρημένη σε byte
		Ονομασία Ροής (Stream Name)	Εάν Υπάρχει
00	8	Χρονοσφραγίδα δημιουργίας	Ημερομηνία/ώρα που το αρχείο δημιουργήθηκε στον τόμο
08	8	Χρονοσφραγίδα τροποποίησης αρχείου	Ημερομηνία/ώρα που το περιεχόμενο του αρχείου τροποποιήθηκε
10	8	Χρονοσφραγίδα τροποποίησης εγγραφής $MFT	Ημερομηνία/ώρα που το περιεχόμενο της εγγραφής $MFT του αρχείου τροποποιήθηκε
18	8	Χρονοσφραγίδα τελευταίας	Ημερομηνία/ώρα που το αρχείο προσπελάστηκε για

		προσπέλασης	τελευταία φορά
20	4	Σημαίες τύπου αρχείου	Βλέπε ακόλουθο πίνακα
24	4	Μέγιστος αριθμός εκδόσεων	Απενεργοποιημένο πάντα 0x00
28	4	Αριθμός έκδοσης	Η έκδοση του αρχείου
2C	4	Αναγνωριστικό κλάσης	Αναγνωριστικός αριθμός κλάσης
30	4	Αναγνωριστικό κατόχου	Αναγνωριστικό κατόχου για Quota
34	4	Αναγνωριστικό Ασφάλειας	Αναφορά στο αρχείο $Secure (NTFS v3.0 +)
38	8	Χρεωμένα Quota	Αριθμός bytes από το Quota του χρήστη (NTFS v3.0 +)
40	8	Αριθμός αναβάθμισης ακολουθίας (Update Sequence Number)	Δείκτης στο αρχείο $USN (NTFS v3.0 +)
Δομή δεδομένων χαρακτηριστικού $Standard_Information			

Το κόκκινο περίγραμμα στον ανωτέρω πίνακα περικλείει την επικεφαλίδα του χαρακτηριστικού, η οποία έχει την ίδια δομή για όλα τα χαρακτηριστικά. Οι δύο επόμενες γραμμές αποτελούν την επικεφαλίδα τοπικού χαρακτηριστικού αφού το $Standard_Information αποτελεί ένα τέτοιο. Εάν το χαρακτηριστικό φέρει ονομασία ροής, αυτή ακολουθεί την επικεφαλίδα τοπικού χαρακτηριστικού και παρατίθεται σε αυτό το σημείο.

Οι εναπομείνασες γραμμές είναι η δομή δεδομένων για το περιεχόμενο του χαρακτηριστικού $Standard_Information. Σημαντικό γεγονός που πρέπει να προσεχτεί είναι ότι η αρίθμηση των θέσεων σε αυτό το σημείο ξεκινάει ξανά από το μηδέν καθώς το περιεχόμενο του χαρακτηριστικού θεωρείται ξεχωριστή οντότητα για το NTFS.

Οι σημαίες είδους αρχείου (File Type Flags) χρησιμοποιούν “συμπυκνωμένα” bytes (packed bytes) οπού κάθε bit αποτελεί μία διαφορετική σημαία. Παρατίθενται στον ακόλουθο πίνακα:

Δεκαεξαδική Τιμή (0x)	Δυαδική Τιμή	Περιγραφή
00 01	0000 0000 0000 0001	Read only
00 02	0000 0000 0000 0010	Κρυφό
00 04	0000 0000 0000 0100	Αρχείο Συστήματος
00 20	0000 0000 0010 0000	Archive
00 40	0000 0000 0100 0000	Device
00 80	0000 0000 1000 0000	Κανονικό
01 00	0000 0001 0000 0000	Προσωρινό
02 00	0000 0010 0000 0000	Αραιό
04 00	0000 0100 0000 0000	Reparse Point
08 00	0000 1000 0000 0000	Συμπιεσμένο
10 00	0001 0000 0000 0000	Offline
20 00	0010 0000 0000 0000	Μη ταξινομημένο
40 00	0100 0000 0000 0000	Κρυπτογραφημένο
Σημαίες τύπων αρχείου		

Οι σημαίες δύναται να συνδυάζονται ώστε να παρέχουν πολυδιάστατη πληροφόρηση για το αρχείο. Για παράδειγμα ένα κρυφό, Read only αρχείο θα φέρει την δεκαεξαδική τιμή 0x0003 στο ανάλογο πεδίο.

Χρονοσφραγίδες στο Σύστημα Αρχείων NTFS

Το NTFS αποθηκεύει τις ημεροχρονολογίες και ώρες σε μία μορφή που αποκαλείται FILETIME. Το FILETIME είναι μία unsigned τιμή μεγέθους 64 bit που αντιπροσωπεύει τον αριθμό διαστημάτων των 100 νάνο-δευτερολέπτων που έχουν παρέλθει από την 1η Ιανουαρίου 1601 σε UTC. Συνεπώς όλες οι χρονοσφραγίδες στο NTFS αποθηκεύονται σε ζώνη ώρας UTC (Universal Coordinated Time) και όχι στην τοπική ώρα που βρίσκεται ρυθμισμένο το ψηφιακό μέσο.

Όταν το Λειτουργικό Σύστημα ή ένα εγκληματολογικό λογισμικό διαβάσουν το FILETIME, φροντίζουν να κάνουν τις κατάλληλες μετατροπές ώστε να παρουσιάζουν την τοπική ώρα στην οποία είναι ρυθμισμένη η συσκευή. Αυτό είναι πολύ σημαντικό στοιχείο σε μία εγκληματολογική εξέταση καθώς ο εξεταστής πρέπει πάντοτε να γνωρίζει τη ζώνη ώρας στην οποία είναι ρυθμισμένο το μέσο το οποίο εξετάζει προκειμένου να δώσει σωστά αποτελέσματα στην αναφορά του. Συχνά οι χρονοσφραγίδες στην εγκληματολογική πρακτική αναφέρονται ως ώρες MAC (MAC times) εξαιτίας των αρχικών Modified, Accessed, Created.

Οι ορισμοί για τις χρονοσφραγίδες είναι:

- Χρονοσφραγίδα δημιουργίας (Created Time) = η χρονική στιγμή κατά την οποία το αρχείο δημιουργήθηκε στον παρόντα τόμο. Η δημιουργία αρχείου σε ένα τόμο μπορεί να προκύψει είτε από την αρχική του δημιουργία σε αυτόν ή από μεταφορά ήδη δημιουργημένου αρχείου από άλλον τόμο σε αυτόν (λειτουργίες αντιγραφής/αποκοπής και επικόλλησης)
- Χρονοσφραγίδα τροποποίησης (Modified Time) = η χρονική στιγμή κατά την οποία το προσβάσιμο προς τον χρήστη περιεχόμενο του αρχείου τροποποιήθηκε τελευταία φορά
- Χρονοσφραγίδα τροποποίησης εγγραφής $MFT ($MFT Modified Time) = η χρονική στιγμή που το περιεχόμενο της εγγραφής αρχείου στον πίνακα $MFT τροποποιήθηκε τελευταία φορά
- Χρονοσφραγίδα τελευταίας προσπέλασης (Last Accessed Time) = η χρονική στιγμή που το αρχείο προσπελάστηκε τελευταία φορά είτε από ενέργεια του χρήστη (π.χ. άνοιγμα του αρχείου), είτε από ενέργεια λογισμικού (π.χ. έλεγχος αντιβιοτικού προγράμματος στα αρχεία του δίσκου), ή από αυτοματοποιημένη ενέργεια του Συστήματος Αρχείων (π.χ. αλλαγή εξουσιοδοτήσεων στο αρχείο λόγω αλλαγής γκρουπ ή domain του χρήστη).

Όπως φαίνεται και από τους ορισμούς τα πράγματα δεν είναι τόσο ξεκάθαρα όσον αφορά τις χρονοσφραγίδες των αρχείων. Σε αυτή τη δυσκολία έρχεται να προστεθεί η διαφορετική συμπεριφορά που έχουν ως προς την υλοποίησή τους τα διάφορα Λειτουργικά Συστήματα.

Για παράδειγμα στα Windows XP η χρονοσφραγίδα τελευταίας προσπέλασης ενός αρχείου ανανεώνεται ακόμα και όταν ο χρήστης κάνει δεξί κλικ επί του αρχείου ώστε να δει τις ιδιότητές του. Επίσης πρέπει να σημειωθεί ότι από το Λειτουργικό Σύστημα Windows Vista και έπειτα, η χρονοσφραγίδα τελευταίας προσπέλασης ενός αρχείου έχει διαπιστωθεί εργαστηριακά να μην ανανεώνεται, εκτός των αρχείων Microsoft Office τα οποία αποτελούν εξαίρεση του κανόνα.

Παρόλαυτα η σωστή ερμηνεία του συνδυασμού των χρονοσφραγίδων μπορεί να δώσει μεγάλη πληροφόρηση για την διαδρομή που έχει ακολουθήσει το αρχείο στο μέσο και την πιθανή διάδραση του χρήστη με αυτό, ερώτημα που είναι καίριας σημασίας στις περισσότερες

εγκληματολογικές έρευνες. Μία πλήρη ανάλυση όλων των δυνατών σεναρίων θα ξεπερνούσε τους σκοπούς της παρούσας αλλά παρακάτω θα παρουσιαστούν κάποιοι από τους πιο βασικούς συνδυασμούς.

Ερμηνεία Χρονοσφραγίδων στο Σύστημα Αρχείων NTFS

- Στην περίπτωση που η χρονοσφραγίδα δημιουργίας και τροποποίησης είναι ίδιες τότε προκύπτει με βεβαιότητα ότι το αρχείο έχει δημιουργηθεί εξ' αρχής στον παρών τόμο, απόρροια του οποίου είναι η αδιάψευστη γνώση του χρήστη της συσκευής για το αρχείο εφόσον αυτός τυγχάνει δημιουργός του.
- Στην περίπτωση που η χρονοσφραγίδα τροποποίησης εμφανίζει χρόνο προγενέστερο της ημερομηνίας δημιουργίας τότε προκύπτει με βεβαιότητα ότι το αρχείο είχε αρχικά δημιουργηθεί σε άλλο τόμο ή δίσκο και κατόπιν αντιγράφηκε στον παρόντα.
- Στην περίπτωση που ένας πολύ μεγάλος αριθμός αρχείων (ειδικά διαφορετικών τύπων μεταξύ τους) παρουσιάζει εξαιρετικά κοντινούς χρόνους στις χρονοσφραγίδες προσπέλασης τους, τότε ενδέχεται τα αρχεία να έχουν προσπελαστεί μαζικά από ένα πρόγραμμα (π.χ αντιβιοτικό λογισμικό) και όχι από τον χρήστη.
- Στην περίπτωση που ένας πολύ μεγάλος αριθμός αρχείων παρουσιάζει εξαιρετικά κοντινούς χρόνους στις χρονοσφραγίδες δημιουργίας τους, τότε ενδέχεται τα αρχεία να έχουν αποθηκευτεί στο μέσο ως αποτέλεσμα είτε αποσυμπίεσης από αρχείο συμπίεσης ή διαδικασίας κατωφόρτωσης download από κάποιου είδους δικτυακό περιβάλλον.

Οι χρονοσφραγίδες όπως φαίνεται αποτελούν και την ουσιαστικότερη πληροφορία που μπορεί να παράσχει το χαρακτηριστικό $Standard_Information.

Τα χαρακτηριστικά ακολουθούν το αμέσως προηγούμενο (ή την επικεφαλίδα της εγγραφής αρχείου στην περίπτωση που είναι το πρώτο) και γράφονται στην εγγραφή αρχείου με την αριθμητική σειρά που έχουν οι κωδικοί αναγνώρισής τους. Κατά συνέπεια το χαρακτηριστικό $Standard_Information θα καταλαμβάνει πάντα την πρώτη θέση μετά την επικεφαλίδα εγγραφής.

Τα χαρακτηριστικά στο NTFS, όπως επίσης και τα περιεχόμενά τους, ξεκινούν και τελειώνουν πάντα σε θέση πολλαπλάσια της τιμής 8 από την αρχή της εγγραφής αρχείου (8 byte boundary). Για να είναι βέβαιο ότι αυτό θα επιτευχθεί σε κάθε περίπτωση, ένα χαρακτηριστικό μπορεί να περιέχει γέμισμα (padding) μετά το περιεχόμενό του. Ως αυτού η μετάφραση των πεδίων που δίνουν τα μεγέθη χαρακτηριστικών και περιεχομένων τους θα πρέπει να γίνεται με προσοχή ώστε να αποφευχθεί λανθασμένη ανάλυση αυτών.

Το επόμενο χαρακτηριστικό που είναι απαραίτητο για την εγκληματολογική ανάκτηση αρχείων είναι το χαρακτηριστικό $File_Name.

2.2.7 Το Χαρακτηριστικό $File_Name

Το χαρακτηριστικό $File_Name παρέχει μία πλειάδα διαφορετικών πληροφοριών και για αυτόν τον λόγο είναι απαραίτητο σε μία εγκληματολογική εξέταση. Χρησιμοποιείται αρχικά για να αποθηκεύσει το όνομα του αρχείου και είναι πάντα τοπικό χαρακτηριστικό στην εγγραφή.

Πέρα από αυτή του τη λειτουργία το χαρακτηριστικό αυτό περιέχει ένα αντίγραφο των χρονοσφραγίδων που περιέχονται στο χαρακτηριστικό $Standard_Information με μόνη διαφορά ότι αυτές αναφέρονται στο χαρακτηριστικό αυτό και ανανεώνονται μόνο κατά τη ονομασία/μετονομασία του αρχείου ή την μετακίνηση του, διαφορετικά παραμένουν αδρανείς και ξεπερασμένες.

Τα πιο ενδιαφέροντα στοιχεία που προσφέρει το αναλυόμενο χαρακτηριστικό θα παρουσιαστούν στη συνέχεια αφού πρώτα παρουσιαστεί η δομή δεδομένων του στον παρακάτω πίνακα:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Ονομασία	Περιγραφή
00	4	Κωδικός χαρακτηριστικού	0x30 00 00 00
04	4	Μήκος του χαρακτηριστικού	Μήκος σε bytes
08	1	Σημαία μη τοπικού περιεχομένου (Content Non-resident Flag)	•0x00 = τοπικό περιεχόμενο (Resident) •0x01 = μη τοπικό περιεχόμενο (Non-resident)
09	1	Μήκος της ονομασίας ροής (Stream Name)	Αριθμός χαρακτήρων Unicode
0A	2	Θέση της ονομασίας ροής (Stream Name)	Σε αριθμό bytes από την αρχή του χαρακτηριστικού
0C	2	Σημαίες χαρακτηριστικού (Flags)	•0x0001=συμπιεσμένο •0x4000=κρυπτογραφημένο •0x8000=αραιό
0E	2	Αναγνωριστικό χαρακτηριστικού	Αύξων αριθμός χαρακτηριστικού που προστέθηκε στην εγγραφή με διαδοχική σειρά
10	4	Μέγεθος του περιεχομένου	Μέγεθος σε byte
14	2	Θέση του περιεχομένου	Συναρτήσει της αρχής του χαρακτηριστικού, μετρημένη σε byte
		Ονομασία Ροής (Stream Name)	Εάν Υπάρχει
00	6	Αριθμός εγγραφής αρχείου $MFT του γονικού καταλόγου	Ο αριθμός εγγραφής αρχείου στο $MFT για τον κατάλογο εντός του οποίου βρίσκεται το αρχείο
06	2	Αριθμός διαδοχής του γονικού καταλόγου (Sequence Number)	Πόσες φορές η εγγραφή αρχείου του γονικού καταλόγου έχει χρησιμοποιηθεί
08	8	Χρονοσφραγίδα δημιουργίας ονομασίας αρχείου	Μη αξιόπιστες χρονοσφραγίδες, ανανεώνονται μόνο σε περίπτωση μετονομασίας ή μετακίνησης του αρχείου
10	8	Χρονοσφραγίδα τροποποίησης ονομασίας αρχείου	
18	8	Χρονοσφραγίδα τροποποίησης εγγραφής $MFT	
20	8	Χρονοσφραγίδα τελευταίας προσπέλασης	

28	8	Φυσικό μέγεθος (Allocated Size)	Αξιόπιστα μόνο σε περίπτωση καταλόγου
30	8	Λογικό μέγεθος (Actual Size)	
38	4	Σημαίες τύπων αρχείου	Βλ. σχετικό πίνακα παραπάνω
3C	4	Τιμή Reparse	Χρησιμοποιείται από το χαρακτηριστικό Reparse
40	1	Μέγεθος ονόματος αρχείου	Αφορά αριθμό bytes για χαρακτήρες Unicode
41	1	Τύπος ονόματος αρχείου (Namespace)	•0x00=POSIX •0x01=Win32 (LFN) •0x02=DOS (SFN) •0x03=Win32 & DOS
42	Ποικίλλει	Όνομα αρχείου	
Δομή δεδομένων χαρακτηριστικού $File_Name			

Πληροφορίες Γονικού Καταλόγου (Parent $MFT Record & Sequence Number)

Το στοιχεία που παρατίθενται στο χαρακτηριστικό $File_Name και αφορούν τον γονικό κατάλογο στον οποίο περιέχεται το αρχείο, είναι ο ένας από τους δύο τρόπος που παρέχονται από το Σύστημα Αρχείων NTFS για να επιτευχθεί η οικοδόμηση της ιεραρχικής δομής των φακέλων στο Σύστημα. Αυτή η λειτουργικότητα που προσφέρει το χαρακτηριστικό $File_Name εισάγει την έννοια των ορφανών αρχείων και το καθιστά άκρως σημαντικό στη διαδικασία ανάκτησης δεδομένων της ψηφιακής εγκληματολογικής έρευνας.

Ορφανά Αρχεία (Orphan Files)

Ο πίνακας $MFT αποτελεί μία λίστα όλων των αρχείων που υπάρχουν στον τόμο. Όταν ένας κατάλογος διαγράφεται με το περιεχόμενό του, μπορεί εκ νέου να ανακτηθεί και ολόκληρη η ιεραρχική του δομή να ανακατασκευαστεί, μέχρις ότου οι εγγραφές αρχείων του να χρησιμοποιηθούν ξανά για την αποθήκευση κάποιου άλλου αρχείου.

Το Σύστημα Αρχείων NTFS χρησιμοποιεί έναν αλγόριθμο για την πλήρωση θέσεων στον πίνακα $MFT ώστε να επιτυγχάνεται αυξημένη αποδοτικότητα και να αποφεύγεται ο κατακερματισμός του. Αυτός ο αλγόριθμος λειτουργεί με βάση την προσέγγιση “top down - first available” η οποία αναγκάζει το Σύστημα να ελέγχει όλον τον πίνακα από την αρχή του, κάθε φορά που χρειάζεται να αποθηκεύσει ένα αρχείο στον τόμο, και να χρησιμοποιεί την πρώτη μη χρησιμοποιούμενη εγγραφή που βρίσκει για την αποθήκευση της νέας εγγραφής αρχείου. Εάν δεν υπάρχει διαθέσιμη ελεύθερη εγγραφή από τις ήδη υπάρχουσες μόνο τότε το Σύστημα δημιουργεί μία νέα στο τέλος του πίνακα. Αυτή η προσέγγιση έχει ως αποτέλεσμα πολλές φορές η εγγραφή αρχείου ενός γονικού καταλόγου να επαναγράφεται ενώ οι εγγραφές των αρχείων – παιδιών του να παραμένουν στον πίνακα $MFT.

Κατά τη διενέργεια μίας ψηφιακής εγκληματολογικής εξέτασης, είναι δυνατό κάποια αρχεία να μπορούν να ανακτηθούν χωρίς το ίδιο να συμβαίνει για τους γονικούς τους καταλόγους. Τα αρχεία αυτά ονομάζονται ορφανά καθώς η σχέση γονιού-παιδιού έχει χαθεί. Τα ορφανά αρχεία μπορούν να ανακτηθούν πλήρως αλλά δεν μπορούν να συσχετιστούν με τους γονικούς τους καταλόγους και συνεπώς να επανατοποθετηθούν σε μία ιεραρχική πυραμίδα δομής. Η διαδικασία διαμόρφωσης δίσκου είναι μία ενέργεια που κατεξοχήν δημιουργεί ορφανά αρχεία.

Επαλήθευση Γονικής Σχέσης

Στην επαλήθευση της γονικής σχέσης μεταξύ καταλόγου και αρχείου καταλυτική θέση κατέχει το χαρακτηριστικό $File_Name της εγγραφής. Το χαρακτηριστικό αυτό φέρει τον αριθμό εγγραφής $MFT του γονικού καταλόγου και τον αριθμό διαδοχής του γονικού καταλόγου του αρχείου. Ο αριθμός διαδοχής του ίδιου του γονικού καταλόγου και η κατάσταση χρήσης της εγγραφής του χρησιμοποιούνται για να διαπιστωθεί εάν το περιεχόμενο της εγγραφής περιγράφει ακόμα το γονικό κατάλογο ή εάν η εγγραφή αρχείου έχει επαναγραφεί. Υπάρχουν τέσσερα πιθανά σενάρια:

- Οι αριθμοί διαδοχής ταιριάζουν = Σε αυτή την περίπτωση η γονική εγγραφή είναι ακόμα κατανεμημένη και η σχέση γονέα-παιδιού μεταξύ των δύο εγγραφών βρίσκεται σε ισχύ.
- Ο αριθμός διαδοχής στη γονική εγγραφή είναι μεγαλύτερος κατά μία μονάδα (του αριθμού που δίνει το χαρακτηριστικό File_Name του αρχείου-παιδιού για αυτόν) και η κατάσταση χρήσης της εγγραφής του είναι 0x02 «διαγραμμένος κατάλογος» = Η εγγραφή αρχείου του γονικού καταλόγου έχει διαγραφεί αλλά δεν έχει επαναχρησιμοποιηθεί. Η σχέση γονέα-παιδιού μεταξύ των δύο εγγραφών βρίσκεται σε ισχύ.
- Ο αριθμός διαδοχής στη γονική εγγραφή είναι μεγαλύτερος κατά μία μονάδα (του αριθμού που δίνει το χαρακτηριστικό File_Name του αρχείου-παιδιού για αυτόν) και η κατάσταση χρήσης της εγγραφής του είναι 0x01 «κατανεμημένο αρχείο» ή 0x03 «κατανεμημένος κατάλογος» = Η εγγραφή αρχείου του γονικού καταλόγου έχει διαγραφεί και έχει επαναχρησιμοποιηθεί. Η σχέση γονέα-παιδιού μεταξύ των δύο εγγραφών δεν βρίσκεται σε ισχύ. Το αρχείο χαρακτηρίζεται ορφανό.
- Ο αριθμός διαδοχής στη γονική εγγραφή είναι μεγαλύτερος κατά περισσότερων της μίας μονάδας = Η εγγραφή αρχείου του γονικού καταλόγου έχει διαγραφεί και έχει επαναχρησιμοποιηθεί πολλές φορές. Η σχέση γονέα-παιδιού μεταξύ των δύο εγγραφών δεν βρίσκεται σε ισχύ. Το αρχείο χαρακτηρίζεται ορφανό.

Τύπος Ονόματος Αρχείου (File Name NameSpace)

Με τις πολλαπλές εκδόσεις Λειτουργικών Συστημάτων της Microsoft, οι κανόνες για την ονομασία των αρχείων έχει αλλάξει. Αναλύοντας τους τύπους POSIX και Win32 θα διαπιστώσουμε ότι ένα όνομα αρχείου μπορεί να περιέχει ως 254 χαρακτήρες κεφαλαίων ή μικρών γραμμάτων γι' αυτό και αυτά τα συστήματα χαρακτηρίζονται ως LFN (Long File Names). Μπορεί ακόμα να περιέχει ειδικούς χαρακτήρες εκτός κάποιων κατειλημμένων (/, \, :, *, ?, “, <, >, |, .).

Η διαφορά μεταξύ των δύο τύπων είναι ότι το POSIX αναγνωρίζει την κεφαλαιοποίηση των χαρακτήρων ενώ το Win32 όχι. Για παράδειγμα τα “File” και “file” λαμβάνονται ως δύο διαφορετικά ονόματα αρχείων και μπορούν να συνυπάρξουν εντός του ίδιου καταλόγου για το σύστημα ονομασίας POSIX αλλά όχι για το Win32.

Στις παλαιότερες εκδόσεις Λειτουργικών τα ονόματα των αρχείων έπρεπε να είναι σύντομα (Short File Names) και χρησιμοποιούσαν τον κανόνα ονοματοδοσίας MS-DOS 8.3. Σύμφωνα με αυτόν επιτρεπόταν η χρήση μέχρι οκτώ κεφαλαίων γραμμάτων για το όνομα του αρχείου και τριών κεφαλαίων γραμμάτων για την επέκταση του αρχείου. Το NTFS επέτρεψε αναχρονιστική συμβατότητα για παλαιότερες εφαρμογές και προγράμματα χρησιμοποιώντας πολλαπλά χαρακτηριστικά $File_Name εάν η ονομασία ενός αρχείου δεν ήταν συμβατή με τον ανωτέρω κανόνα.

Όταν ένα SFN όνομα δημιουργείται από το NTFS για την επίτευξη αναχρονιστικής συμβατότητας, οι πρώτοι 6 χαρακτήρες του LFN ονόματος χρησιμοποιούνται, μετατρέπονται σε κεφαλαίοι και ακολούθως ο χαρακτήρας "~" προσαρτάται μαζί με έναν αριθμό. Επίσης μόνο τα τρία πρώτα γράμματα της επέκτασης χρησιμοποιούνται. Σε περίπτωση δημιουργίας πανομοιότυπων ονομάτων κατά την πραγματοποίηση της προηγούμενης διαδικασίας το Σύστημα Αρχείων θα κρατήσει μόνο τους δύο πρώτους χαρακτήρες του ονόματος και στη συνέχεια τέσσερα γράμματα θα παραχθούν τυχαία με μία μαθηματική αλγοριθμική διαδικασία. Ο αριθμός ~1 θα προστεθεί στο τέλος ή άλλος εάν αυτό χρειάζεται για να αποφευχθεί εκ νέου πρόβλημα πανομοιότυπης ονομασίας.

Εάν ένα όνομα συμμορφώνεται με τον κανόνα MS-DOS 8.3 αλλά περιέχει και μικρά γράμματα αντί μόνο κεφαλαίων, θεωρείται MS-DOS συμβατό όνομα και εμπίπτει στην κατηγορία 0x03 (Win32 & DOS). Η μετατροπή από μικρά σε κεφαλαία πραγματοποιείται με τη χρήση του αρχείου μεταδεδομένων $UpCase.

Μία εγγραφή αρχείου μπορεί να φέρει πολλαπλά χαρακτηριστικά $File_Name είτε για να επιτύχει αναχρονιστική αναδρομή ή ως τεχνική εξοικονόμησης χώρου στο δίσκο, γεγονός που εισαγάγει την έννοια του συνδέσμου Hard Link.

Σύνδεσμοι Hard Links

Ένας σύνδεσμος Hard Link έχει παρόμοια φύση με τα αρχεία LNK των Windows όσον αφορά τη λειτουργικότητα τους ως δείκτες (Pointers). Ο σύνδεσμος Hard Link επιτρέπει σε ένα αρχείο να υπάρχει σε περισσότερους του ενός γονικούς καταλόγους την ίδια στιγμή που μόλις ένα αντίγραφο του περιεχομένου του αρχείου υπάρχει αποθηκευμένο στο μέσο. Κάθε σύνδεσμος περιέχεται μέσα στο δικό του χαρακτηριστικό $File_Name και περιέχει τα δικά του ξεχωριστά στοιχεία πληροφορίας γονικού καταλόγου (Parent $MFT Record & Sequence Number). Με αυτόν τον τρόπο είναι δυνατή η δόμηση της ιεραρχίας εντός του Συστήματος για κάθε ένα από τα αρχεία αυτά, ενώ όλα μοιράζονται και δείχνουν στο ίδιο χαρακτηριστικό με αποτέλεσμα να μην κατασπαταλάται χώρος στο μέσο για την αποθήκευση της ίδιας πληροφορίας πολλαπλές φορές. Ένα αρχείο διαγράφεται μόνο όταν και ο τελευταίος σύνδεσμος τουHard Link διαγραφεί.

Η απολύτως απαραίτητη πληροφόρηση που χρειάζεται ένας εξεταστής για την επιτυχή εγκληματολογική ανάκτηση ενός αρχείου, ολοκληρώνεται με την ανάλυση του χαρακτηριστικού $Data.

2.2.8 Το Χαρακτηριστικό $Data

Όλα τα υπόλοιπα χαρακτηριστικά που εξετάσαμε μέχρι τώρα παρέχουν μεταδεδομένα για το αρχείο και πληροφόρηση για τη θέση του στο μέσο, τις ιδιότητες του κλπ. Το χαρακτηριστικό $Data, σε αντίθεση με όλα τα άλλα χαρακτηριστικά, περιέχει τα πραγματικά δεδομένα του αρχείου ή δείκτες για το που αυτά τα δεδομένα βρίσκονται στο μέσο. Συνεπώς αυτό το χαρακτηριστικό ταυτίζεται ουσιαστικά με το ίδιο το αρχείο.

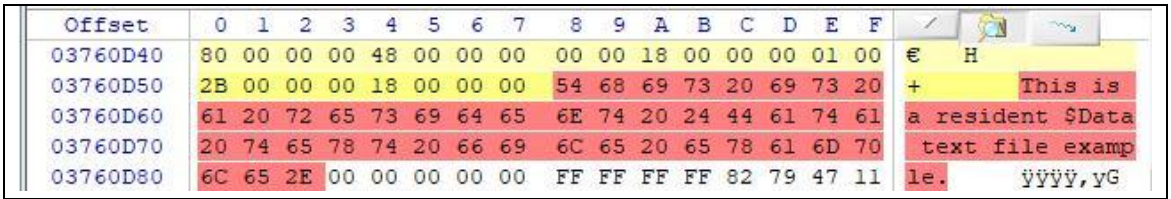
Το Σύστημα Αρχείων NTFS είναι σχεδιασμένο έτσι ώστε να εντοπίζει το αρχείο κατόπιν αίτησης του Λειτουργικού Συστήματος ή κάποιας εφαρμογής και να το προστατεύει από αλλοίωση (corruption) και παραβιάσεις ασφάλειας. Η δομή του χαρακτηριστικού διαφέρει ανάλογα με το αν το περιεχόμενό του είναι τοπικό ή όχι.

Χαρακτηριστικό με τοπικό περιεχόμενο (Content Resident Attribute)

Εάν το περιεχόμενο του χαρακτηριστικού είναι τοπικό τότε χρησιμοποιείται μόνο η δομή της επικεφαλίδας χαρακτηριστικού και της επικεφαλίδας τοπικού χαρακτηριστικού. Σε αυτή την περίπτωση το τοπικό περιεχόμενο του χαρακτηριστικού αποτελεί και τα δεδομένα του αρχείου. Μόνο πολύ μικρά αρχεία έχουν το χαρακτηριστικό $Data τοπικό, συνήθως αρχεία μεγέθους 600 μέχρι 900 bytes, αναλόγως του μεγέθους του ονόματος του αρχείου και της ύπαρξης λοιπών χαρακτηριστικών στην εγγραφή αρχείου. Τα αρχεία περιήγησης διαδικτύου cookies είναι χαρακτηριστικό παράδειγμα αρχείων με τοπικό χαρακτηριστικό $Data. Παρακάτω παρατίθεται η δομή δεδομένων ενός τοπικού χαρακτηριστικού $Data:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Ονομασία	Περιγραφή
00	4	Κωδικός χαρακτηριστικού	0x80 00 00 00
04	4	Μήκος του χαρακτηριστικού	Μήκος σε bytes
08	1	Σημαία μη τοπικού περιεχομένου (Content Non-resident Flag)	•0x00 = τοπικό περιεχόμενο (Resident) •0x01 = μη τοπικό περιεχόμενο (Non-resident)
09	1	Μήκος της ονομασίας ροής (Stream Name)	Αριθμός χαρακτήρων Unicode
0A	2	Θέση της ονομασίας ροής (Stream Name)	Σε αριθμό bytes από την αρχή του χαρακτηριστικού
0C	2	Σημαίες χαρακτηριστικού (Flags)	•0x0001=συμπιεσμένο •0x4000=κρυπτογραφημένο •0x8000=αραιό
0E	2	Αναγνωριστικό χαρακτηριστικού	Αύξων αριθμός χαρακτηριστικού που προστέθηκε στην εγγραφή με διαδοχική σειρά
10	4	Μέγεθος του περιεχομένου	Μέγεθος σε byte
14	2	Θέση του περιεχομένου	Συναρτήσει της αρχής του χαρακτηριστικού, μετρημένη σε byte
		Ονομασία Ροής (Stream Name)	Εάν Υπάρχει
18	Ποικίλλει	Τοπικό περιεχόμενο χαρακτηριστικού	Δεδομένα του αρχείου
Δομή δεδομένων τοπικού χαρακτηριστικού $Data			

Στην παρακάτω εικόνα παρουσιάζεται ένα αρχείο txt με τοπικό χαρακτηριστικό $Data. Οι επικεφαλίδες χαρακτηριστικού και τοπικού περιεχομένου έχουν σημανθεί με απαλό και έντονο κίτρινο χρώμα αντίστοιχα. Το περιεχόμενο του αρχείου έχει σημανθεί με κόκκινο χρώμα. Παρατηρείται το γέμισμα (padding) που πραγματοποίησε το Σύστημα ώστε να τηρηθεί το όριο θέσης του πολλαπλάσιου των 8 byte, καθώς στις θέσεις 0x03760D83 έως και 0x03760D87 το Σύστημα έχει γεμίσει το περιεχόμενο με την δεκαεξαδική τιμή 0x00. Κατόπιν ακολουθεί η τιμή 0xFF FF FF FF που σηματοδοτεί το τέλος της εγγραφής αρχείου αφού το χαρακτηριστικό $Data στην παρούσα εγγραφή τυχαίνει να είναι και το τελευταίο της.



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
03760D40	80	00	00	00	48	00	00	00	00	00	18	00	00	00	01	00	€ H
03760D50	2B	00	00	00	18	00	00	00	54	68	69	73	20	69	73	20	+ This is
03760D60	61	20	72	65	73	69	64	65	6E	74	20	24	44	61	74	61	a resident $Data
03760D70	20	74	65	78	74	20	66	69	6C	65	20	65	78	61	6D	70	text file examp
03760D80	6C	65	2E	00	00	00	00	00	FF	FF	FF	FF	82	79	47	11	le. ÿÿÿÿ,yG

Τοπικό χαρακτηριστικό $Data

Διακρίνονται οι ακόλουθες τιμές:

Επικεφαλίδα Χαρακτηριστικού

- Κωδικός χαρακτηριστικού – 0x80 00 00 00 = $Data
- Μήκος χαρακτηριστικού – 0x48 (Little Endian) = 72 bytes
- Σημαία μη τοπικού περιεχομένου – 0x00 = περιεχόμενο τοπικό
- Μήκος της ονομασίας ροής – 0x00 = δεν υπάρχει Name Stream
- Σημαίες χαρακτηριστικού – 0x00 = δεν έχουν τεθεί
- Αναγνωριστικό χαρακτηριστικού – 0x01 =δεύτερο χαρακτηριστικό που προστέθηκε στην εγγραφή αρχείου

Τοπική Επικεφαλίδα

- Μέγεθος περιεχομένου – 0x2B = 43 bytes
- Θέση περιεχομένου – 0x18 = 24 bytes (από την αρχή του χαρακτηριστικού)

Τοπικό Περιεχόμενο

- «This is a resident $Data text file example. »

Η ανάκτηση ενός αρχείου με τοπικό χαρακτηριστικό $Data απλά ταυτίζεται με την εύρεση της εγγραφής αρχείου του στον πίνακα $MFT. Εάν το αρχείο είναι διαγεγραμμένο τότε δύναται να ανακτηθεί μόνο μέχρι τη χρονική στιγμή που η εγγραφή του δεν έχει επαναχρησιμοποιηθεί από το Σύστημα για την αποθήκευση άλλου αρχείου. Σε διαφορετική περίπτωση η νέα εγγραφή αρχείου διαγράφει όλο τον προηγούμενο κατανεμημένο χώρο στην καταχώριση και συνεπώς το αρχείο καθίσταται μη ανακτήσιμο.

Χαρακτηριστικό με Μη Τοπικό Περιεχόμενο (Content Non Resident Attribute)

Μία εγγραφή αρχείου είναι σταθερού μεγέθους (1024 bytes στην παρούσα έκδοση NTFS) και περιέχει πάντα τουλάχιστον τρία χαρακτηριστικά. Κάποιες φορές το περιεχόμενο κάποιου χαρακτηριστικού είναι τόσο μεγάλο που δεν μπορεί να χωρέσει στον περιορισμένο χώρο που παρέχει η εγγραφή αρχείου. Όταν αυτό συμβεί, το περιεχόμενο αυτού του χαρακτηριστικού μετακινείται σε άλλο μέρος του τόμου και αποκαλείται μη τοπικό χαρακτηριστικό. Παρόλαυτα η επικεφαλίδα του περιεχομένου παραμένει στην εγγραφή αρχείου αφού μόνο το περιεχόμενό του μετακινείται, οπότε ουσιαστικά πρόκειται για ένα χαρακτηριστικό με μη τοπικό περιεχόμενο.

Από τη στιγμή που το περιεχόμενο ενός χαρακτηριστικού γίνει μη τοπικό, δεν υπάρχει περίπτωση να ξαναγίνει τοπικό ποτέ. Αυτό σημαίνει ότι εάν ποτέ το περιεχόμενο συρρικνωθεί σε τέτοιο βαθμό που η εγγραφή αρχείου θα μπορούσε και πάλι να το φιλοξενήσει ολόκληρο, αυτό δε συμβαίνει αντιθέτως το περιεχόμενο παραμένει μη τοπικό. Μοναδική εξαίρεση σε αυτόν τον κανόνα αποτελεί η περίπτωση που ένας τόμος διαμορφωμένος με το Σύστημα Αρχείων NTFS συνδεθεί σε ένα Λειτουργικό Σύστημα Linux. Τότε, λόγω του τρόπου που οι οδηγοί του NTFS (drivers) έχουν ρυθμιστεί για το Linux, ένα μη τοπικό περιεχόμενο μπορεί να επανέλθει στην εγγραφή αρχείου και να γίνει πάλι τοπικό.

Στην περίπτωση του μη τοπικού χαρακτηριστικού, μία επικεφαλίδα μη τοπικού περιεχομένου θα ακολουθεί την επικεφαλίδα χαρακτηριστικού. Η δομή δεδομένων της επικεφαλίδας μη τοπικού περιεχομένου δείχνει το μέγεθος του περιεχομένου (το οποίο κατ' επέκταση ταυτίζεται με το μέγεθος του αρχείο) και την τοποθεσία που βρίσκεται αυτό εντός του τόμου, πληροφορία που αποκαλείται data runs και συνήθως αποθηκεύεται ως μία λίστα συνεχόμενων τιμών. Στον παρακάτω πίνακα παρουσιάζεται η δομή δεδομένων του μη τοπικού χαρακτηριστικού και στη συνέχεια αναλύονται τα βασικά του χαρακτηριστικά:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Ονομασία	Περιγραφή
00	4	Κωδικός χαρακτηριστικού	0x80 00 00 00
04	4	Μήκος του χαρακτηριστικού	Μήκος σε bytes
08	1	Σημαία μη τοπικού περιεχομένου (Content Non-resident Flag)	•0x00 = τοπικό περιεχόμενο (Resident) •0x01 = μη τοπικό περιεχόμενο (Non-resident)
09	1	Μήκος της ονομασίας ροής (Stream Name)	Αριθμός χαρακτήρων Unicode
0A	2	Θέση της ονομασίας ροής (Stream Name)	Σε αριθμό bytes από την αρχή του χαρακτηριστικού
0C	2	Σημαίες χαρακτηριστικού (Flags)	•0x0001=συμπιεσμένο •0x4000=κρυπτογραφημένο •0x8000=αραιό
0E	2	Αναγνωριστικό χαρακτηριστικού	Αύξων αριθμός χαρακτηριστικού που προστέθηκε στην εγγραφή με διαδοχική σειρά
10	8	Αριθμός έναρξης εικονικού cluster (VCN) της λίστας data run	Virtual Cluster Number – σε σχέση με την αρχή των ίδιων των δεδομένων στο αρχείο
18	8	Αριθμός λήξης εικονικού cluster (VCN) της λίστας data run	
20	2	Θέση της λίστας data runs	Αριθμός byte σε συνάρτηση με την αρχή του χαρακτηριστικού
22	2	Χρησιμοποιείται για συμπίεση	~
24	4	Μη χρησιμοποιούμενο	~
28	8	Κατανεμημένο μέγεθος	Φυσικό μέγεθος του περιεχομένου σε bytes
30	8	Πραγματικό μέγεθος	Λογικό μέγεθος του περιεχομένου σε bytes
38	8	Αρχικοποιημένο μέγεθος	Αρχικοποιημένο μέγεθος του περιεχομένου σε bytes
~	Ποικίλλει	Ονομασία Ροής (Stream Name)	Εάν Υπάρχει
~	Ποικίλλει	Λίστα data runs	Δεδομένα του αρχείου
Δομή δεδομένων μη τοπικού χαρακτηριστικού $Data			

Αρίθμηση Εικονικών Clusters (Virtual Cluster Numbering)

Όταν ένας τόμος διαμορφώνεται με το Σύστημα Αρχείων (NTFS), οι τομείς του ομαδοποιούνται σε γκρουπ διαδοχικών τομέων που ονομάζονται clusters. Σε κάθε cluster ανατίθεται μια διεύθυνση σχετική με την τοποθεσία του εντός του τόμου, η οποία ονομάζεται λογικός αριθμός cluster (Logical Cluster Number - LCN).

Τα clusters μπορούν επίσης να παρατηρηθούν και να αριθμηθούν σε σχέση με ένα αρχείο. Σε κάθε cluster που κατανέμεται σε ένα αρχείο, ανατίθεται ένας αριθμός σχετικός με την θέση του στο αρχείο ο οποίος ονομάζεται (Virtual Cluster Number - VCN). Η εικονική αρίθμηση αναθέτει εκ νέου αριθμό σε κάθε cluster που κατανέμεται σε ένα ενεργό αρχείο.

Είναι συχνό φαινόμενο ένα μεγάλο αρχείο να μη μπορεί να αποθηκευτεί σε διαδοχικά clusters στον τόμο. Όταν αυτό συμβεί το αρχείο κατακερματίζεται και γράφεται σε διαφορετικά κομμάτια του τόμου (fragments). Ο εικονικός αριθμός cluster που ανατίθεται σε κάθε cluster που το αρχείο καταλαμβάνει, επιτρέπει στο αρχείο να ανοικοδομείται με τη σωστή σειρά ανεξαρτήτως της σειράς των λογικών αριθμών cluster που καταλαμβάνει κατά την αποθήκευσή του.

Αυτό είναι ακόμα πιο σημαντικό σε αρχεία που είναι αρκετά κατακερματισμένα. Δεν είναι ασυνήθιστο για μία εγγραφή αρχείου να είναι πολύ μικρή για να περιέχει ακόμα και μία ολοκληρωμένη λίστα data runs του περιεχομένου του αρχείου. Σε αυτή την περίπτωση θα υπάρξουν επιπλέον εγγραφές αρχείων στο $MFT οι οποίες θα περιέχουν χαρακτηριστικά $Data τα οποία θα περιγράφουν τις εναπομείνασες λίστες data runs. Όταν αυτές οι λίστες αναλύονται το Σύστημα πρέπει να γνωρίζει σε πιο κομμάτι του αρχείου δείχνει η κάθε λίστα, ώστε να τοποθετηθούν στη κατάλληλη σειρά, γεγονός που το πετυχαίνει με την ανάλυση των VCN.

Τα VCNs αντιπροσωπεύουν τη θέση (σε clusters) σε σχέση με την αρχή του αρχείου και παρέχουν μία χαρτογράφηση της αποθήκευσής του στο μέσο. Στο παρακάτω σχήμα περιγράφεται η εικονική αρίθμηση cluster:

LCN 0	LCN 1	LCN 2	LCN 3	LCN 4 VCN 3	LCN 5 VCN 4	LCN 6 VCN 5	LCN 7 VCN 6
LCN 8 VCN 7	LCN 9 VCN 8	LCN 10 VCN 9	LCN 11 VCN 10	LCN 12 VCN 11	LCN 13 VCN 12	LCN 14 VCN 13	LCN 15 VCN 14
LCN 16 VCN 0	LCN 17 VCN 1	LCN 18 VCN 2	LCN 19 VCN 3	LCN 20 VCN 4	LCN 21	LCN 22	LCN 23
LCN 24	LCN 25	LCN 26 VCN 0	LCN 27 VCN 1	LCN 28 VCN 2	LCN 29	LCN 30	LCN 31
Λογική αρίθμηση cluster							

Στο παραπάνω σχήμα φαίνονται αποθηκευμένα δύο αρχεία.

- Το μπορντό αρχείο είναι κατακερματισμένο. Το πρώτο του κομμάτι βρίσκεται αποθηκευμένο στους λογικούς αριθμούς clusters 26 – 28 LCN για τον τόμο, που αντίστοιχα αποτελούν τους εικονικούς αριθμούς clusters 0 – 2 VCN για το ίδιο το αρχείο. Το δεύτερο κομμάτι του είναι αποθηκευμένο στους λογικούς clusters 4 – 15 LCN οι οποίοι αποτελούν τους εικονικούς clusters 3 – 14 VCN
- Το γαλάζιο αρχείο είναι συνεχές. Ξεκινάει στους λογικούς αριθμούς clusters 16 – 20 LCN για τον τόμο, που αντίστοιχα αποτελούν τους εικονικούς αριθμούς clusters 0 – 4 VCN για το ίδιο το αρχείο.

Κατανεμημένο μέγεθος (Allocated Size)

Το μέγεθος των cluster σε ένα τόμο είναι προκαθορισμένο κατά τη διαμόρφωση του. Ένα cluster μπορεί να κατανεμηθεί μόνο σε ένα αρχείο. Ο αριθμός των clusters που ένα αρχείο χρειάζεται για να αποθηκευτεί ονομάζεται κατανεμημένο μέγεθος του αρχείου, αποκαλούμενο συνηθέστερα ως φυσικό μέγεθος του αρχείου (physical size).

Πραγματικό μέγεθος (Allocated Size)

Το μέγεθος του πραγματικού περιεχομένου του αρχείου σε bytes αποκαλείται πραγματικό μέγεθος ή λογικό (logical size). Το μέγεθος των περισσοτέρων αρχείων δεν πρόκειται να είναι διαιρέσιμο ακριβώς με το μέγεθος των clusters και συνεπώς αυτή η διαφορά θα δημιουργεί ένα ελεύθερο χώρο μεταξύ του τέλους του λογικού αρχείου και του κατανεμημένου για αυτό χώρου στο μέσο. Αυτή η διαφορά εισάγει την έννοια του χώρου υπολείμματος (slack space) η οποία θα αναλυθεί εκτενώς στο κεφάλαιο 2.4.

Αρχικοποιημένο μέγεθος (Initialized Size)

Κάποιες φορές όταν ένα αρχείο δημιουργείται στον τόμο δεν είναι διαθέσιμα όλα τα δεδομένα του, όπως για παράδειγμα όταν ένα αρχείο κάτω-φορτώνεται (download) από το διαδίκτυο. Σε αυτές τις περιπτώσεις καταλαμβάνεται χώρος στον τόμο από το Σύστημα Αρχείων χωρίς να εγγράφονται πραγματικά δεδομένα σε αυτόν. Αυτός ο χώρος αποκαλείται αρχικοποιημένο μέγεθος του αρχείου και η ύπαρξή του αποτρέπει τις περιττές ενέργειες εγγραφής/ανάγνωσης βελτιώνοντας κατά αυτόν τον τρόπο την απόδοση του Συστήματος.

Λίστες Data Runs (Data Run Lists)

Μία λίστα data run (ή αλλιώς run list) είναι μία συλλογή δεικτών προς το περιεχόμενο του χαρακτηριστικού $Data. Κάθε ξεχωριστός δείκτης αποκαλείται καταχώριση run list (run list entry) και δείχνει προς ένα μπλοκ συνεχούς περιεχομένου (fragment). Το επιτυγχάνει αυτό με το να παρέχει το μήκος του fragment σε clusters και την θέση έναρξής του στον τόμο.

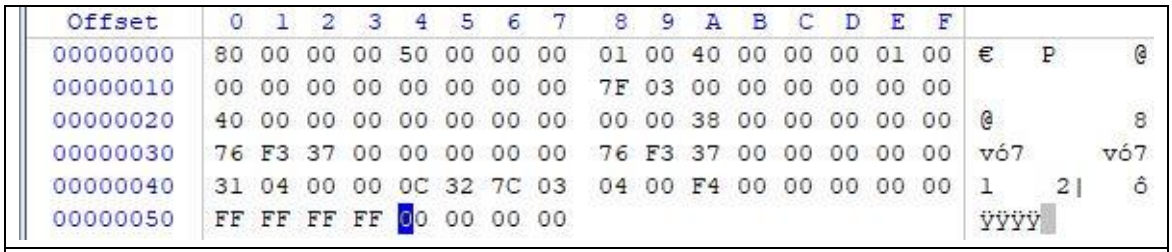
Κάθε καταχώριση στη run list έχει επικεφαλίδα και περιεχόμενο. Η επικεφαλίδα έχει μήκος ένα byte και δείχνει το μέγεθος του περιεχομένου χρησιμοποιώντας κάθε nibble (4 bits) της για να αναπαραστήσει τον αριθμό των byte που θα χρησιμοποιηθούν για τα πεδία του μήκος του fragment και της θέσης έναρξης. Δηλαδή

- Αριστερό nibble και δεξί nibble επικεφαλίδας = αριθμός bytes του περιεχομένου της καταχώρισης
- Το αριστερό nibble της επικεφαλίδας αντιπροσωπεύει πόσα bytes χρησιμοποιεί το πεδίο για την θέση έναρξης του fragment
- Το δεξί nibble της επικεφαλίδας αντιπροσωπεύει πόσα bytes χρησιμοποιεί το πεδίο για το μήκος του fragment

Ένα πολύ σημαντικό στοιχείο για την ορθή ερμηνεία των run lists είναι η σχετικότητα της θέσης έναρξης. Μόνο το πρώτο run list έχει τιμή που σχετίζεται με τα λογικά clusters του τόμου. Οι υπόλοιπες θέσεις έναρξης δίνονται σε σχέση με την θέση έναρξης του προηγούμενου run list κατά σειρά, δηλαδή κάθε προηγούμενη θέση έναρξης πρέπει να προστεθεί στις επόμενες ώστε να αποκτηθεί ο λογικός αριθμός cluster LCN για τη θέση έναρξης του τρέχοντος fragment. Αυτή η τεχνική επιτυγχάνει μείωση του μεγέθους των καταχωρίσεων σε πολύ μεγάλους τόμους για τη χρήση των οποίων το NTFS είναι πλήρως βελτιστοποιημένο. Το τέλος μίας run list σηματοδοτείται από μία επικεφαλίδα με την δεκαεξαδική τιμή 0x00.

Είναι επίσης πολύ σημαντικό να τονιστεί ότι η τιμή της θέσης έναρξης είναι ένας Signed Integer και συνεπώς μπορεί να έχει τόσο θετικές όσο και αρνητικές τιμές για να επιτυγχάνει την

πλήρη περιγραφή του τόμου. Όλα τα παραπάνω θα αναλυθούν στο επόμενο παράδειγμα όπου ένα μη τοπικό χαρακτηριστικό $Data απεικονίζεται:



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	80	00	00	00	50	00	00	00	01	00	40	00	00	00	01	00	€ P @
00000010	00	00	00	00	00	00	00	00	7F	03	00	00	00	00	00	00	
00000020	40	00	00	00	00	00	00	00	00	00	38	00	00	00	00	00	@ 8
00000030	76	F3	37	00	00	00	00	00	76	F3	37	00	00	00	00	00	vó7 vó7
00000040	31	04	00	00	0C	32	7C	03	04	00	F4	00	00	00	00	00	1 2| ô
00000050	FF	FF	FF	FF	00	00	00	00									ÿÿÿÿ

Μη τοπικό χαρακτηριστικό $Data

Διακρίνονται οι ακόλουθες τιμές:

Επικεφαλίδα Χαρακτηριστικού

- Κωδικός χαρακτηριστικού – 0x80 00 00 00 = $Data
- Μήκος χαρακτηριστικού – 0x50 (Little Endian) = 80 bytes
- Σημαία μη τοπικού περιεχομένου – 0x01 = περιεχόμενο μη τοπικό
- Μήκος της ονομασίας ροής – 0x00 = δεν υπάρχει Name Stream
- Σημαίες χαρακτηριστικού – 0x00 = δεν έχουν τεθεί
- Αναγνωριστικό χαρακτηριστικού – 0x01 =δεύτερο χαρακτηριστικό που προστέθηκε στην εγγραφή αρχείου

Μη τοπική Επικεφαλίδα

- Αρχικό VCN – 0 = πρώτο cluster του αρχείου
- Τελικό VCN – 0x37F = 895 cluster (άρα συνολικά 896 cluster κατανεμημένα στο αρχείο)
- Θέση της run list – 0x40 = 64 bytes από την αρχή του χαρακτηριστικού
- Κατανεμημένο μέγεθος – 0x38 00 00 = 3.670.016 bytes
- Πραγματικό μέγεθος – 0x37 F3 76 = 3.666.806 bytes
- Αρχικοποιημένο μέγεθος – 0x37 F3 76 = 3.666.806 bytes

Η λίστα Data Runs είναι η εξής: 31 04 00 00 0C 32 7C 03 04 00 F4 00

- Πρώτη καταχώριση στη run list: Η επικεφαλίδα 0x31 δίνει το συνολικό μήκος του περιεχομένου, ήτοι 3+1 = 4 bytes. Επομένως το περιεχόμενο της πρώτης καταχώρισης είναι το 04 00 00 0C. Το δεξί nibble της επικεφαλίδας (1) δείχνει πόσα bytes χρησιμοποιούνται για το μέγεθος του fragment. Άρα στην προκειμένη πρόκειται για ένα κομμάτι μήκους 0x04 clusters. Το αριστερό nibble (3) δείχνει πόσα bytes χρησιμοποιούνται για το μέγεθος της θέσης έναρξης. Άρα το fragment ξεκινάει στη θέση 0x0C 00 00 που αντιστοιχεί στο λογικό cluster 786.432 (από την αρχή του τόμου). Συνεπώς έχουμε

Μήκος (clusters)	θέση έναρξης	Αρχικό LCN	Τελικό LCN
4	786.432	786.432	786.435

Πρώτη καταχώριση στη λίστα Data runs

- Δεύτερη καταχώριση στη run list: Η επικεφαλίδα 0x32 δίνει το συνολικό μήκος του περιεχομένου, ήτοι 3+2 = 5 bytes. Επομένως το περιεχόμενο της δεύτερης καταχώρισης είναι το 7C 03 04 00 F4. Το δεξί nibble της επικεφαλίδας (2) δείχνει πόσα bytes χρησιμοποιούνται για το μέγεθος του fragment. Άρα στην προκειμένη πρόκειται για ένα κομμάτι μήκους 0x03 7C clusters (892), λόγω της εφαρμογής της μεθόδου αποθήκευσης Little Endian. Το αριστερό nibble (3) δείχνει πόσα bytes

χρησιμοποιούνται για το μέγεθος της θέσης έναρξης. Άρα το fragment ξεκινάει στη θέση 0xF4 00 04 (-786.428) που αντιστοιχεί στο λογικό cluster 786.432 + (-786.428) = 4 (από την αρχή του τόμου). Συνεπώς έχουμε

Μήκος (clusters)	Θέση έναρξης	Αρχικό LCN	Τελικό LCN
892	-786.428	4	895
Δεύτερη καταχώριση στη λίστα Data runs			

- Τρίτη καταχώριση στη run list: Η επόμενη επικεφαλίδα φέρει την τιμή 0x00 που σηματοδοτεί το τέλος της λίστας.

2.3 Εγγραφή – Διαγραφή αρχείων στο Σύστημα Αρχείων NTFS

Οι βασικές ενέργειες που πραγματοποιεί το Σύστημα Αρχείων NTFS είναι η εγγραφή και διαγραφή αρχείων στον τόμο, σύμφωνα με τις ενέργειες που πραγματοποιεί ο χρήστης επί του μέσου και τις οδηγίες που λαμβάνει από το Λειτουργικό Σύστημα.

Και ενώ η εγγραφή ενός αρχείου στον τόμο ταυτίζεται ως έννοια με την εγγραφή των δεδομένων του σε αυτόν, δεν ισχύει το ίδιο στην περίπτωση της διαγραφής, γεγονός που αποτελεί και σε μεγάλο βαθμό την ίδια την ουσία μιας ψηφιακής εγκληματολογικής εξέτασης.

Παρατηρώντας τα βήματα που λαμβάνουν χώρα κατά την πραγματοποίηση των δύο προαναφερθέντων εννοιών μπορεί κανείς να αντιληφθεί την μέθοδο που χρησιμοποιεί το NTFS (και όλα τα άλλα Συστήματα Αρχείων) προκειμένου να επιτύχει μικρούς χρόνους απόκρισης, αποδοτικότητα και βελτιστοποίηση της εμπειρίας ενός χρήστη.

Δημιουργία Αρχείου

Όταν ένα αρχείο (ή κατάλογος) δημιουργούνται στον τόμο η ακόλουθη σειρά βημάτων λαμβάνει χώρα:

- Μία καταχώριση για την ενέργεια της εγγραφής πραγματοποιείται στο αρχείο μεταδεδομένων $Log_File
- Μία καταχώριση για την ενέργεια της εγγραφής πραγματοποιείται στο αρχείο μεταδεδομένων $USN Journal
- Μία εγγραφή αρχείου παραχωρείται για το αρχείο στον πίνακα $MFT
- Το χαρακτηριστικό $Bitmap του $MFT ανανεώνεται προκειμένου να παρουσιάσει αυτή την εγγραφή αρχείου ως κατανεμημένη
- Η σημαία κατάστασης στην επικεφαλίδα της εγγραφής αρχείου αλλάζει προκειμένου να παρουσιάσει αυτή την εγγραφή αρχείου ως κατανεμημένο αρχείο ή κατάλογο
- Τα απαραίτητα χαρακτηριστικά που απαιτούνται για να περιγράψουν πλήρως το αρχείο γράφονται στην εγγραφή
- Εάν το περιεχόμενο κάποιων χαρακτηριστικών είναι μη τοπικό τότε το αρχείο μεταδεδομένων $Bitmap ανανεώνεται προκειμένου να παρουσιάσει τα οικεία clusters ως κατανεμημένα και στη συνέχεια το Σύστημα Αρχείων εγγράφει σε αυτά τα δεδομένα
- Οι απαιτούμενοι ενταμιευτές ευρετηρίου (index buffers) δημιουργούνται ωστέ να απεικονίζουν την νέα πλέον ιεραρχική δομή των καταλόγων και των αρχείων στον γονικό κατάλογο του συστήματος.

Διαγραφή Αρχείου

Όταν ένα αρχείο (ή κατάλογος) διαγράφονται από έναν τόμο η ακόλουθη σειρά βημάτων λαμβάνει χώρα:

- Μία καταχώριση για την ενέργεια της διαγραφής πραγματοποιείται στο αρχείο μεταδεδομένων $Log_File
- Μία καταχώριση για την ενέργεια της διαγραφής πραγματοποιείται στο αρχείο μεταδεδομένων $USN Journal
- Ο αριθμός αθροίσματος διαδοχής (Sequence Count) της επικεφαλίδας της εγγραφής αρχείου στο $MFT αυξάνεται κατά μια μονάδα
- Η σημαία κατάστασης στην επικεφαλίδα της εγγραφής αρχείου αλλάζει προκειμένου να παρουσιάσει αυτή την εγγραφή αρχείου ως διαγεγραμμένο αρχείο ή κατάλογο
- Το χαρακτηριστικό $Bitmap του $MFT ανανεώνεται προκειμένου να παρουσιάσει αυτή την εγγραφή αρχείου ως μη κατανεμημένη
- Εάν το περιεχόμενο κάποιων χαρακτηριστικών που χρησιμοποιούσε το αρχείο/κατάλογος ήταν μη τοπικό τότε το αρχείο μεταδεδομένων $Bitmap ανανεώνεται προκειμένου να παρουσιάσει τα οικεία clusters ως μη κατανεμημένα
- Οι απαιτούμενοι ενταμιευτές ευρετηρίου (index buffers) διαγράφονται ωστέ να απεικονίζουν την νέα πλέον ιεραρχική δομή των καταλόγων και των αρχείων στον γονικό κατάλογο του συστήματος.

Καμία επιπλέον αλλαγή δε λαμβάνει χώρα και συνεπώς όλη η υπόλοιπη εγγραφή αρχείου παραμένει άθικτη τουλάχιστον μέχρι να επαναχρησιμοποιηθεί από το Σύστημα. Αυτό το γεγονός έχει ως αποτέλεσμα το αρχείο να είναι ανακτήσιμο αρκεί τα βήματα να αντιστραφούν.

Εάν το περιεχόμενο του αρχείου (περιεχόμενο του χαρακτηριστικού $Data) ήταν τοπικό τότε μπορεί να ανακτηθεί όσο η εγγραφή αρχείου δεν ξαναχρησιμοποιηθεί. Εάν τα δεδομένα του αρχείου ήταν μη τοπικά, τότε το περιεχόμενο του αρχείου βρίσκεται ακόμα στον μη κατανεμημένο χώρο του δίσκου και μπορεί να ανακτηθεί μέχρις ότου τα cluster που καταλάμβανε χρησιμοποιηθούν για την αποθήκευση κάποιου άλλου αρχείου. Με τα σύγχρονα αποθηκευτικά μέσα και τις χωρητικότητες που αυτά φέρουν, είναι συχνό φαινόμενο τα αρχεία να παραμένουν στον τόμο πολύ καιρό αφού οι εγγραφές τους έχουν επαναχρησιμοποιηθεί.

Η διαδικασία ανάκτησης λοιπόν χωρίζεται σε δύο μεγάλες κατηγορίες, στην ανάκτηση των αρχείων για τα οποία η εγγραφή αρχείου στο $MFT ακόμα διασώζεται και για εκείνα η εγγραφή των οποίων έχει επαναχρησιμοποιηθεί από το $MFT.

Στην πρώτη περίπτωση η διαδικασία ανάκτησης διακλαδίζεται ακόμα περισσότερο αναλόγως του τοπικού ή μη περιεχομένου. Σε ένα αρχείο τοπικού περιεχομένου η ανάκτηση του είναι τόσο απλή όσο η εύρεση της εγγραφής του και η πλοήγηση στο χαρακτηριστικό του $Data. Για ένα μη τοπικό χαρακτηριστικό η λίστα Data runs πρέπει να αναλυθεί και το αρχείο να επανακτηθεί με την ένωση των διαφόρων fragment από τα οποία αποτελείται (εφόσον αυτά βρίσκονται όλα άθικτα στον μη κατανεμημένο χώρο του τόμου).

Στη δεύτερη περίπτωση χρησιμοποιούνται διαφορετικές μέθοδοι το σύνολο των οποίων ονομάζεται χάραξη αρχείων (Data Carving) και αναλύονται στο κεφάλαιο 2.7. Και οι δύο περιπτώσεις θα παρουσιαστούν στο πρακτικό μέρος της εργασίας στη συνέχεια.

Εκτός του βασικού αυτού κορμού βημάτων το Σύστημα Αρχείων NTFS, με την πολυπλοκότητα του, φέρει πολλές ακόμη δομές πληροφορίας που είναι κομβικής σημασίας σε μία ψηφιακή εγκληματολογική έρευνα. Παρακάτω παρουσιάζονται οι βασικότερες εξ' αυτών.

2.4 Χώρος Υπολείμματος Αρχείου (Slack Space)

Χώρος υπολείμματος αρχείου αποκαλείται η περιοχή του τελευταίου cluster, το οποίο έχει κατανεμηθεί σε ένα αρχείο, η οποία δεν χρειάζεται για να αποθηκευτεί το λογικό μέγεθος του

αρχείου. Είναι δηλαδή χώρος που παραμένει ανεκμετάλλευτος από το αρχείο καθώς υπερβαίνει το λογικό του μέγεθος και δημιουργείται όταν το μέγεθος του αρχείο δεν είναι ακριβώς διαιρέσιμο με το μέγεθος των cluster του τόμου.

Το σημαντικό στοιχείο του χώρου υπολείμματος είναι ότι το Λειτουργικό Σύστημα τον αγνοεί καθώς κατά την χρήση του αρχείου ενδιαφέρεται μόνο για το ενεργό μέρος των δεδομένων του, ήτοι το λογικό μέγεθος του.

Ο χώρος υπολείμματος αρχείου χωρίζεται σε χώρο υπολείμματος RAM (RAM Slack) και κατάλοιπο χώρο (Residual Slack).

Χώρος Υπολείμματος RAM (RAM Slack)

Είναι η περιοχή από το τέλος του λογικού αρχείου μέχρι το τέλος του τελευταίου τομέα που αυτό καταλαμβάνει. Ο τομέας είναι η μικρότερη εγγράψιμη μονάδα στο Σύστημα Αρχείων NTFS. Ακόμα και αν ο τελευταίος τομέας που καταλαμβάνει το αρχείο δεν απαιτείται ολόκληρος για την εγγραφή του, το Σύστημα πρέπει ακόμα να γράψει κάτι στα εναπομείναντα bytes. Κατά τα παλαιότερα χρόνια τα δεδομένα που γράφονταν σε αυτή την περιοχή αποτελούσαν δεδομένα που το Σύστημα είχε πάρει από την μνήμη RAM με τυχαίο τρόπο. Εξαιτίας αυτής της υλοποίησης ο χώρος υπολείμματος Ram μπορούσε να περιέχει οποιαδήποτε πληροφορία βρισκόταν στους ενταμιευτές της κύριας μνήμης εκείνη τη στιγμή όπως ιστοσελίδες, αρχεία κειμένου, κωδικούς πρόσβασης κλπ. Εντοπίζοντας έλλειμμα ασφάλειας σε αυτή την τεχνική η Microsoft με την έκδοση του Service Pack 1 για τα Windows 95 άλλαξε την υλοποίηση της RAM slack γεμίζοντας τον χώρο αυτό με την δεκαεξαδική τιμή 0x00 (κενή τιμή). Ως συνέπεια αυτής της αλλαγής ο χώρος υπολείμματος RAM δεν παρουσιάζει κανένα εργαστηριακό ενδιαφέρον.

Κατάλοιπος Χώρος (Residual Slack)

Είναι η περιοχή από το τέλος του χώρου υπολείμματος RAM μέχρι το τέλος του τελευταίου cluster που καταλαμβάνει το αρχείο. Ο κατάλοιπος χώρος ενδέχεται να περιέχει δεδομένα από αρχεία που καταλάμβαναν τα ίδια cluster σε στιγμή προγενέστερη της εγγραφής του τρέχοντος αρχείου στον τόμο. Συνεπώς αυτός ο χώρος μπορεί να παράσχει στον εξεταστή πειστήρια σε διάφορες μορφές όπως έγγραφα, φωτογραφίες, ιστορικό περιήγησης διαδικτύου και συνομιλιών κλπ.

Επειδή τα δεδομένα που περιέχονται στον κατάλοιπο χώρο είναι ατελή ο εξεταστής θα πρέπει να είναι πολύ προσεκτικός με τη διαμόρφωση πεποίθησης βασισμένος μόνο σε αυτά καθώς προέρχονται από μερικώς διαγεγραμμένα αρχεία. Σε σύγχρονα αποθηκευτικά μέσα όπου το μέγεθος του κάθε cluster μπορεί να φτάνει ακόμα και τους 64 τομείς, ο κατάλοιπος χώρος μπορεί να περιέχει πολύ σημαντικά μεγέθη πληροφορίας.

Με το ακόλουθο παράδειγμα που απεικονίζεται στις παρακάτω τρείς εικόνες θα παρουσιαστούν οι έννοιες του χώρου υπολείμματος αρχείου. Υποθέτουμε ότι ο τόμος φέρει clusters μεγέθους τεσσάρων τομέων.

1ος Τομέας	2ος Τομέας	3ος Τομέας	4ος Τομέας
512 bytes	512 bytes	512 bytes	512 bytes
Cluster μεγέθους τεσσάρων τομέων			

Το κόκκινο αρχείο μεγέθους 1900 bytes δημιουργείται στον τόμο. Θα χρειαστεί ένα cluster για να αποθηκευτεί. Το ένα cluster όμως έχει συνολικό μέγεθος 4x512= 2048 bytes. Κατά συνέπεια το κόκκινο αρχείο θα καταλάβει ολόκληρους τους πρώτους τρείς τομείς και 364 bytes από τον τελευταίο τομέα δημιουργώντας χώρο RAM Slack μεγέθους 148 byte που γεμίζεται από το Σύστημα με μηδενικές τιμές.

1ος Τομέας	2ος Τομέας	3ος Τομέας	4ος Τομέας

512	512	512	364	148
Κόκκινο αρχείο και RAM Slack				

Στη συνέχεια το κόκκινο αρχείο διαγράφεται. Ακολουθούνται τα βήματα που περιγράφηκαν παραπάνω και κατόπιν ένα άλλο αρχείο, το γκρι αρχείο μεγέθους 600 bytes καταλαμβάνει το ίδιο cluster και αποθηκεύεται σε αυτό, διαγράφοντας μερικώς το κόκκινο αρχείο. Το γκρι αρχείο λόγω μεγέθους θα καταλάβει όλο τον πρώτο τομέα και 88 bytes από τον δεύτερο. Το Σύστημα πρέπει οπωσδήποτε να γράψει ολόκληρο τον τομέα αυτόν οπότε γεμίζει με μηδενικές τιμές τα υπολειπόμενα 424 bytes του δεύτερου τομέα. Οι υπόλοιποι δύο τομείς αποτελούν κατάλοιπο χώρο για το γκρι αρχείο και παραμένουν άθικτοι διατηρώντας ένα μέρος του κόκκινου αρχείου.

$1^{ος}$ Τομέας	$2^{ος}$ Τομέας		$3^{ος}$ Τομέας	$4^{ος}$ Τομέας	
512	88	424	512	364	148
Γκρι αρχείο και κατάλοιπος χώρος Residual Slack					

Η παρουσία μέρους του κόκκινου διαγεγραμμένου αρχείου σε cluster το οποίο είναι ήδη κατανεμημένο σε άλλο αρχείο αυξάνει την πολυπλοκότητα της ανάλυσης του. Η εγγραφή του αρχείου έχει επαναχρησιμοποιηθεί και την ίδια στιγμή το αρχείο δε βρίσκεται αυτούσιο στον μη κατανεμημένο χώρο όπου ειδικές διαδικασίες θα μπορούσαν να πραγματοποιηθούν προκειμένου να ανακτηθεί (χάραξη δεδομένων).

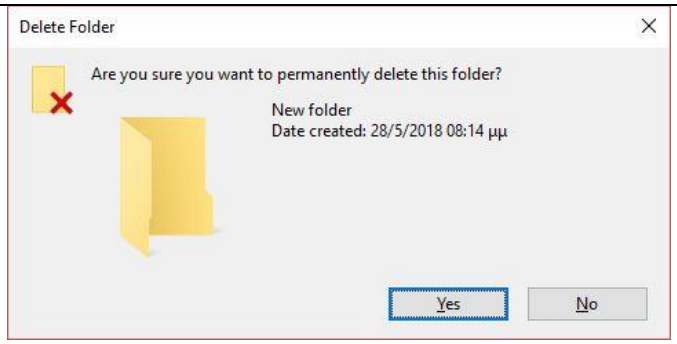
Ένεκα των προαναφερθέντων η διαδικασία ανάκτησης δεδομένων από τον κατάλοιπο χώρο δεν δύναται να αυτοματοποιηθεί από τα σύγχρονα εγκληματολογικά λογισμικά και απαιτεί επίπονη χειροκίνητη εξέταση από τον ίδιο τον ερευνητή. Τα αποτελέσματα που αποφέρει ποικίλουν και η αποδεικτική τους αξία στο δικαστήριο είναι περιορισμένη αφού τα ανακτηθέντα δεδομένα αποτελούν μόνο μέρος ενός αρχείου και δε συνοδεύονται από λοιπά μεταδεδομένα τα οποία θα μπορούσαν να προσφέρουν σφαιρική άποψη γύρω από το αρχείο.

2.5 Κάδος Ανακύκλωσης και Ανάκτηση Δεδομένων

Ο κάδος ανακύκλωσης σχεδιάστηκε από την Microsoft για να παρέχει μία επιπλέον δικλείδα ασφαλείας από ακούσιες διαγραφές δεδομένων. Οι χρήστες προστατεύονται από τον ίδιο τον εαυτό τους καθώς αντί για ολική διαγραφή των αρχείων λαμβάνει χώρα η μετακίνησή τους στον κάδο ανακύκλωσης από όπου είναι και πάλι πλήρως ανακτήσιμα με μια απλή επιλογή επαναφοράς (restore).

Ο κάδος ανακύκλωσης είναι μία ομάδα κρυφών καταλόγων συστήματος που περιέχουν τα ανεπιθύμητα αρχεία. Όταν ένας χρήστης “διαγράψει” ένα αρχείο σε μία συσκευή με λειτουργικό Windows που φέρει το Σύστημα Αρχείων NTFS, το ίδιο το περιεχόμενο του αρχείου δεν διαγράφεται. Αντιθέτως η εγγραφή αρχείου στο $MFT αλλάζει για να δείξει ότι το αρχείο πλέον βρίσκεται στον κάδο ανακύκλωσης.

Σε όλες τις εκδόσεις των Λειτουργικών Συστημάτων Windows εάν ο χρήστης πατήσει ταυτόχρονα τα πλήκτρα shift και delete τότε ο κάδος ανακύκλωσης παρακάμπτεται και τα αρχεία διαγράφονται “μόνιμα” (permanently) με τις διαδικασίες που περιγράφηκαν στο μέρος 2.3 του παρόντος κεφαλαίου.



Παράκαμψη κάδου ανακύκλωσης με τη χρήση των πλήκτρων Shift + Delete

Η διαδικασία ανακύκλωσης εξαρτάται από την έκδοση του Λειτουργικού Συστήματος και τον τύπο του Συστήματος Αρχείων που χρησιμοποιείται από το μέσο. Η αρχιτεκτονική του κάδου ανακύκλωσης στα Windows XP διαφέρει από τα λοιπές σύγχρονες εκδόσεις και δεν θα αναλυθεί στην παρούσα εργασία ως μία έκδοση παρωχημένη. Αντιθέτως τα Windows Vista,7,8 και 10 χρησιμοποιούν τις ίδιες διαδικασίες οι οποίες παρουσιάζονται παρακάτω.

Κάδος Ανακύκλωσης ($RECYCLE.BIN) Windows Vista,7,8,10

Η ονομασία που φέρει ο κάδος ανακύκλωσης στις εν λόγω εκδόσεις Λειτουργικών Συστημάτων είναι $RECYCLE.BIN. Ο κάδος ανακύκλωσης δημιουργείται σε κάθε λογικό τόμο που συνδέεται στο Σύστημα και φέρει την ένδειξη “Fixed Disk”, πληροφορία που παρέχεται από τον τομέα εκκίνησης. Γίνεται μνεία ότι την προαναφερθείσα ένδειξη φέρουν και αφαιρούμενα αποθηκευτικά μέσα όπως φορητοί σκληροί δίσκοι ή φορητά αποθηκευτικά μέσα διασύνδεσης USB.

Στο Σύστημα Αρχείων NTFS, την πρώτη φορά που ένας χρήστης συνδέεται στο μέσο (login) ένας υποκατάλογος δημιουργείται υπό του καταλόγου $RECYCLE.BIN, ο οποίος φέρει ως όνομα το αναγνωριστικό ασφαλείας του χρήστη (SID) και σχετικό μονοπάτι στον τόμο «%γράμμα τόμου%\Recycle.bin\%SID%». Οτιδήποτε διαγραφεί εφεξής από τον χρήστη μετακινείται σε αυτόν τον φάκελο. Το SID του χρήστη που χρησιμοποιεί το Σύστημα Αρχείων για την ονομασία του φακέλου είναι αποθηκευμένο στο μητρώο Registry του υπολογιστή και μπορεί να αντιστοιχηθεί με το όνομα του από το κλειδί «HKLM\SOFTWARE\ Microsoft\WindowsNT\CurrentVersion\ProfileList».

Ενώ, λοιπόν, η ανάκτηση ενός αρχείου από τον κάδο ανακύκλωσης είναι μία απλή διαδικασία σε ένα σύστημα που βρίσκεται σε λειτουργία, δεν ισχύει το ίδιο κατά τη διενέργεια μίας εγκληματολογικής εξέτασης.

Κατά την εργαστηριακή εξέταση, όπως προαναφέρθηκε στο πρώτο κεφάλαιο, πρέπει να τηρούνται ορισμένες συνθήκες ώστε το μέσο να προφυλάσσεται από αλλοιώσεις και τα αποτελέσματα να είναι αποδεκτά εντός των δικαστικών αιθουσών. Η επαναφορά ενός αρχείου φυσικά και δε θα μπορούσε να πραγματοποιηθεί καθόσον πρώτον αυτό προϋποθέτει το αρχικό αποθηκευτικό μέσο/πειστήριο να τεθεί σε λειτουργία, ενέργεια απαγορευμένη κατά την πρακτική της Ψηφιακής Εγκληματολογίας, και δεύτερον αυτό θα αποτελούσε μία αλλοίωση της αρχικής εικόνας που είχε το πειστήριο κατά τη στιγμή της κατάσχεσής του από τον φερόμενο ως δράστη.

Το Σύστημα Αρχείων NTFS παρέχει, παρόλαυτα, δομές δεδομένων που σχετίζονται με τα ανακυκλούμενα αρχεία και τις οποίες ο εξεταστής μπορεί να εκμεταλλευτεί ώστε να αποκτήσει πλήρη πληροφόρηση για τα αρχεία, τηρώντας παράλληλα όλες τις βασικές αρχές τις Ψηφιακής Εγκληματολογίας.

Δομές Δεδομένων Κάδου Ανακύκλωσης

Όταν ένα αρχείο διαγράφεται, η εγγραφή αρχείου του αλλάζει προκειμένου να δείχνει ότι ο νέος αριθμός εγγραφής $MFT για το γονικό κατάλογο του αρχείου είναι πλέον αυτός του $RECYCLE.BIN. Στη συνέχεια το ο κάδος ανακύκλωσης δημιουργεί ένα ζευγάρι αρχείων που χρησιμοποιεί για να εντοπίζει ένα διαγεγραμμένο στοιχείο.

Το πρώτο αρχείο από το ζεύγος φέρει ονομασία της μορφής “$Rxxxxxx”, ήτοι $R ακολουθούμενου από ένα όνομα 6 τυχαίων αλφαριθμητικών χαρακτήρων ενώ παράλληλα διατηρεί την επέκταση του αρχικού αρχείου. Αυτό το αρχείο περιέχει το περιεχόμενο του αρχείου που ανακυκλώθηκε από τον χρήστη και ταυτίζεται με την αρχική εγγραφή αρχείου στον πίνακα $MFT. Για παράδειγμα το αρχείο picture.jpg θα μετατραπεί σε $R123ASD.jpg.

Το δεύτερο αρχείο που δημιουργείται αρχίζει με τους χαρακτήρες $I και ακολουθείται από την ίδια τυχαία αλφαριθμητική ονομασία και την ίδια επέκταση όπως και το αρχείο $R. Αυτό το αρχείο περιέχει την χρονοσφραγίδα διαγραφής καθώς και το αρχικό πλήρες μονοπάτι όπου το αρχείο βρισκόταν αποθηκευμένο πριν την ανακύκλωσή του. Στη συνέχεια του ανωτέρω παραδείγματος λοιπόν θα είχαμε τη δημιουργία του αρχείου $I123ASD.jpg. Αυτό το δεύτερο αρχείο διαφέρει μεταξύ των διαφόρων εκδόσεων των Λειτουργικών Windows.

Για τα Windows Vista, 7 και 8 το μέγεθός του είναι προκαθορισμένο στα 544 bytes αλλά στα Windows 10 χρησιμοποιεί επιπλέον πεδία ώστε να καθορίζει ακριβώς το μέγεθος του μονοπατιού και συνεπώς να χρησιμοποιεί μόνο τα απαραίτητα byte για την αποθήκευση της πληροφορίας του, συμβάλλοντας έτσι στην αποδοτικότητα του Συστήματος. Τέλος οι επικεφαλίδες του αρχείου διαφέρουν επίσης μεταξύ των συστημάτων, με τα Windows Vista, 7 και 8 να χρησιμοποιούν τη δεκαεξαδική τιμή 0x01 00 00 00 00 00 00 00 ενώ τα Windows 10 την τιμή 0x02 00 00 00 00 00 00 00.

Η δομή δεδομένων για το αρχείο $I στα Λειτουργικά Συστήματα Windows Vista, 7 και 8 δίνεται στον παρακάτω πίνακα:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Περιγραφή
00	8	Επικεφαλίδα αρχείου
08	8	Αρχικό μέγεθος αρχείου σε bytes
10	8	Χρονοσφραγίδα ανακύκλωσης αρχείου (FILETIME σε UTC)
20	ποικίλει	Αρχικό μονοπάτι αρχείου και ονομασία (χαρακτήρες Unicode)
Δομή Δεδομένων αρχείου $I στα Windows Vista, 7 , 8		

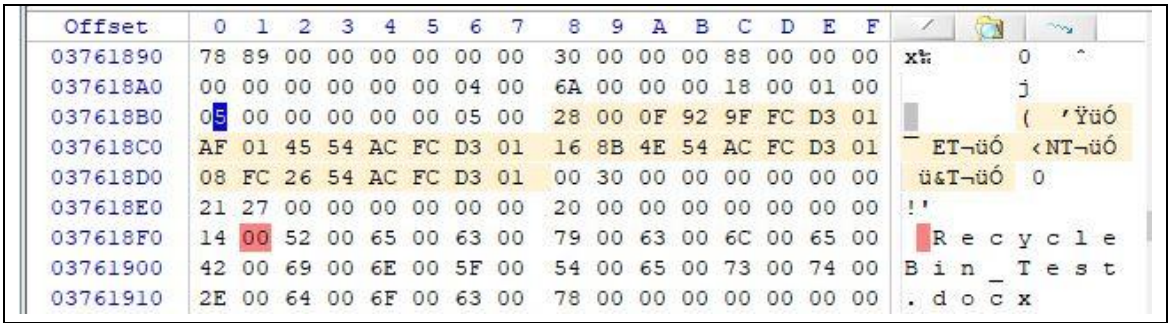
Η δομή δεδομένων για το αρχείο $I στο Λειτουργικό Σύστημα Windows 10 δίνεται στον παρακάτω πίνακα:

Θέση (δεκαεξαδικά 0x)	Μήκος (Bytes)	Περιγραφή
00	8	Επικεφαλίδα αρχείου
08	8	Αρχικό μέγεθος αρχείου σε bytes
10	8	Χρονοσφραγίδα ανακύκλωσης αρχείου (FILETIME σε UTC)
18	4	Μέγεθος αρχικού μονοπατιού (χαρακτήρες ASCII)
1C	ποικίλει	Αρχικό μονοπάτι αρχείου και ονομασία (χαρακτήρες Unicode)
Δομή Δεδομένων αρχείου $I στα Windows 10		

Σημειώνεται ότι οι χαρακτήρες ASCII απαιτούν τη χρήση ενός byte πληροφορίας για να απεικονιστούν, ενώ οι χαρακτήρες Unicode τη χρήση δύο bytes.

Στο παρακάτω παράδειγμα εμφαίνεται η χρήση των δομών ανάκτησης που προσφέρει ο κάδος ανακύκλωσης κατά τη διάρκεια μίας εγκληματολογικής εξέτασης. Στο μέσο που έχει χρησιμοποιηθεί ανατέθηκε το γράμμα “E:” από το Λειτουργικό Σύστημα κατά τη διαδικασία εκκίνησής του (mount). Το αρχείο κειμένου Word με την ονομασία “RecycleBin_Test.docx” δημιουργήθηκε στον ριζικό κατάλογο του τόμου.

Εξετάζεται η εγγραφή αρχείου στου πίνακα $MFT για το αρχείο κειμένου. Όπως φαίνεται στη θέση 0x037618B1 ως 0x037618B5 τα πρώτα έξι bytes του περιεχομένου του χαρακτηριστικού δείχνουν τον αριθμό εγγραφής $MFT για τον γονικό κατάλογο του αρχείου. Η δεκαεξαδική τιμή 0x05 00 00 00 00 00 ισούται με τη δεκαδική τιμή 5 και συνεπώς υποδεικνύει τον κατάλογο root του μέσου (6η κατά σειρά καταχώριση στο $MFT με αριθμό εγγραφής 5)

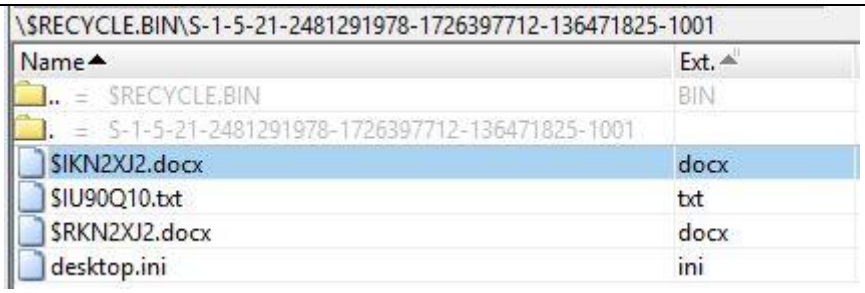


Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03761890	78	89	00	00	00	00	00	00	30	00	00	00	88	00	00	00
037618A0	00	00	00	00	00	00	04	00	6A	00	00	00	18	00	01	00
037618B0	05	00	00	00	00	00	05	00	28	00	0F	92	9F	FC	D3	01
037618C0	AF	01	45	54	AC	FC	D3	01	16	8B	4E	54	AC	FC	D3	01
037618D0	08	FC	26	54	AC	FC	D3	01	00	30	00	00	00	00	00	00
037618E0	21	27	00	00	00	00	00	00	20	00	00	00	00	00	00	00
037618F0	14	00	52	00	65	00	63	00	79	00	63	00	6C	00	65	00
03761900	42	00	69	00	6E	00	5F	00	54	00	65	00	73	00	74	00
03761910	2E	00	64	00	6F	00	63	00	78	00	00	00	00	00	00	00

Χαρακτηριστικό $File_Name για το αρχείο “RecycleBin_Test.docx”

Επιπλέον παρουσιάζονται έτερα στοιχεία πληροφορίας όπως χρονοσφραγίδες, μεγέθη που αφορούν το αρχείο καθώς και η ονομασία του “ RecycleBin_Test.docx ” η οποία ξεκινάει στη θέση 0x037618F2 .

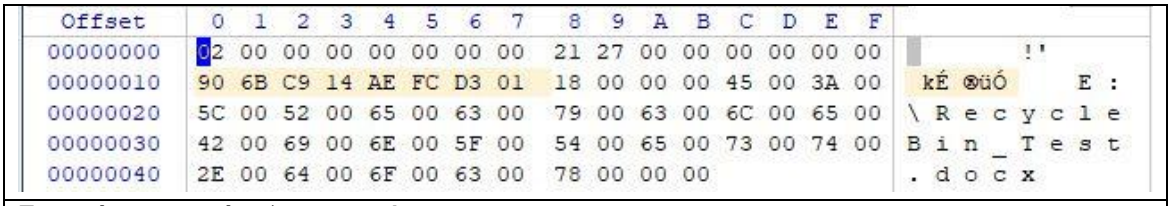
Στη συνέχεια το αρχείο “διαγράφεται” με την απλή χρήση του πλήκτρου delete. Ο κατάλογος Recycle.bin εμπεριέχει ήδη ένα κατάλογο με το αναγνωριστικό ασφαλείας του χρήστη (S-1-5-21-2481291978-1726397712-136471825-1001), ο οποίος και πλέον μετά τη διαγραφή περιέχει μεταξύ άλλων και επιπλέον δύο αρχεία, τα $RKN2XJ2.docx και $IKN2XJ2.docx



Περιεχόμενα του κάδου ανακύκλωσης μέσω του λογισμικού Winhex

Επόμενο βήμα είναι να συνδέσουμε τον χρήστη με την ενέργεια της διαγραφής του αρχείου, ενέργεια ιδιαίτερα σημαντική σε μία εγκληματολογική εξέταση καθώς όχι μόνο αποδεικνύει γνώση του αρχείου αλλά φανερώνει και βούληση του χρήστη να το εξαφανίσει. Επειδή σε πολλές έρευνες αντιμετωπίζεται το ενδεχόμενο να υπάρχουν περισσότεροι του ενός καταχωρημένοι χρήστες σε μία συσκευή είναι απαραίτητο να ερμηνευτεί ορθώς ποιος χρήστης αντιστοιχεί στο SID “ S-1-5-21-2481291978-1726397712-136471825-1001”. Για αυτό το σκοπό χρησιμοποιούνται ειδικά λογισμικά καθώς κατά τη διενέργεια μίας εγκληματολογικής εξέτασης επί ενός ανενεργού μέσου (dead analysis) ο εξεταστής δε μπορεί να έχει πρόσβαση στο μητρώο Registry.

Στη συνέχεια αναλύοντας το νεοδημιουργηθέν αρχείο $IKN2XJ2.docx παρατηρούμε το περιεχόμενό του

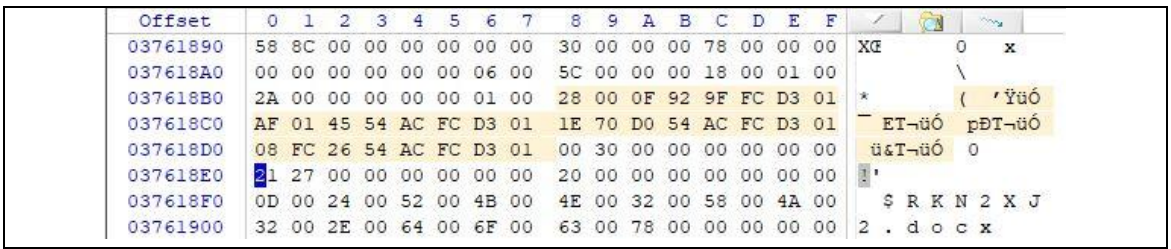


Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	02	00	00	00	00	00	00	00	21	27	00	00	00	00	00	00	!'
00000010	90	6B	C9	14	AE	FC	D3	01	18	00	00	00	45	00	3A	00	kÉ ®üÓ E :
00000020	5C	00	52	00	65	00	63	00	79	00	63	00	6C	00	65	00	\ R e c y c l e
00000030	42	00	69	00	6E	00	5F	00	54	00	65	00	73	00	74	00	B i n _ T e s t
00000040	2E	00	64	00	6F	00	63	00	78	00	00	00					. d o c x

Περιεχόμενο αρχείου $IKN2XJ2.docx

- Επικεφαλίδα χαρακτηριστικού – 0x02 00 00 00 00 00 00 00 = Λειτουργικό Σύστημα σε χρήση Windows 10
- Αρχικό μέγεθος αρχείου σε bytes – 0x27 21 (Little Endian) = 10.017 bytes
- Χρονοσφραγίδα ανακύκλωσης αρχείου – 0x01 D3 FC AE 14 C9 6B 90 (Little Endian)= 05-06-2018, 09:17:52 utc
- Μέγεθος αρχικού μονοπατιού – 0x18 = 24 χαρακτήρες ASCII
- Αρχικό μονοπάτι αρχείου και ονομασία = E:\RecycleBin_Test.docx

Στη θέση της αρχικής εγγραφής του αρχείου δείχνει πλέον το αρχείο $RKN2XJ2.docx. Οι λίστες data runs περιγράφουν τη θέση του περιεχομένου του αρχείου στον τόμο ενώ πλέον το χαρακτηριστικό $File_Name έχει αλλάξει προκειμένου να δείχνει τον τωρινό γονικό κατάλογο, αριθμό διαδοχής και ονομασία.

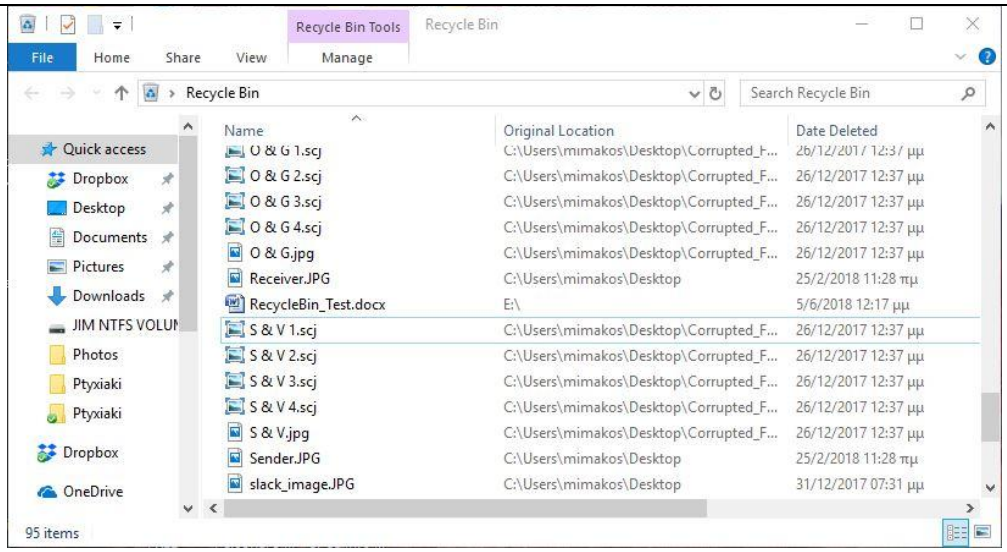


Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
03761890	58	8C	00	00	00	00	00	00	30	00	00	00	78	00	00	00	XŒ 0 x
037618A0	00	00	00	00	00	00	06	00	5C	00	00	00	18	00	01	00	\
037618B0	2A	00	00	00	00	00	01	00	28	00	0F	92	9F	FC	D3	01	* ('ŸüÓ
037618C0	AF	01	45	54	AC	FC	D3	01	1E	70	D0	54	AC	FC	D3	01	¯ ET¬üÓ pÐT¬üÓ
037618D0	08	FC	26	54	AC	FC	D3	01	00	30	00	00	00	00	00	00	ü&T¬üÓ 0
037618E0	21	27	00	00	00	00	00	00	20	00	00	00	00	00	00	00	!'
037618F0	0D	00	24	00	52	00	4B	00	4E	00	32	00	58	00	4A	00	$ R K N 2 X J
03761900	32	00	2E	00	64	00	6F	00	63	00	78	00	00	00	00	00	2 . d o c x

Περιεχόμενο αρχείου $RKN2XJ2.docx

Παρατηρούμε στη θέση 0x037618B1 ως 0x037618B5 τα πρώτα έξι bytes του περιεχομένου του χαρακτηριστικού δείχνουν τον αριθμό εγγραφής $MFT για τον γονικό κατάλογο του αρχείου που πλέον φέρει τη δεκαεξαδική τιμή 0x2A 00 00 00 00 00. Αυτή η τιμή ισούται με τη δεκαδική τιμή 42 και συνεπώς υποδεικνύει την εγγραφή $MFT για τον κατάλογο S-1-5-21-2481291978-1726397712-136471825-1001 εντός του οποίου πλέον βρίσκεται το αρχείο.

Σημαντικό στοιχείο που πρέπει να επισημανθεί είναι ότι τα αρχεία $R, $I αποτελούν εσωτερικές δομές δεδομένων του Συστήματος Αρχείων NTFS. Σε ένα σύστημα που βρίσκεται σε λειτουργία ο χρήστης ποτέ δε θα δει αυτά τα αρχεία καθώς το Λειτουργικό Σύστημα κάνει ανάλυση αυτών των δομών και παρουσιάζει τα πραγματικά αρχεία στο γραφικό περιβάλλον. Συνεπώς ο χρήστης που θα περιηγηθεί στον κάδο ανακύκλωσης αυτού του συστήματος θα δει τα διαγραφέντα αρχεία με τη μορφή που αυτά καταλάμβαναν στον τόμο πριν τη διαγραφή τους.



Περιεχόμενο κάδου ανακύκλωσης στο γραφικό περιβάλλον των Windows

Το αρχείο “RecycleBin_Test.docx” που ανευρίσκεται εντός του κάδου ανακύκλωσης παρουσιάζεται αυτούσιο αλλά διαφέρει στη χρονοσφραγίδα διαγραφής (Date Deleted) κατά τρείς ώρες σε σχέση με την πληροφορία που παρείχε το αρχείο “$IKN2XJ2.docx”. Αυτό συμβαίνει γιατί η πληροφορία στο Σύστημα Αρχείων NTFS αποθηκεύεται σε μορφή FILETIME και είναι πάντα σε ώρα ζώνης UTC ενώ το γραφικό περιβάλλον των Windows παρέχει την τοπική ώρα που είναι ρυθμισμένη η συσκευή σύμφωνα με τις πληροφορίες του μητρώου Registry.

Επαναφορά - Διαγραφή Αρχείων από τον Κάδο Ανακύκλωσης

Όταν ο χρήστης αποφασίσει να επαναφέρει το αρχείο από τον κάδο ανακύκλωσης αυτό θα λάβει και πάλι την αντίστοιχη θέση που καταλάμβανε πριν τη διαγραφή του καθώς το Σύστημα μπορεί να ανοικοδομήσει όλη την ιεραρχία των καταλόγων με τη βοήθεια του αρχείου $I. Το αρχείο θα εμφανιστεί και πάλι στο γραφικό περιβάλλον και ο χρήστης θα μπορεί να το χειριστεί όπως και πριν τη διαδικασία ανακύκλωσής του.

Το ίδιο το αρχείο $I παραμένει ενεργό στον αντίστοιχο φάκελο χρήστη του Recycle.bin. Το αρχείο $R παύει να υπάρχει και η εγγραφή αρχείου του επανέρχεται στην αρχική της μορφή, με την αρχική ονομασία και αριθμό εγγραφής $MFT γονικού καταλόγου του προηγουμένως ανακυκλωθέντος αρχείου. Οι χρονοσφραγίδες των χαρακτηριστικών $Standard_Information και $File_Name παραμένουν ίδιες για τα πεδία χρόνου δημιουργίας, χρόνου τροποποίησης και χρόνου τελευταίας προσπέλασης και η μόνη τιμή που αλλάζει είναι η ημεροχρονολογία και ώρα τροποποίησης της εγγραφής $MFT λόγω της μετακίνησης του αρχείου σε διαφορετικό κατάλογο και της αντίστοιχης ενημέρωσης που έλαβε χώρα εκείνη τη στιγμή.

Όταν ο χρήστης επιλέξει να διαγράψει ένα αρχείο από τον κάδο ανακύκλωσης οι εγγραφές αρχείων για το ζευγάρι των αρχείων $R, $I που το περιγράφουν διαγράφονται ολοκληρωτικά σύμφωνα με τις διαδικασίες που έχουν περιγραφεί στο μέρος 2.3 του παρόντος κεφαλαίου. Επειδή ένα ενεργό Σύστημα με Λειτουργικό Windows πραγματοποιεί διαρκώς εργασίες στο παρασκήνιο, οι εγγραφές αρχείων για τα δύο προαναφερθέντα αρχεία στο $MFT μπορούν να επαναγραφούν πολύ γρήγορα χωρίς απαραίτητα ο χρήστης να εκτελέσει κάποια ενέργεια. Αυτό καθιστά την επανάκτηση του αρχείου αρκετά δύσκολη και πραγματοποιήσιμη μόνο μέσω της διαδικασίας χάραξης από τον μη κατανεμημένο χώρο.

2.6 Διαδικασία Διαμόρφωσης (FORMAT)

Μία συνηθισμένη διαδικασία καταστροφής δεδομένων από το δράστη μίας έκνομης ενέργειας είναι η διαδικασία διαμόρφωσης δίσκου Format, κατά την οποία το μέσο αναδιαμορφώνει εξ' αρχής τις δομές του και παρουσιάζεται ως κενό περιεχομένου στο γραφικό περιβάλλον. Η πραγματική όμως κατάσταση του δίσκου διαφέρει από αυτή που παρουσιάζει το γραφικό περιβάλλον διεπαφής στον χρήστη.

Η συμπεριφορά του Συστήματος Αρχείων NTFS κατά τη διαδικασία διαμόρφωσης δίσκου και η τελική μορφή που θα αποκτήσει ο τόμος μετά τη διενέργεια αυτής, διαφέρει αναλόγως του Λειτουργικού Συστήματος και του είδους του Format που λαμβάνει χώρα. Υπάρχουν δύο είδη διαδικασιών διαμόρφωσης η γρήγορη και η πλήρης.

Διαδικασία Γρήγορης Διαμόρφωσης Δίσκου (Quick Format)

Σε όλες τις εκδόσεις των Λειτουργικών Συστημάτων Windows, μία γρήγορη διαδικασία διαμόρφωσης δίσκου θα επανεγράψει την κατανεμημένη περιοχή του νέου πίνακα $MFT με την δεκαεξαδική τιμή 0x00 (κενή τιμή). Αυτό έχει ως συνέπεια την απώλεια όλων των εγγραφών αρχείου που βρίσκονταν σε εκείνο το σημείο του πίνακα. Εάν ο νέος πίνακας $MFT είναι μικρότερου μεγέθους από τον προηγούμενο (το οποίο εξαρτάται από το μέγεθος των clusters) ή εάν ο πρώτος είχε κατακερματιστεί, τότε υπάρχουν εγγραφές που διασώζονται εκτός του μέρους που επαναγγράφηκε με την κενή τιμή και δύναται να ανακτηθούν. Η διαδικασία διαμόρφωσης θα επαναγράψει ακόμα τον τομέα εκκίνησης, θα επαναφέρει το αρχείο $Bitmap ώστε να παρουσιάζει όλα τα clusters του τόμου ως μη κατανεμημένα και δεν θα επηρεάσει κανένα άλλο αρχείο που βρίσκεται αποθηκευμένο στο μέσο. Το γραφικό περιβάλλον του Συστήματος θα παρουσιάζει τον τόμο ως κενό περιεχομένου αλλά στην πραγματικότητα όλα τα αρχεία που ενυπήρχαν στον τόμο πριν τη διαδικασία διαμόρφωσης βρίσκονται ακόμα σε αυτόν άθικτα.

Η αξιοσημείωτη διαφορά μεταξύ των λειτουργικών συστημάτων είναι το μέγεθος του πίνακα $MFT μετά τη διαμόρφωση. Ένας τόμος στα Windows XP έχει έναν πίνακα $MFT μεγέθους 32.768 bytes (8 clusters) μετά τη διαμόρφωσή του, ενώ στα Windows Vista και επόμενα οι τόμοι αποκτούν πίνακες μεγέθους 262,144 bytes (64 clusters).

Επίσης υπάρχει διαφορά στο πως γράφονται οι νέες μη χρησιμοποιούμενες εγγραφές αρχείου στο $MFT. Για τα Windows XP μόνο τα δύο πρώτα bytes γράφονται στην εγγραφή, τα οποία είναι η καταχώριση της τιμής Fix-up. Στα Windows Vista + οι μη χρησιμοποιούμενες εγγραφές φέρουν πλήρη επικεφαλίδα εγγραφής η οποία ακολουθείται από την τιμή λήξης της εγγραφής 0XFF FF FF FF.

Μετά από μια διαδικασία γρήγορης διαμόρφωσης δίσκου, εγγραφές αρχείων από την προηγούμενη διαμόρφωση του τόμου δύναται να εντοπιστούν στον μη κατανεμημένο χώρο και τα αρχεία τους να ανακτηθούν πλήρως, μετά από αναζήτηση της δεκαεξαδικής τιμής 0x46 49 4C 45 ή του όρου «FILE» που αποτελούν αναγνωριστικό κωδικό έναρξης εγγραφής αρχείου.

Διαδικασία Πλήρους Διαμόρφωσης Δίσκου (Full Format)

Σε μία διαδικασία πλήρους διαμόρφωσης τα Windows XP θα έχουν ακριβώς την ίδια αντίδραση όπως και στη γρήγορη διαδικασία και επίσης θα επιχειρήσουν να διαβάσουν από κάθε τομέα του μέσου ώστε να εντοπίσουν τυχόν ελαττωματικούς τομείς (bad sectors). Κανένα άλλο αρχείο ή κατάλογος δεν επαναγράφονται με την πλήρη διαδικασία διαμόρφωσης και τα δεδομένα τους παραμένουν αναλλοίωτα στον τόμο προς ανάκτηση.

Στα Windows Vista και επόμενα, μία διαδικασία πλήρους διαμόρφωσης δίσκου θα επαναγράψει όλους τους τομείς στο μέσο με την δεκαεξαδική τιμή 0x00 (κενή τιμή). Κατόπιν θα

κατασκευάσει εκ νέου τη δομή του Συστήματος Αρχείου με την δημιουργία όλων των απαραίτητων αρχείων μεταδεδομένων του NTFS.

Αυτή είναι και η μόνη περίπτωση κατά την οποία κάθε πληροφορία που τυχόν υπήρχε εντός του τόμου καταστρέφεται και πλέον κανένα αρχείο δεν είναι επανακτήσιμο.

2.7 Χάραξη Δεδομένων (Data Carving)

Μέχρι τώρα παρουσιάστηκαν τεχνικές ανάλυσης δεδομένων οι οποίες εκμεταλλεύονται δομές του Συστήματος Αρχείων. Τα δεδομένα που ανακτώνται έχουν ακόμα ενεργή συσχέτιση με το Σύστημα και η πληροφόρηση που αυτό το ίδιο προσφέρει για τα δεδομένα αποτελεί αναπόσπαστο μέρος της διαδικασίας ανάκτησης.

Υπάρχουν όμως περιπτώσεις όπου το αρχείο πλέον δεν φέρει κανένα ενεργό δεσμό με το Σύστημα Αρχείων. Η οριστική διαγραφή ενός αρχείου και η επαναχρησιμοποίηση της εγγραφής του στο $MFT οδηγούν κάθε πληροφορία σχετική για το αρχείο, πέραν του ίδιου του περιεχομένου του, να χαθεί από το Σύστημα. Σε αυτές τις περιπτώσεις το αρχείο ανήκει πλέον στον μη κατανεμημένο χώρο του τόμου και παραμένει αναλλοίωτο μέχρις ότου τα clusters τα οποία καταλαμβάνει να αποδοθούν σε άλλο αρχείο προς αποθήκευση. Η ανάκτηση ενός τέτοιου αρχείου είναι δυνατή μόνο μέσω των τεχνικών χάραξης δεδομένων (Data Carving).

«Χάραξη δεδομένων αποκαλείται η διαδικασία εξαγωγής μίας συλλογής δεδομένων από ένα μεγαλύτερο σετ δεδομένων. Οι τεχνικές χάραξης συχνά λαμβάνουν χώρα κατά τη διενέργεια ψηφιακής έρευνας όταν ο μη κατανεμημένος χώρος του Συστήματος Αρχείων αναλύεται προκειμένου να εξαχθούν αρχεία. Τα αρχεία “χαράσσονται” από τον μη κατανεμημένο χώρο με τη χρήση συγκεκριμένων τιμών επικεφαλίδας (header) και κατάληξης (footer). Οι δομές του Συστήματος Αρχείων δεν χρησιμοποιούνται κατά τη διαδικασία.» [http://dfrws.org].

Οι τιμές επικεφαλίδας και κατάληξης είναι δεκαεξαδικές αλφαριθμητικές τιμές που χαρακτηρίζουν ορισμένους τύπους αρχείων και συχνά αποκαλούνται υπογραφή του αρχείου (file signature). Η τιμή επικεφαλίδας καταλαμβάνει τα πρώτα bytes ενός αρχείου και μπορεί να φέρει μήκος από δύο έως και αρκετά bytes. Η τιμή κατάληξης αποτελεί τα τελευταία bytes του αρχείου και μπορεί να έχει αντίστοιχο μήκος. Δεν είναι απαραίτητο όλα τα αρχεία να φέρουν αυτές τις τιμές ενώ άλλα μπορεί να φέρουν μόνο τιμή επικεφαλίδας. Τέλος διαφορετικές εκδόσεις του ίδιου τύπου αρχείου μπορεί να έχουν την αρχική επικεφαλίδα της βασικής έκδοσης με επιπλέον τιμές προσαρτημένες στο τέλος της.

Η χάραξη δεδομένων χωρίζεται σε αρκετές κατηγορίες ανάλογα με την μεθοδολογία που ακολουθείται για να επιτευχθεί η ανάκτηση του αρχείου. Κάποιες από αυτές είναι:

- Βασική χάραξη = μέθοδος κατά την οποία η ανάκτηση δεδομένων επιτυγχάνεται με τη χρήση τιμών επικεφαλίδας (header) και κατάληξης (footer)
- Χάραξη επικεφαλίδας/μέγιστου μεγέθους = μέθοδος κατά την οποία η ανάκτηση δεδομένων επιτυγχάνεται με τη χρήση τιμών επικεφαλίδας και μίας τιμής μέγιστου μεγέθους που μπορεί να φέρει το αρχείο
- Στατιστική χάραξη = μέθοδος κατά την οποία η ανάκτηση δεδομένων επιτυγχάνεται με τη χρήση στατιστικής ανάλυσης ή ανάλυση χαρακτηριστικών του περιεχομένου του αρχείου
- Σημασιολογική χάραξη = μέθοδος κατά την οποία η ανάκτηση δεδομένων επιτυγχάνεται με τη χρήση γλωσσικής ανάλυσης του περιεχομένου του αρχείου

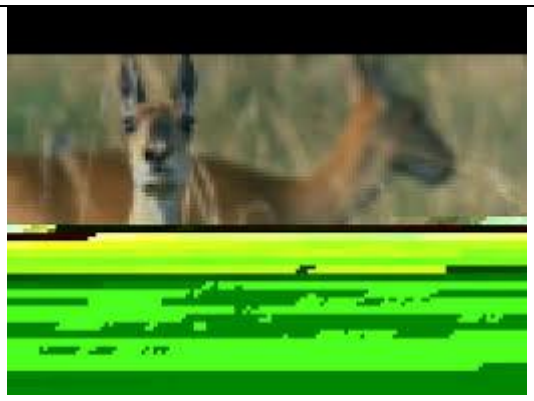
Βασική Χάραξη Δεδομένων

Η βασική χάραξη δεδομένων χρησιμοποιεί τις τιμές επικεφαλίδας και κατάληξης προκειμένου να οριοθετήσει τον χώρο του τόμου τον οποίο θα εξάγει ως αρχείο. Επειδή οι σύγχρονες υλοποιήσεις των Συστημάτων Αρχείων έχουν ως βασική επιδίωξη την αποτροπή κατακερματισμού των αρχείων, η βασική μέθοδος χάραξης επιτυγχάνει ικανοποιητικά αποτελέσματα στην ανάκτηση αρχείων από τον μη κατανεμημένο χώρο. Πολλά εγκληματολογικά λογισμικά προσφέρουν τη μέθοδο αυτή ως αναπόσπαστο μέρος του βασικού πακέτου τους καθώς είναι η πλέον αποδοτική.

Η βασική χάραξη δεδομένων φέρει όμως και ορισμένες αδυναμίες στην υλοποίησή της. Προϋποθέτει η αρχή του αρχείου να μην έχει επαναγραφεί από άλλο αρχείο ώστε η επικεφαλίδα να είναι αναγνωρίσιμη, το αρχείο να είναι συνεχές και όχι κατακερματισμένο ώστε η επικεφαλίδα και η κατάληξη να εμπερικλείουν το σύνολο του αρχείου που πρόκειται να εξαχθεί και τέλος το αρχείο να μην είναι συμπιεσμένο από το Σύστημα Αρχείων κατά την αποθήκευσή του. Σε περίπτωση που κάποιες εκ των ανωτέρω προϋποθέσεων δεν πληρούνται, η βασική μεθοδολογία χάραξης αποτυγχάνει να ανακτήσει τα δεδομένα ή παρέχει ημιτελή ή ψευδή αποτελέσματα (false positives) αναλόγως της ειδικής φύσης των ανακτηθέντων αρχείων. Η βασική χάραξη δεδομένων είναι επίσης ευάλωτη απέναντι στις τεχνικές παραποίησης επικεφαλίδας (file signature spoofing) καθώς η τροποποίηση των αρχικών byte ενός αρχείου έχουν ως απόρροια την αλλοίωση της αρχικής επικεφαλίδας του και καθιστούν την τεχνική χάραξης άχρηστη.

Σε κάθε περίπτωση, η πλειονότητα των αρχείων βρίσκονται αποθηκευμένα στο μέσο με τέτοιο τρόπο ώστε να τηρούνται όλες οι ανωτέρω προϋποθέσεις και συνεπώς η βασική χάραξη φέρει αναλογικά με τις λοιπές μεθόδους τα βέλτιστα αποτελέσματα τόσο ποσοτικά όσο και ποιοτικά.

Τέλος, αρχεία που έχουν ανακτηθεί μερικώς μπορεί (στις πλείστες περιπτώσεις) να μην είναι αξιοποιήσιμα στις δικαστικές αίθουσες αλλά δύναται να παρέχουν σημαντική πληροφόρηση και κατευθυντήριες γραμμές στον εξεταστή επιβεβαιώνοντας υποθέσεις και σενάρια κατά την εργαστηριακή εξέταση συνήθως σε σχέση με υποθέσεις που σχετίζονται με πορνογραφία ανηλίκων.



Μερικώς ανακτημένο αρχείο εικόνας

Η βασική χάραξη δεδομένων θα αναλυθεί με πρακτικό παράδειγμα στο επόμενο κεφάλαιο.

2.8 Τεχνικές Απόκρυψης Δεδομένων

Μέχρι αυτό το κεφάλαιο παρουσιάστηκαν τεχνικές ανάκτησης δεδομένων που είχαν διαγραφεί από τον χρήστη σε μία προσπάθεια εξαφάνισης οποιασδήποτε πληροφορίας θα

μπορούσε να συνδεθεί με ποινικώς κολάσιμες πράξεις. Αυτή τη μορφή φέρει και η πλειοψηφία των υποθέσεων για τις οποίες η Ψηφιακή Εγκληματολογία καλείται να δώσει λύση.

Ο δεύτερος πιο συνηθισμένος τρόπος του δράστη για να αποφύγει τις συνέπειες μίας έκνομης ενέργειας είναι η απόκρυψη των δεδομένων εκείνων που την αποδεικνύουν. Αν και σε αυτές τις περιπτώσεις τα αρχεία δεν έχουν διαγραφεί αλλά ενυπάρχουν ακόμα στον τόμο, η ανεύρεσή τους είναι εξίσου δύσκολη. Ένας αριθμός διαφορετικών τεχνικών μπορούν να εφαρμοστούν για να επιτευχθεί η απόκρυψη πληροφορίας, ειδικά σε Συστήματα Αρχείων τόσο πολύπλοκα όσο το NTFS το οποίο περιέχει πολλές διαφορετικές δομές για να επιτευχθεί αυτό.

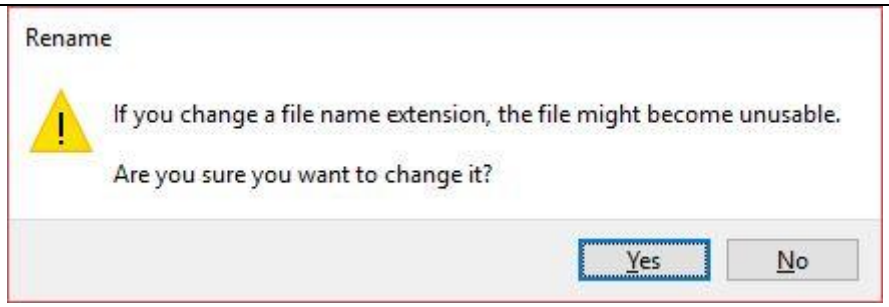
Στο παρόν μέρος του κεφαλαίου δεν θα αναλυθούν αυτοματοποιημένες τεχνικές που πραγματοποιούνται με τη χρήση εξειδικευμένου λογισμικού όπως η κρυπτογράφηση ή η στεγανογραφία, αντιθέτως θα παρουσιαστούν δομές που δύναται να χειραγωγηθούν χειροκίνητα ώστε να αποκρύψουν δεδομένα από το Λειτουργικό Σύστημα και τον εξεταστή.

Τροποποίηση Επέκτασης Αρχείου

Το Σύστημα Αρχείων NTFS αποθηκεύει τα αρχεία επί του τόμου προσαρτώντας τους μία σύντμηση γραμματικών χαρακτήρων μετά την ονομασία τους η οποία ονομάζεται επέκταση αρχείου. Το Λειτουργικό Σύστημα χρησιμοποιεί αυτές τις επεκτάσεις προκειμένου να αναγνωρίζει τον τύπο του αρχείου και να τον συνδέει με τις ανάλογες εφαρμογές που δύναται να το διαβάσουν, επεξεργαστούν και τροποποιήσουν.

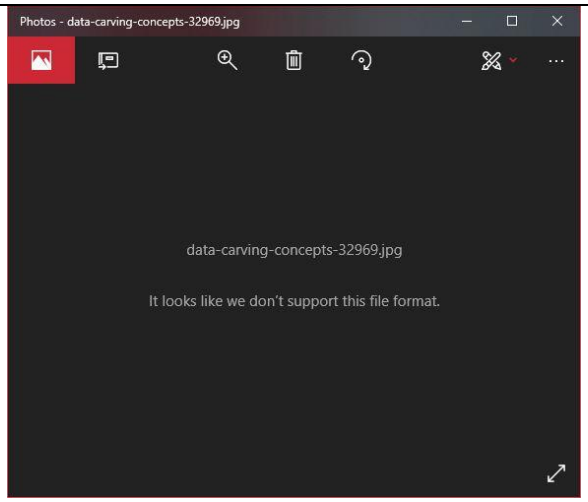
Η αλλαγή αυτής της επέκτασης αποτελεί την πλέον απλοϊκή τεχνική απόκρυψης πληροφορίας από έναν κακόβουλο χρήστη. Επιτυγχάνεται με την απλή μετονομασία του αρχείου και οδηγεί το Λειτουργικό Σύστημα σε παραγωγή μηνυμάτων λάθους όταν γίνει προσπάθεια να προσπελαστεί το αρχείο. Υπάρχουν εφαρμογές που θα ανοίξουν ένα αρχείο διαφορετικής επέκτασης (όπως για παράδειγμα ένας κειμενογράφος) αλλά θα παρουσιάσουν αλλοιωμένο το περιεχόμενό του.

Στο ακόλουθο παράδειγμα ένα αρχείο pdf με την ονομασία «data-carving-concepts-32969.pdf» τροποποιήθηκε σε αρχείο εικόνας jpeg. Το Σύστημα προειδοποιεί τον χρήστη για την αλλαγή και τις δυνητικές συνέπειές της.



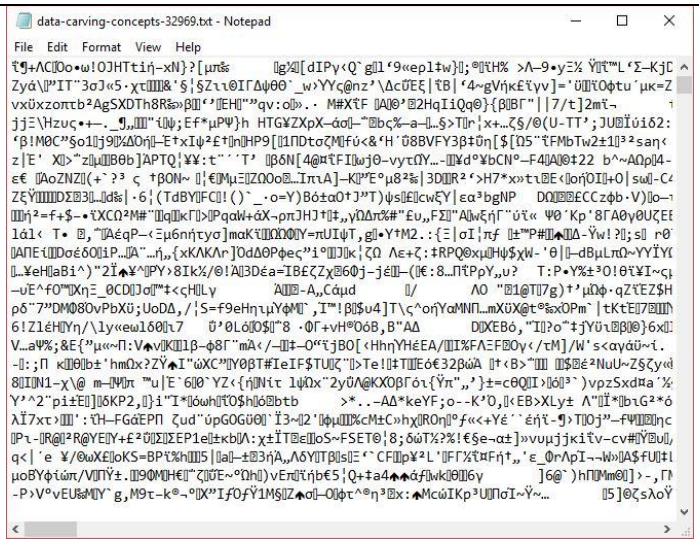
Προειδοποιητικό μήνυμα τροποποίησης επέκτασης

Μετά την τροποποίηση της επέκτασης το αρχείο προσπελάστηκε με διπλό κλικ. Αυτή η ενέργεια αναγκάζει το Λειτουργικό Σύστημα να ανατρέξει στο μητρώο Registry και να εντοπίσει την προεπιλεγμένη εφαρμογή με την οποία έχει συνδέσει τον τύπο αρχείου που ο χρήστης προσπαθεί να ανοίξει. Στην περίπτωση του αρχείου jpeg η εφαρμογή Photo Viewer των Windows επιχείρησε να ανοίξει το αρχείο δίνοντας φυσικά ένα μήνυμα λάθους. Το μήνυμα αυτό δεν είναι διαγνωστικό, συνεπώς ο χρήστης δεν γνωρίζει τον λόγο που το αρχείο δεν μπορεί να προσπελαστεί, αποτέλεσμα το οποίο θα μπορούσε να οφείλεται σε πολλαπλούς λόγους.



Μήνυμα μη υποστήριξης του αρχείου

Το ίδιο αρχείο τροποποιημένο σε αρχείο κειμένου επέκτασης .txt έχει προσπελαστεί με τον κειμενογράφο notepad των Windows χωρίς μήνυμα λάθους αλλά με εμφανώς αλλοιωμένη δομή δεδομένων η οποία οδηγεί εμμέσως στην απόκρυψη της πραγματικής πληροφορίας.



Αρχείο κειμένου pdf τροποποιημένο σε αρχείο κειμένου διαμόρφωσης .txt

Όσο απλοϊκός και αν είναι ο συγκεκριμένος τρόπος απόκρυψης δεδομένων θα μπορούσε δυνητικά να προκαλέσει μία πολύ σημαντική επιπλοκή στη διαδικασία της εγκληματολογικής εξέτασης. Αυτό γιατί υπάρχουν συγκεκριμένες υποθέσεις που σύμφωνα με την αρχική Παραγγελία του Εισαγγελέα, Ανακριτή ή και του Δικαστικού Συμβουλίου ο ερευνητής περιορίζεται να εξετάσει μόνο έναν ορισμένο τύπο αρχείων, όπως για παράδειγμα μόνο ηλεκτρονική αλληλογραφία ή αρχεία εικόνων. Σε μία τέτοια περίπτωση ένα μεγάλο μέρος αρχείων ενδιαφέροντος θα μπορούσε να παρακαμφθεί κατά την εξέταση και οι πληροφορίες του να χαθούν με σοβαρές συνέπειες για την υπόθεση.

Η αντιμετώπιση αυτής της τεχνικής απόκρυψης δεδομένων επιτυγχάνεται με τη διαδικασία της ανάλυσης υπογραφής αρχείου (File Signature Analysis) η οποία εξετάζει το εσωτερικό περιεχόμενο των αρχείων, στο επίπεδο των bytes, και αποφέρει σίγουρα αποτελέσματα για τον

πραγματικό τύπο αυτών. Τα εγκληματολογικά λογισμικά διαθέτουν αυτοματοποιημένες διαδικασίες κατά τις οποίες κάθε αρχείο στο μέσο ελέγχεται εσωτερικά στα πρώτα bytes των δεδομένων του για την ύπαρξη γνωστής επικεφαλίδας αρχείου (file header) και στη συνέχεια ο τύπος αρχείου που προκύπτει από αυτή συγκρίνεται με την επέκταση του αρχείου. Εάν τα δύο διαφέρουν το αρχείο επισημαίνεται ως αρχείο με κακή υπογραφή (Bad Signature) και ο πραγματικός τύπος του παρέχεται από το εγκληματολογικό εργαλείο βάση της επικεφαλίδας του. Τι συμβαίνει όμως σε περίπτωση που και η επικεφαλίδα του αρχείου έχει αλλοιωθεί;

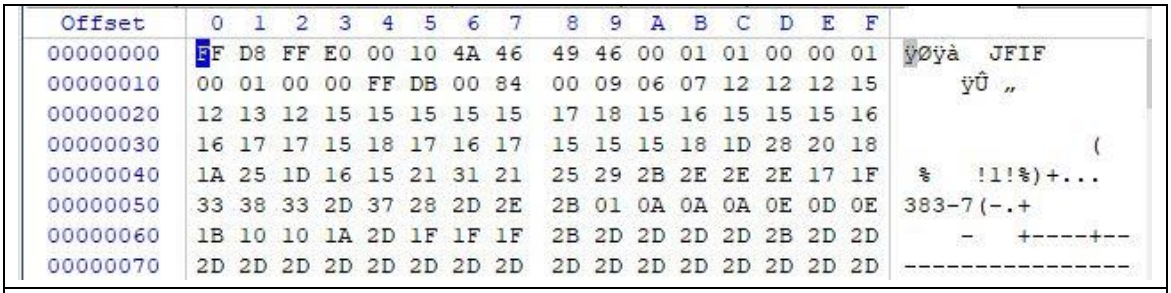
Τροποποίηση Επικεφαλίδας Αρχείου

Αυτή η τεχνική αποτελεί μία πιο εξειδικευμένη τεχνική απόκρυψης δεδομένων σε σχέση με την προηγούμενη. Ο κακόβουλος χρήστης χρειάζεται ένα πρόγραμμα επεξεργασίας δεκαεξαδικών τιμών (Hex Editor) και μία βαθύτερη γνώση της δομής δεδομένων των αρχείων. Κατέχοντας τον συνδυασμό αυτόν δεν είναι δύσκολο να αλλοιώσει την επικεφαλίδα ενός αρχείου και να το καταστήσει μη αναγνώσιμο.

Η πλειονότητα των διαφορετικών τύπων αρχείων φέρει χαρακτηριστικές ροές bytes στις πρώτες θέσεις του περιεχομένου τους με τις οποίες χαρακτηρίζεται ο τύπος τους. Αυτές οι ροές αποτελούν την υπογραφή του αρχείου και αναλύονται από τα προγράμματα που το χρησιμοποιούν ώστε να το αναλύσουν σωστά και να παρουσιάσουν το περιεχόμενό του. Μία αλλαγή αυτών των τιμών καθιστά το αρχείο μη προσπελάσιμο αλλά δεν καταστρέφει το περιεχόμενό του. Η μετατροπή της επικεφαλίδας στην αρχική της μορφή οδηγεί το αρχείο να διαβαστεί σωστά και να είναι και πάλι αξιοποιήσιμο.

Τα σύγχρονα εγκληματολογικά εργαλεία και οι αυτοματοποιημένες διαδικασίες που προσφέρουν για την ταυτοποίηση του είδους των αρχείων, δε μπορούν να ανταπεξέλθουν σε αυτή τη μέθοδο απόκρυψης αρχείων. Αυτό συμβαίνει διότι λειτουργούν με βάσεις δεδομένων στις οποίες περιέχεται το σύνολο των γνωστών επικεφαλίδων αρχείων και συνεπώς δε μπορεί να προβλεφτεί η μορφολογία που θα έχει κάποια εξ' αυτών ύστερα από μία χειροκίνητη μεταβολή της.

Στο ακόλουθο παράδειγμα ένα αρχείο jpeg έχει αναλυθεί από έναν επεξεργαστή δεκαεξαδικών τιμών. Παρατηρείται ότι η επικεφαλίδα που φέρει το αρχείο στα πρώτα δέκα bytes της είναι η 0xFF D8 FF E0 xx xx 4A 46 49 46 (όπου τα xx αντιπροσωπεύουν οποιαδήποτε τιμή), επικεφαλίδα που υποδηλώνει αρχεία εικόνας JPEG/JFIF.



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	FF	D8	FF	E0	00	10	4A	46	49	46	00	01	01	00	00	01	ÿØÿà JFIF
00000010	00	01	00	00	FF	DB	00	84	00	09	06	07	12	12	12	15	ÿÛ „
00000020	12	13	12	15	15	15	15	15	17	18	15	16	15	15	15	16	
00000030	16	17	17	15	18	17	16	17	15	15	15	18	1D	28	20	18	(
00000040	1A	25	1D	16	15	21	31	21	25	29	2B	2E	2E	2E	17	1F	% !1!%)+...
00000050	33	38	33	2D	37	28	2D	2E	2B	01	0A	0A	0A	0E	0D	0E	383-7(-.+
00000060	1B	10	10	1A	2D	1F	1F	1F	2B	2D	2D	2D	2D	2B	2D	2D	- +----+--
00000070	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	----------

Τυπική επικεφαλίδα αρχείου JPEG/JFIF

Η εικόνα σε αυτή τη κατάσταση μπορεί να προσπελαστεί με το πρόγραμμα Photo Viewer των Windows απροβλημάτιστα.

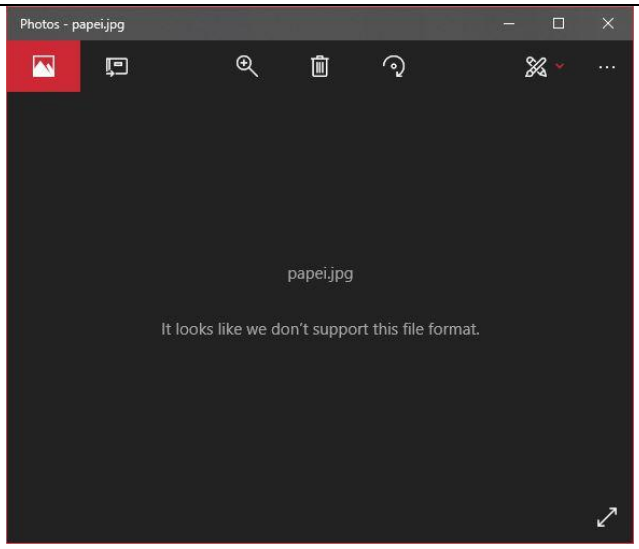
Περιεχόμενο αρχείου εικόνας

Στη συνέχεια η υπογραφή του αρχείου τροποποιείται, με τα πρώτα οκτώ bytes της επικεφαλίδας του να παίρνουν διαδοχικές αριθμητικές τιμές ως ακολούθως:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	01	02	03	04	05	06	07	08	49	46	00	01	01	00	00	01	IF
00000010	00	01	00	00	FF	DB	00	84	00	09	06	07	12	12	12	15	ÿÛ „
00000020	12	13	12	15	15	15	15	15	17	18	15	16	15	15	15	16	
00000030	16	17	17	15	18	17	16	17	15	15	15	18	1D	28	20	18	(
00000040	1A	25	1D	16	15	21	31	21	25	29	2B	2E	2E	2E	17	1F	% !1!%)+...
00000050	33	38	33	2D	37	28	2D	2E	2B	01	0A	0A	0A	0E	0D	0E	383-7(-.+
00000060	1B	10	10	1A	2D	1F	1F	1F	2B	2D	2D	2D	2D	2B	2D	2D	- +----+--
00000070	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	----------

Τροποποιημένη επικεφαλίδα αρχείου εικόνας

Και το αποτέλεσμα στην νέα προσπάθεια προσπέλασης του αρχείου αποφέρει μήνυμα λάθους από το πρόγραμμα ένεκα της αδυναμίας του να αναλύσει την αλλοιωμένη επικεφαλίδα.



Μήνυμα λάθους λόγω τροποποίησης της επικεφαλίδα αρχείου εικόνας

Η επαναφορά της επικεφαλίδας στην αρχική της μορφή έχει ως αποτέλεσμα το αρχείο να είναι ξανά αναγνωρίσιμο από το πρόγραμμα και το περιεχόμενό του να απεικονίζεται κανονικά.

Κάποια εκ των εργαλείων που χρησιμοποιούνται στην Ψηφιακή Εγκληματολογία θα προσφέρουν ως μόνη αυτοματοποιημένη βοήθεια, την επισήμανση αυτών των αρχείων ως

αγνώστου τύπου (Unknown Types) ή απλά δεν θα τα χαρακτηρίσουν. Η επέκταση που τυχόν φέρουν θα αποτελεί το μόνο κριτήριο για την κατάταξή τους ή ομαδοποίησή τους αναλόγως του εγκληματολογικού λογισμικού αλλά όπως αναλύθηκε παραπάνω αυτή θα μπορούσε επίσης να είναι τροποποιημένη.

Η ευθύνη σε αυτές τις περιπτώσεις ανήκει στον εξεταστή ο οποίος θα πρέπει να αναλύσει χειροκίνητα κάθε αρχείο σε επίπεδο bytes. Υπάρχουν τύποι αρχείων που παρέχουν επιπλέον μεταδεδομένα εντός των δομών τους και κατ' επέκταση μπορούν να αναγνωριστούν από αυτές ενώ άλλα αρχεία φέρουν χαρακτηριστικές τιμές κατάληξης (footer) οι οποίες (αν δεν έχουν τροποποιηθεί και αυτές) μπορούν να παρέχουν μία ένδειξη περί του είδους του αρχείου.

Τέλος υπάρχουν τύποι αρχείων με χαρακτηριστική εικόνα δομής στο επίπεδο των bytes η οποία παρουσιάζει επαναληψιμότητα και εν συνεπεία μία οπτική ανάλυση από έναν έμπειρο εξεταστή ενδέχεται να οδηγήσει στην αναγνώριση του είδους του αρχείου. Αφού αυτό επιτευχθεί και πλέον ο εξεταστής έχει ανακαλύψει το είδος του αρχείου που έχει να αντιμετωπίσει, μπορεί με τη σειρά του να αντιστρέψει τη διαδικασία και να τροποποιήσει εκ νέου την επικεφαλίδα του αρχείου με τις αυθεντικές τιμές του. Η όλη διαδικασία πρέπει να καταγραφεί και να δικαιολογηθεί επαρκώς, αφού σε αυτές τις περιπτώσεις πολλές εκ των βασικών αρχών της Ψηφιακής Εγκληματολογίας παραβιάζονται όπως η αλλοίωση της αρχικής εικόνας που είχε το αρχείο όταν το μέσο κατασχέθηκε και της διαφοροποίησης που θα φέρει πλέον η μοναδική αλφαριθμητική του ταυτότητα.

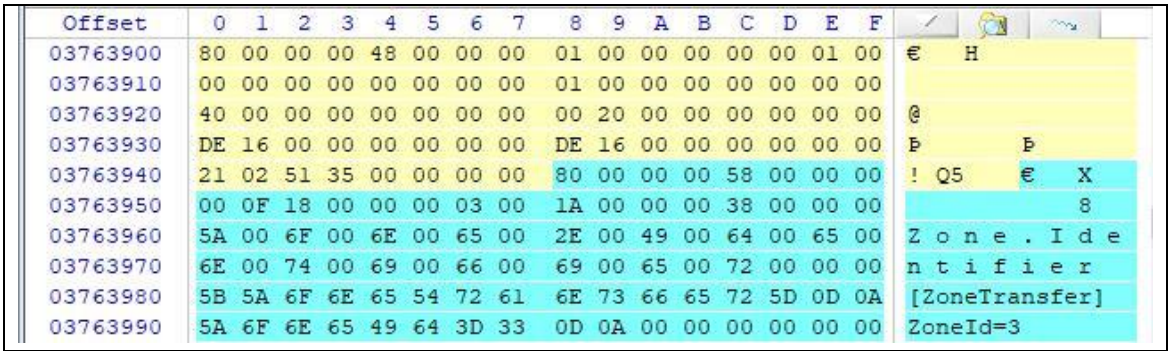
Εναλλακτικές Ροές Δεδομένων (Alternate Data Streams)

Οι εναλλακτικές ροές δεδομένων ADS είναι ένα χαρακτηριστικό του NTFS το οποίο επιτρέπει πολλαπλές πηγές δεδομένων να αποθηκεύονται για ένα αρχείο στο επίπεδο του Συστήματος Αρχείων. Εισήχθη σαν χαρακτηριστικό στο NTFS προκειμένου να συμβάλει στη συμβατότητα του Συστήματος με αρχεία που προέρχονταν από το Σύστημα Αρχείων HFS της Macintosh, αφού το τελευταίο χρησιμοποιεί διακλαδώσεις προμηθειών (resource forks) για να αποθηκεύσει πρόσθετα χαρακτηριστικά σε αρχεία.

Τα πρόσθετα δεδομένα αποθηκεύονται σε χαρακτηριστικά $Data που δημιουργούνται στην εγγραφή αρχείου του $MFT για το αρχείο. Αυτά τα χαρακτηριστικά, σε αντίθεση με τα κυρίως χαρακτηριστικά $Data που αποθηκεύουν το περιεχόμενο του αρχείου, φέρουν ονομασία ροής (Stream Name) η οποία χρησιμοποιείται και ως αναγνωριστικό του είδους της πληροφορίας που προσφέρουν.

Στο Λειτουργικό Windows τα ποιο συνηθισμένα ADS φέρουν την ονομασία "Zone.Identifier" και παρέχουν πληροφόρηση σχετικά με την πηγή από την οποία προήλθε το αρχείο που αποθηκεύτηκε στο μέσο και ποια η βαθμίδα εμπιστοσύνης αυτής.

Στην παρακάτω εικόνα απεικονίζονται τα δύο χαρακτηριστικά $Data ενός αρχείου εικόνας jpeg. Με τον κίτρινο χρωματισμό έχει σημανθεί το χαρακτηριστικό $Data που περιέχει τα πραγματικά δεδομένα του αρχείου. Όπως φαίνεται στη θέση 0x03763909 δεν υπάρχει ονομασία ροής για αυτό. Με τον γαλάζιο χρωματισμό έχει σημανθεί το χαρακτηριστικό $Data που περιέχει το ADS. Η ονομασία του είναι "Zone.Identifier" και το περιεχόμενό του είναι τοπικό. Η τιμή του περιεχομένου του "[ZoneTransfer] ZoneId=3" υποδηλώνει ότι το συγκεκριμένο αρχείο έχει κατωφορτωθεί (download) από το διαδίκτυο.



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03763900	80	00	00	00	48	00	00	00	01	00	00	00	00	00	01	00
03763910	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00	00
03763920	40	00	00	00	00	00	00	00	00	20	00	00	00	00	00	00
03763930	DE	16	00	00	00	00	00	00	DE	16	00	00	00	00	00	00
03763940	21	02	51	35	00	00	00	00	80	00	00	00	58	00	00	00
03763950	00	0F	18	00	00	00	03	00	1A	00	00	00	38	00	00	00
03763960	5A	00	6F	00	6E	00	65	00	2E	00	49	00	64	00	65	00
03763970	6E	00	74	00	69	00	66	00	69	00	65	00	72	00	00	00
03763980	5B	5A	6F	6E	65	54	72	61	6E	73	66	65	72	5D	0D	0A
03763990	5A	6F	6E	65	49	64	3D	33	0D	0A	00	00	00	00	00	00

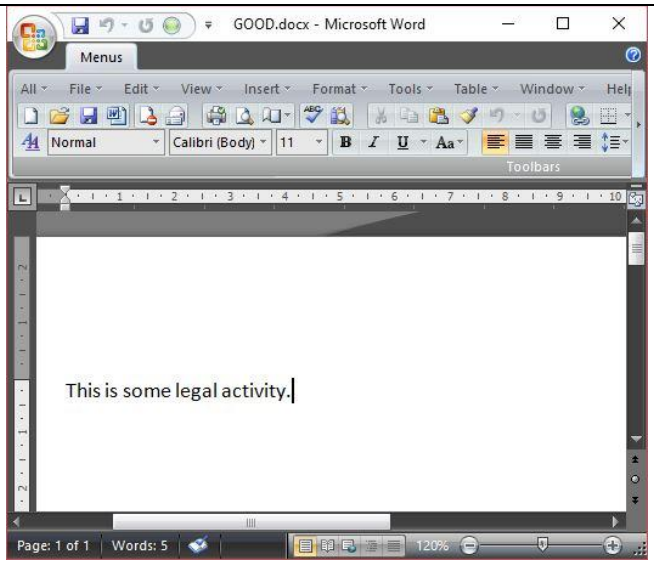
Εναλλακτική ροή δεδομένων για αρχείο εικόνας προερχόμενο από το διαδίκτυο

Η σημαντικότερη πτυχή της υλοποίησης τους εντοπίζεται στο γεγονός ότι τα ADS καίτοι συνοδεύουν το αρχείο δεν αποτελούν μέρος αυτού. Τα ADS περιγράφουν το αρχείο ως μία επιπρόσθετη πηγή πληροφορίας αλλά μόνο σε επίπεδο Συστήματος Αρχείου. Το περιεχόμενό τους ποτέ δεν παρουσιάζεται στο γραφικό περιβάλλον του χρήστη και το μέγεθός τους δεν συμπεριλαμβάνεται στο μέγεθος του αρχείου. Αυτά τα χαρακτηριστικά τους κάνουν τα ADS μία άριστη τοποθεσία απόκρυψης δεδομένων. Πολλά κακόβουλα λογισμικά (malware) εκμεταλλεύονται τις ιδιότητες των ADS για να εισχωρήσουν με κεκαλυμμένη μορφή εντός ενός συστήματος.

Και εάν τα κακόβουλα λογισμικά μπορούν να εντοπιστούν κατόπιν ελέγχου από ένα αντιβιοτικό πρόγραμμα, τι γίνεται με ένα απλό αρχείο που αποθηκεύεται σε μία εναλλακτική ροή δεδομένων; Αυτός ο εναλλακτικός τρόπος απόκρυψης δεδομένων είναι σαφώς πιο εξεζητημένος και απαιτεί βαθύτερη γνώση από αυτή που κατέχει ο μέσος χρήστης αλλά στις περιπτώσεις μίας εγκληματολογικής εξέτασης ο δράστης ενδέχεται να έχει προσπαθήσει αρκετά ώστε να αποκρύψει τα ίχνη ενός ενοχοποιητικού αρχείου.

Υπάρχουν συγκεκριμένα εγκληματολογικά λογισμικά τα οποία παρέχουν αυτοματοποιημένες διαδικασίες ανάλυσης των ADS αλλά ο τεράστιος αριθμός των αρχείων που ενυπάρχουν σε ένα σύγχρονο αποθηκευτικό μέσο καθιστά την περαιτέρω έρευνα των εναλλακτικών ροών σε μεγάλο βαθμό μία διαδικασία χειροκίνητου ελέγχου από τον εξεταστή.

Στο ακόλουθο παράδειγμα θα παρουσιαστεί η δημιουργία ενός ADS και η αποθήκευση σε αυτό ενός αρχείου εικόνας που ενδεχομένως θα μπορούσε να αποτελεί ενοχοποιητικό υλικό. Έστω λοιπόν ότι στον τόμο που εξετάζεται προϋπάρχει ήδη ένα φυσιολογικό, αρχείο κειμένου word το οποίο φέρει καθόλα νόμιμο περιεχόμενο. Το αρχείο ονομάζεται “GOOD.docx” και ιδού η απεικόνιση του περιεχομένου του.



Περιεχόμενο του αρχείου GOOD.docx

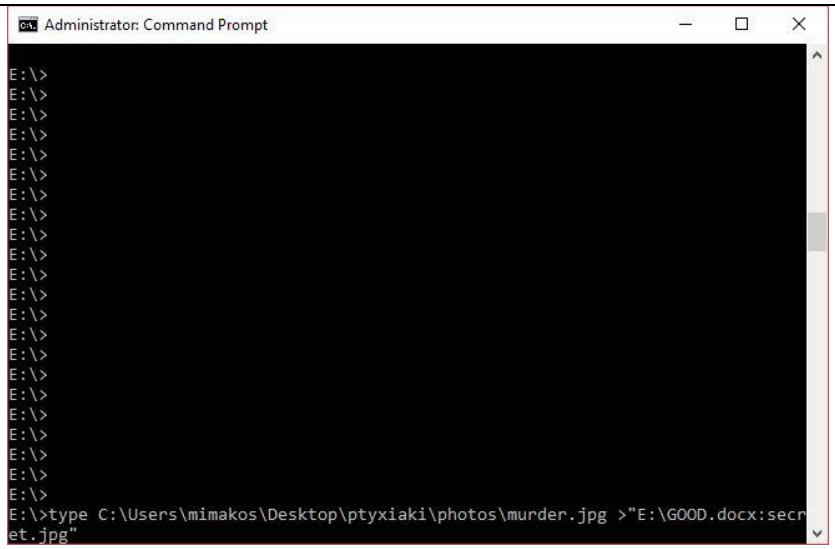
Στην παρούσα φάση το αρχείο δε φέρει εναλλακτικές ροές δεδομένων στην εγγραφή του. Η εγγραφή αρχείου στο $MFT περιλαμβάνει μόνο το απαραίτητα χαρακτηριστικά. Με γαλάζιο χρώμα υπάρχει το χαρακτηριστικό $File_Name οπού εντοπίζεται η ονομασία του αρχείου “GOOD.docx” στη θέση 0x037644F2. Ακολουθεί το χαρακτηριστικό $Object_ID με αναγνωριστικό κωδικό 0x40 00 00 00 που έχει σημανθεί με πράσινο χρώμα και προσφέρει πληροφόρηση σχετικά τα μοναδικά αναγνωριστικά του αρχείου.

Παρατηρείται ότι σεσημασμένο με ροζ χρώμα είναι το χαρακτηριστικό $Data του αρχείου το οποίο τυγχάνει μη τοπικό, δεν περιλαμβάνει ονομασία ροής και φέρει στη θέση 0x03764570 τη λίστα data runs για την υπόδειξη της τοποθεσίας του περιεχομένου του στον τόμο. Στη θέση 0x03764560 φαίνεται το λογικό μέγεθος του αρχείου με την δεκαεξαδική τιμή 0x27 39 η οποία ισούται με 10.041 bytes. Αμέσως μετά το πέρας του χαρακτηριστικού $Data, στη θέση 0x03764580 εντοπίζεται η τιμή 0xFF FF FF FF που σηματοδοτεί τη λήξης της εγγραφής αρχείου.

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03764480	00	00	00	00	08	01	00	00	00	00	00	00	00	00	00	00
03764490	48	C8	00	00	00	00	00	00	30	00	00	00	70	00	00	00
037644A0	00	00	00	00	00	00	04	00	54	00	00	00	18	00	01	00
037644B0	05	00	00	00	00	00	05	00	F8	B2	67	88	BA	FD	D3	01
037644C0	AA	7D	10	A4	BA	FD	D3	01	57	69	1C	A4	BA	FD	D3	01
037644D0	38	1B	0E	A4	BA	FD	D3	01	00	30	00	00	00	00	00	00
037644E0	39	27	00	00	00	00	00	00	20	00	00	00	00	00	00	00
037644F0	09	00	47	00	4F	00	4F	00	44	00	2E	00	64	00	6F	00
03764500	63	00	78	00	00	00	00	00	40	00	00	00	28	00	00	00
03764510	00	00	00	00	00	00	05	00	10	00	00	00	18	00	00	00
03764520	49	A0	EC	C3	B8	68	E8	11	83	44	00	8C	FA	9E	51	9B
03764530	80	00	00	00	50	00	00	00	01	00	00	00	00	00	03	00
03764540	00	00	00	00	00	00	00	00	02	00	00	00	00	00	00	00
03764550	40	00	00	00	00	00	00	00	00	30	00	00	00	00	00	00
03764560	39	27	00	00	00	00	00	00	39	27	00	00	00	00	00	00
03764570	21	01	6A	07	21	02	E9	2D	00	00	00	00	00	00	00	00
03764580	FF	FF	FF	FF	82	79	47	11	00	00	00	00	00	00	00	00

Εγγραφή αρχείου για το αρχείο GOOD.docx

Στο αρχείο κειμένου GOOD.docx θα προστεθεί ως ADS το αρχείο εικόνας murder.jpg. Αυτό θα επιτευχθεί μέσω εντολής στη γραμμή εντολών cmd των Windows.



Εντολή δημιουργίας ADS για το αρχείο GOOD.docx

Με την ανωτέρω εντολή το αρχείο εικόνας murder.jpg διοχετεύεται μέσω του pipe '>' στο αρχείο κειμένου GOOD.docx ενώ ο τελεστής ':' υποδεικνύει ότι αυτό θα συμβεί με τη μορφή μίας εναλλακτικής ροής δεδομένων ADS με την ονομασία "secret.jpg". Το περιεχόμενο του εγγράφου, μετά την εντολή, έχει παραμείνει απαράλλαχτο ενώ κανένα αρχείο εικόνας δεν εμφανίζεται πουθενά στον τόμο κατά την εξερεύνησή του στο γραφικό περιβάλλον.

Η εγγραφή αρχείου για το αρχείο GOOD.docx φέρει πλέον την ακόλουθη μορφή:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
037644E0	39	27	00	00	00	00	00	00	20	00	00	00	00	00	00	00
037644F0	09	00	47	00	4F	00	4F	00	44	00	2E	00	64	00	6F	00
03764500	63	00	78	00	00	00	00	00	40	00	00	00	28	00	00	00
03764510	00	00	00	00	00	00	05	00	10	00	00	00	18	00	00	00
03764520	49	A0	EC	C3	B8	68	E8	11	83	44	00	8C	FA	9E	51	9B
03764530	80	00	00	00	50	00	00	00	01	00	00	00	00	00	03	00
03764540	00	00	00	00	00	00	00	00	02	00	00	00	00	00	00	00
03764550	40	00	00	00	00	00	00	00	00	30	00	00	00	00	00	00
03764560	39	27	00	00	00	00	00	00	39	27	00	00	00	00	00	00
03764570	21	01	6A	07	21	02	E9	2D	00	00	00	00	00	00	00	00
03764580	80	00	00	00	60	00	00	00	01	0A	40	00	00	00	07	00
03764590	00	00	00	00	00	00	00	00	02	00	00	00	00	00	00	00
037645A0	58	00	00	00	00	00	00	00	00	30	00	00	00	00	00	00
037645B0	50	23	00	00	00	00	00	00	50	23	00	00	00	00	00	00
037645C0	73	00	65	00	63	00	72	00	65	00	74	00	2E	00	6A	00
037645D0	70	00	67	00	00	00	00	00	21	03	55	35	00	00	00	00
037645E0	FF	FF	FF	FF	82	79	47	11	00	00	00	00	00	00	00	00

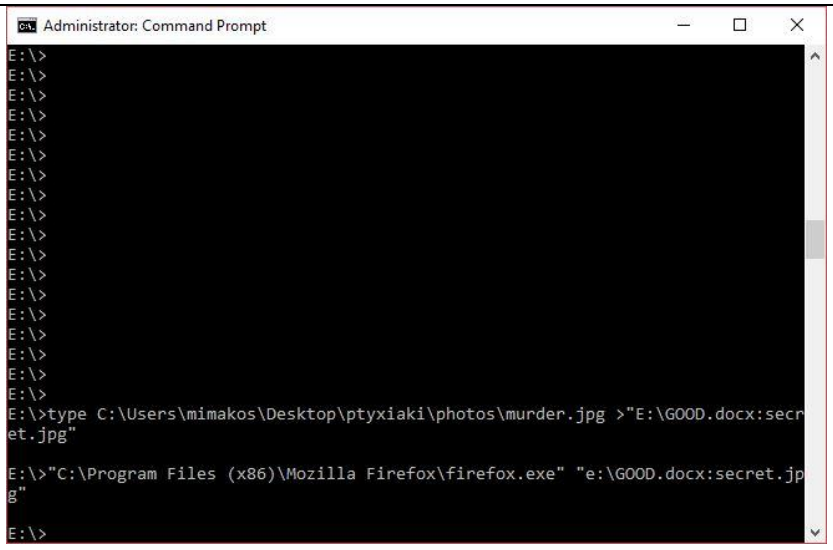
Εγγραφή αρχείου για το αρχείο GOOD.docx μετά τη δημιουργία ADS σε αυτό

Παρατηρείται ότι πλέον στη θέση 0x03764580 έχει προστεθεί έτερο χαρακτηριστικό $Data. Το συγκεκριμένο χαρακτηριστικό φέρει ονομασία ροής και έχει το περιεχόμενο του μη τοπικό. Η

run list για αυτό δίνεται στη θέση 0x037645D8 και υποδεικνύει που βρίσκεται αποθηκευμένο το αρχείο εικόνας στον τόμο. Το αρχείο εικόνας murder.jpg δεν μπορεί να εντοπιστεί και να προσπελαστεί στο γραφικό περιβάλλον αφού δεν απεικονίζεται σε αυτό από το Λειτουργικό Σύστημα.

Ο πρώτος τρόπος για να προσπελαστεί είναι να ανακτηθεί ως διαγραμμένο αρχείου του οποίου η εγγραφή αρχείου διασώζεται, ακολουθώντας τη μεθοδολογία που περιγράφηκε στο κεφάλαιο 2.3.

Ο δεύτερος τρόπος είναι να προσπελαστεί από κατάλληλη εφαρμογή μέσω εντολής του cmd των Windows. Στο παράδειγμα θα κληθεί ο φυλλομετρητής διαδικτύου Firefox για να απεικονίσει το περιεχόμενο του ADS σύμφωνα με την δεύτερη εντολή που έχει πραγματοποιηθεί κατά σειρά και απεικονίζεται στην ακόλουθη εικόνα:

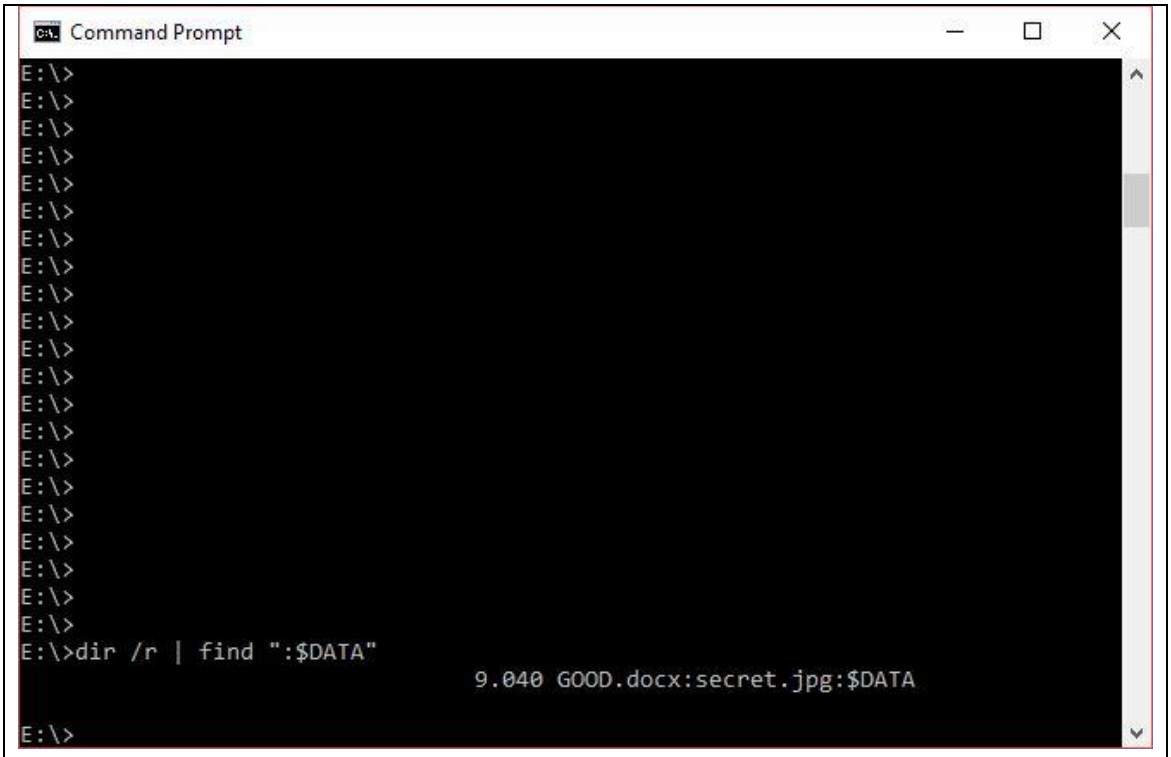


Εντολή κλήσης φυλλομετρητή για την προσπέλαση του ενσωματωμένου αρχείου στο ADS

Η εντολή πρέπει να παρέχει την πλήρη διαδρομή του αρχείου και να υποδεικνύει τη συγκεκριμένη ονομασία ροής για το ζητούμενο ADS, καθώς ένα αρχείο ενδέχεται να φέρει πολλαπλές εναλλακτικές ροές δεδομένων στην εγγραφή του. Το αποτέλεσμα της εντολής απεικονίζεται στην παρακάτω εικόνα:

Περιεχόμενο του ενσωματωμένου αρχείου στο ADS του εγγράφου GOOD.docx

Τέλος ο έλεγχος για την ύπαρξη εναλλακτικών ροών δεδομένων στα αρχεία ενός καταλόγου ή τόμου επιτυγχάνεται μέσω της κάτωθι εντολής:



Εντολή εμφάνισης όλων των ADS τύπου $Data που υπάρχουν στον τόμο

Παρατηρείται ότι με τη σχετική εντολή εμφανίζονται μόνο τα αρχεία του τόμου ή καταλόγου που έχουν εναλλακτικές ροές δεδομένων, αναλύεται ο τύπος του ADS και το ακριβές μέγεθός του.

Η πλέον σημαντική πτυχή της υλοποίησης των εναλλακτικών ροών δεδομένων για τη Ψηφιακή Εγκληματολογία, είναι η απόκρυψη του μεγέθους του αρχείου που δύναται να ενσωματωθεί σε ένα ADS. Στο προηγούμενο παράδειγμα παρατηρείται ότι το μέγεθος του αρχείου κειμένου GOOD.docx παραμένει ακριβώς το ίδιο και μετά τη δημιουργία ADS σε αυτό. Αυτή η ιδιότητα των εναλλακτικών ροών δεδομένων αποτελεί τον ιδανικότερο παράγοντα ώστε αυτές να χρησιμοποιηθούν ως δομές απόκρυψης κακόβουλης πληροφορίας και κατά συνέπεια να επιφέρουν μεγάλο βαθμό πολυπλοκότητας στη διενεργούμενη εξέταση επί του μέσου.

Τροποποίηση Υπογραφής Εγγραφής Αρχείου

Τα πρώτα τέσσερα bytes κάθε εγγραφής αρχείου στο $MFT φέρουν την χαρακτηριστική δεκαεξαδική τιμή 0x46 49 4C 45 η οποία αναπαρίσταται με τους ASCII χαρακτήρες “FILE” και υποδηλώνει την έναρξη μίας έγκυρης εγγραφής αρχείου.

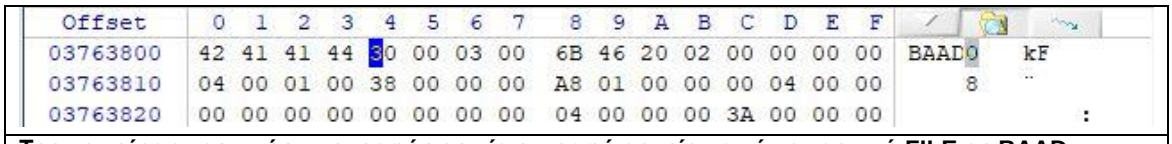
Όπως προαναφέρθηκε το Σύστημα Αρχείων NTFS χρησιμοποιεί την τεχνική ελέγχου τιμών Fix-up για να εντοπίζει κατεστραμμένους ή προβληματικούς τομείς στον πίνακα $MFT. Όταν οι δύο τιμές Fix-up που αποθηκεύονται στο τέλος κάθε τομέα δε συμφωνούν μεταξύ τους, τότε το Σύστημα θεωρεί τους παρακάτω τομείς ως προβληματικούς και επισημαίνει την καταχώριση του $MFT ως μη λειτουργική αντικαθιστώντας την υπογραφή “FILE” με την αντίστοιχη “BAAD” (0x42 41 41 44).

Ο χρήστης μπορεί να πραγματοποιήσει ο ίδιος αυτή την αλλαγή και να εκμεταλλευτεί τις επιπτώσεις της ώστε να αποκρύψει πληροφορία σε αυτή την εγγραφή, αφού το Σύστημα προσφέρει τις ιδανικές συνθήκες για την επιτυχή απόκρυψη ενός αρχείου σε αυτή την περίπτωση. Αυτό συμβαίνει γιατί το Σύστημα εντοπίζοντας την τιμή “BAAD” προσπερνάει τη συγκεκριμένη καταχώριση κάθε φορά που αναλύει το $MFT.

Επομένως, αφενός μεν οποιοδήποτε αρχείο περιέχεται στην εγγραφή δεν παρουσιάζεται στο γραφικό περιβάλλον αφού η εγγραφή του δεν αναλύεται ώστε το Σύστημα να μάθει που βρίσκεται αποθηκευμένο και να το εμφανίσει. Αφετέρου, η εγγραφή δεν κινδυνεύει να επαναγραφεί, και συνεπώς τα στοιχεία του αρχείου να χαθούν, αφού το Σύστημα θεωρεί τους τομείς της εγγραφής ελαττωματικούς και δεν τους χρησιμοποιεί ξανά.

Επιπροσθέτως, το αρχείο μεταδεδομένων $Bitmap έχει ήδη χαρακτηρίσει τα clusters που κατάλαβε το κρυφό αρχείο στον τόμο ως κατανεμημένα, πράξη που αντιστρέφεται μόνο κατά την φυσιολογική διαδικασία διαγραφής ενός αρχείου και κατ’ επέκταση και της εγγραφής του από το $MFT. Ως εκ τούτου τα cluster του αρχείου παραμένουν κατειλημμένα για το σύστημα και δεν υπάρχει κίνδυνος επαναχρησιμοποίησης τους από άλλο αρχείο και διαγραφής του αρχείου που επιδιώκεται να αποκρυφτεί.

Η χειροκίνητη αλλαγή της υπογραφής της εγγραφής αρχείου στο παρακάτω παράδειγμα, έχει ως αποτέλεσμα το αρχείο εικόνας διαμόρφωσης jpeg, που αυτή περιγράφει, να “εξαφανιστεί” από το γραφικό περιβάλλον χρήστη. Το αρχείο συνεχίζει να καταλαμβάνει χώρο στον τόμο και τα δεδομένα του βρίσκονται άθικτα στην αρχική τους θέση.



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03763800	42	41	41	44	30	00	03	00	6B	46	20	02	00	00	00	00
03763810	04	00	01	00	38	00	00	00	A8	01	00	00	00	04	00	00
03763820	00	00	00	00	00	00	00	00	04	00	00	00	3A	00	00	00

Τροποποίηση της τιμής υπογραφής σε μία εγγραφή αρχείου από την αρχική FILE σε BAAD

Ένας τρόπος ελέγχου για να διαπιστωθεί η χρήση αυτού του τρόπου εναλλακτικής απόκρυψης δεδομένων είναι να αναζητηθεί η δεκαεξαδική τιμή της υπογραφής αυτών των τεχνητών ελαττωματικών εγγραφών. Εφόσον διαπιστωθεί η ύπαρξη τέτοιων εγγραφών, στη συνέχεια θα πρέπει να αναλυθεί το αρχείο μεταδεδομένων $BadClus που αποθηκεύει τις θέσεις

των ελαττωματικών clusters του τόμου. Η σύγκριση των αριθμών των cluster που αναφέρονται στο αρχείο Συστήματος $BadClus θα πρέπει να συμφωνούν με τη θέση των τομέων που ανευρέθησαν στο $MFT. Στην περίπτωση χειροκίνητης αλλαγής της επικεφαλίδας “FILE” κάτι τέτοιο δε συμβαίνει οπότε αποδεικνύεται η χρήση της τεχνική παραποίησης.

Κενές Εγγραφές Αρχείου $MFT

Το αρχείο $MFT αποτελεί ένα από τα σημαντικότερα αρχεία μεταδεδομένων του Συστήματος Αρχείων NTFS. Επειδή η οποιαδήποτε τροποποίηση επί των δεδομένων αυτού θα μπορούσε να προκαλέσει εξαιρετική ζημιά στο Σύστημα, προστατεύεται από τη φύση του ως κρυφό αρχείο συστήματος και συνεπώς η διάδραση του χρήστη με αυτό δεν είναι δυνατή στο γραφικό περιβάλλον. Το χαρακτηριστικό του αυτό το καθιστά έναν χώρο δυνητικής απόκρυψης δεδομένων.

Οι καταχωρίσεις του πίνακα $MFT στις θέσεις 16 έως 23 είναι κενές περιεχομένου και θα μπορούσαν να χρησιμοποιηθούν για την απόκρυψη αρχείων. Αυτή η ενέργεια δε θα μπορούσε να πραγματοποιηθεί μέσω του γραφικού περιβάλλοντος του χρήστη αλλά μέσω ενός hex editor που δύναται να τροποποιήσει τα δεδομένα ενός τόμου σε επίπεδο byte.

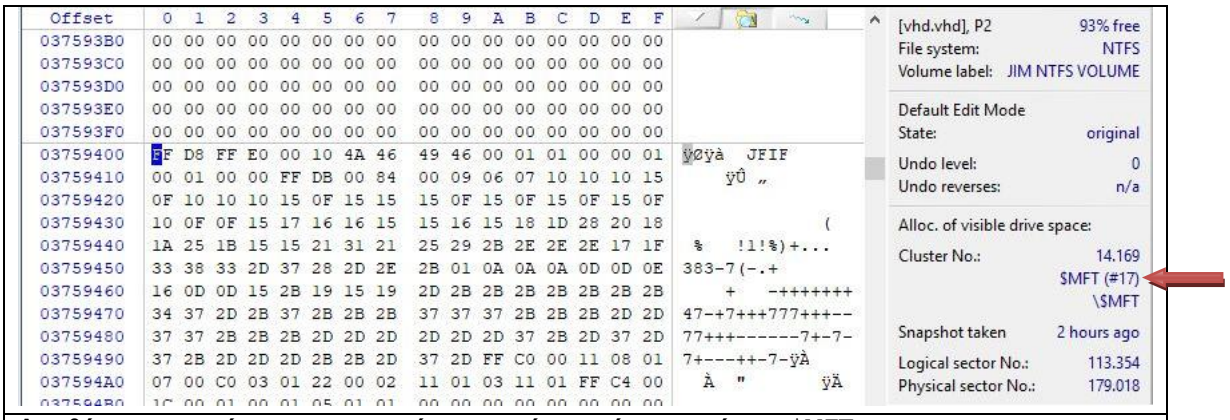
Σε αυτή την περίπτωση, κυριότερο περιορισμό αποτελεί το μικρό μέγεθος που προσφέρει αυτός ο χώρος, καθώς δεδομένου ότι κάθε εγγραφή στην τωρινή έκδοση του NTFS είναι 1024 bytes, ο μέγιστος χώρος απόκρυψης δεδομένων που προσφέρει το $MFT στις θέσεις αυτές είναι συνολικά 8KB.

Εάν τα δεδομένα ενός αρχείου είναι μικρότερα από αυτό το μέγεθος, τότε μπορούν να αντιγραφούν στις οικίες θέσεις και να ενυπάρχουν στο μέσο χωρίς ποτέ το αρχείο να εμφανιστεί στο γραφικό περιβάλλον, λόγω της προφύλαξης που απολαμβάνει το $MFT από το Σύστημα.

Η εν λόγω θέση αποθήκευσης προσφέρει ασφάλεια και για την περίπτωση της απώλειας του αρχείου λόγω επικάλυψής του από τα δεδομένα κάποιου άλλου, καθώς καίτοι οι καταχωρίσεις αυτές είναι κενές περιεχομένου, τα clusters τους είναι σημασμένα ως κατανεμημένα στο αρχείο $Bitmap και ως αυτού δεν μπορούν να κατανεμηθούν σε άλλο αρχείο.

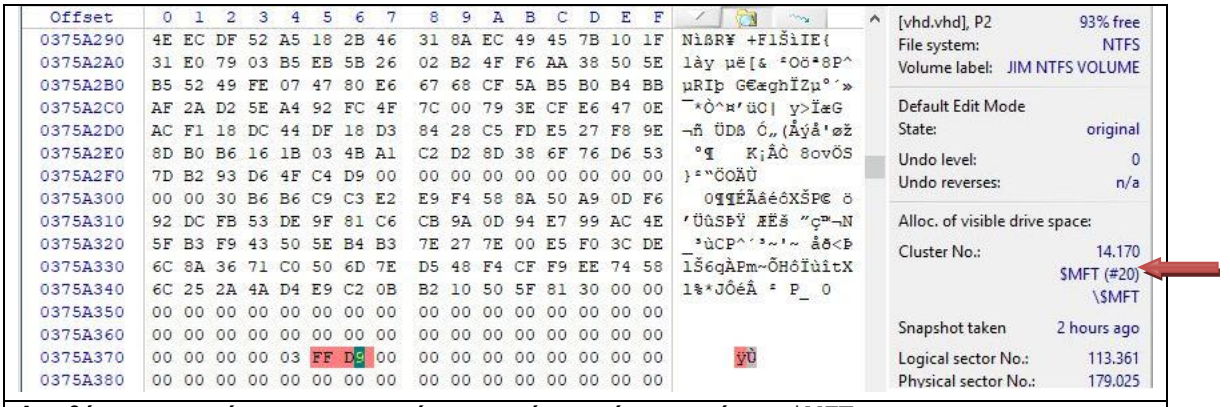
Αρχεία που έχουν αποθηκευτεί σε αυτόν τον ανενεργό χώρο του $MFT μπορούν να ανακτηθούν μόνο με τεχνικές χάραξης δεδομένων. Το γεγονός που κάνει αυτή την εναλλακτική τεχνική απόκρυψης δεδομένων ιδιαίτερα παραπλανητική είναι ότι οι εξεταστές συνήθως βασίζονται σε εγκληματολογικά λογισμικά για την ανάκτηση των αρχείων του κατανεμημένου χώρου και ενεργούν τις διάφορες μεθόδους χάραξης μόνο στον μη κατανεμημένο χώρο του υπό διερεύνηση τόμου. Επομένως, το εγκληματολογικό εργαλείο δεν πρόκειται να εντοπίσει κάποια εγγραφή που να περιγράφει το κρυφό αρχείο και συνεπώς δεν θα το παρουσιάσει στον εξεταστή, ενώ η χάραξη δεδομένων στον μη κατανεμημένο χώρο θα ανακτήσει μόνο αρχεία που βρίσκονται εκεί και κατ’ επέκταση όχι το κρυμμένο αρχείο.

Στην παρακάτω εικόνα παρουσιάζονται τα δεδομένα ενός αρχείου εικόνας διαμόρφωσης jpeg, μεγέθους 4KB, τα οποία έχουν αποθηκευτεί στην αρχή της εγγραφής #17 του $MFT. Στη θέση 0x03759400 εντοπίζεται η επικεφαλίδα (Header) των αρχείων jpeg με τιμή 0xFF D8. Γίνεται λοιπόν ξεκάθαρο ότι με αυτό τον τρόπο δε δημιουργούνται δομές χαρακτηριστικών όπως συμβαίνει σε κάθε εγγραφή, αντιθέτως τα πραγματικά δεδομένα του αρχείου αποθηκεύονται απευθείας στον χώρο. Παρατηρείται ότι ο προηγούμενος τομέας που καταλήγει στη θέση 0x 037593FF και αντιστοιχεί στην εγγραφή αρχείου #16 είναι κενός περιεχομένου.



Αποθήκευση αρχείου στις ανενεργές εγγραφές αρχείου του πίνακα $MFT

Τα δεδομένα του αρχείου καταλήγουν στην εγγραφή #20, καταλαμβάνοντας τέσσερις συνολικά καταχωρίσεις από τις συνολικά οκτώ ανενεργές του πίνακα $MFT. Στη θέση 0x0375A375 παρατηρείται η τιμή κατάληξης (Footer) των αρχείων jpeg η οποία υποδηλώνει το τέλος του αρχείου.



Αποθήκευση αρχείου στις ανενεργές εγγραφές αρχείου του πίνακα $MFT

Ο ανωτέρω τόμος εκκινείται (mount) απροβλημάτιστα από το Λειτουργικό Σύστημα και το αρχείο jpeg παραμένει εμφωλιασμένο στον πίνακα $MFT του και ανύπαρκτο οπουδήποτε αλλού στο γραφικό περιβάλλον.

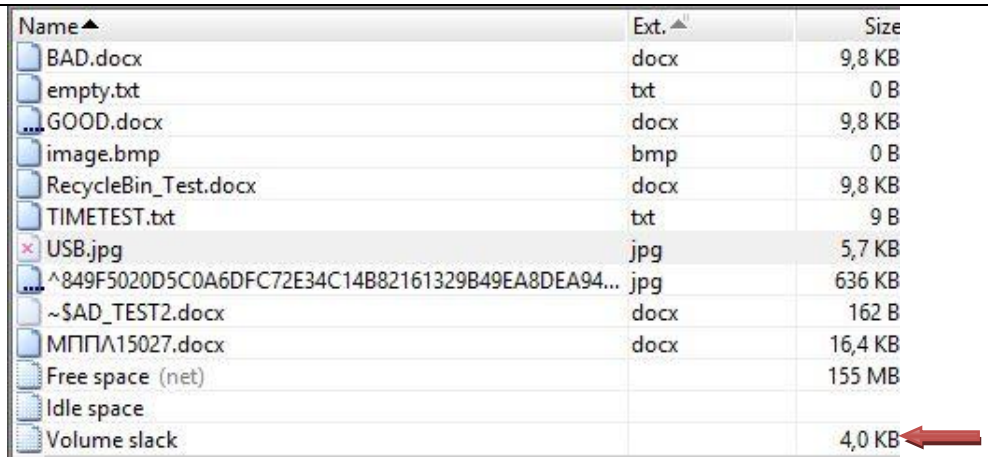
Τα εγκληματολογικά λογισμικά δεν θα εμφανίσουν στον τόμο, κανένα από τα αρχεία που έχουν αποθηκευτεί στον υπό ανάλυση χώρο του $MFT καθόσον ελλείπει κάθε μορφή πληροφόρησης για αυτά και δεν υποστηρίζεται καμία δομή παρουσίασης τους στο εγκληματολογικό, γραφικό περιβάλλον. Η ανακάλυψη της κρυμμένης πληροφορίας μπορεί να συμβεί μόνο είτε κατόπιν προσεκτικής οπτικής παρατήρησης από τον εξεταστή, ή κατόπιν εντολής χάραξης δεδομένων από το σύνολο του τόμου και όχι μόνο του μη κατανεμημένου χώρου.

Αν και εμπορικά εργαλεία που μπορούν να προσφέρουν ένα γραφικό περιβάλλον για αποθήκευση αρχείων στις κενές θέσεις του πίνακα $MFT ενός Συστήματος σε λειτουργία δεν υπάρχουν, είναι δυνατό με έναν διορθωτή hex editor ο κακόβουλος χρήστης να επιχειρήσει να αποκρύψει δεδομένα εκεί.

Κατάλοιπος Χώρος Τόμου (Volume Slack)

Κατάλοιπος χώρος τόμου αποκαλείται η περιοχή η οποία ενώ βρίσκεται εντός της διαμέρισης (Partition) δεν αποτελεί μέρος του τόμου (Volume). Αποτελεί τον μη χρησιμοποιούμενο χώρο μεταξύ του τελευταίου cluster του Συστήματος Αρχείων και της διαμέρισης στην οποία το Σύστημα αυτό ενυπάρχει.

Οι περισσότεροι σύγχρονοι τόμοι έχουν ορισμένο μέγεθος cluster που ισούται με οκτώ τομείς (4096 bytes). Εάν το μέγεθος του τόμου δεν είναι πλήρως διαιρέσιμο με αυτόν τον αριθμό, τότε παραμένουν τομείς στο τέλος του, που αποτελούν μέρος της διαμέρισης αλλά όχι του τόμου.

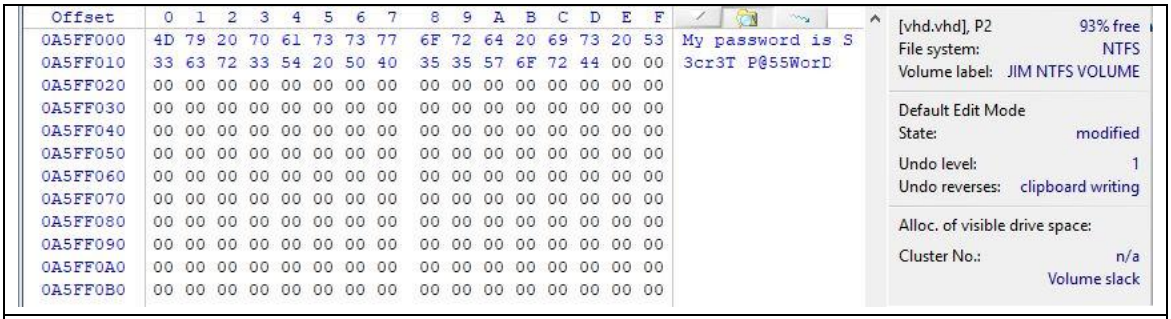


Volume Slack μεγέθους 4,0KB

Εξαιτίας του τρόπου δημιουργίας του, ο κατάλοιπος χώρος τόμου θα είναι πάντα μικρότερος του μεγέθους του cluster που έχει οριστεί για τον τόμο. Σε περίπτωση που το μέσο έχει διαμορφωθεί και πάλι με τη διαδικασία του Format και τα μεγέθη των τόμων έχουν τροποποιηθεί, δύναται να ανευρεθούν δεδομένα από την προηγούμενη διαμόρφωση σε αυτό το μέρος.

Το volume slack παρουσιάζει μεγάλο ενδιαφέρον και αποτελεί ιδανικό εναλλακτικό χώρο απόκρυψης δεδομένων λόγω της ίδιας του της φύσης. Ως χώρος που δεν αποτελεί κομμάτι του Συστήματος Αρχείων κάθε τυχόν περιεχόμενο του δεν δύναται να απεικονιστεί στο γραφικό περιβάλλον χρήστη αλλά ούτε και να επικαλυφτεί από άλλο αρχείο.

Στην ακόλουθη εικόνα απεικονίζεται το περιεχόμενο ενός αρχείου σημειώματος που έχει αποκρυφτεί εντός του κατάλοιπου χώρου τόμου.



Πληροφορία αποθηκευμένη στον χώρο Volume Slack

Το μειονέκτημα αυτής της τεχνικής απόκρυψης δεδομένων είναι ότι ο κατάλοιπος χώρος τόμου έχει πολύ μικρό μέγεθος και η εξέτασή του είναι εύκολη, λόγω έλλειψης οποιασδήποτε μορφής δομών σε αυτό. Παρά το μειονέκτημα αυτό, αυτή η τεχνική και κατ' επέκταση τα δεδομένα που αποκρύπτει, δεν εντοπίζονται αυτομάτως από κανένα εγκληματολογικό λογισμικό, αφού ο κατάλοιπος χώρος τόμου θεωρείται ως ένα μέρος ήσσονος σημασίας στο αποθηκευτικό μέσο και καμία αυτόματη ανάλυση δεν λαμβάνει χώρα επί αυτού, άλλη από την αρχική του αναγνώριση στο μέσο. Συνεπώς, αν και εμπορικά εργαλεία που μπορούν να προσφέρουν ένα γραφικό περιβάλλον για αποθήκευση αρχείων στο Volume Slack ενός live Συστήματος δεν υπάρχουν, είναι δυνατό με έναν διορθωτή hex editor ο κακόβουλος χρήστης να επιχειρήσει να αποκρύψει δεδομένα εκεί.

Με αυτό το μέρος περαιώνεται η ανάλυση δομών δεδομένων που φέρει το Σύστημα Αρχείων NTFS και συνάμα το θεωρητικό μέρος της παρούσης. Στη συνέχεια ακολουθεί το τρίτο κεφάλαιο με την πρακτική ανάλυση παραδειγμάτων των εννοιών που παρουσιάστηκαν στα δύο προηγούμενα κεφάλαια.

Κεφάλαιο 3ο: Πρακτική Ανάκτηση Δεδομένων

3.1 Ανάκτηση Συνεχούς Αρχείου (Contiguous File Recovery)

Στο παρόν μέρος θα ανακτηθεί ένα αρχείο εικόνας διαμόρφωσης jpeg το οποίο βρίσκεται αποθηκευμένο σε διαδοχικά clusters του τόμου. Τα σύγχρονα αποθηκευτικά μέσα με τους τεράστιους χώρους αποθήκευσης σε συνδυασμό με τις νέες εκδόσεις Συστημάτων Αρχείων τα οποία είναι προσανατολισμένα στην αποδοτικότητα και την ταχύτητα, έχουν ως απόρροια τη δημιουργία συνεχών αρχείων κατά την αποθήκευση αυτών στον τόμο.

Ένα συνεχές αρχείο συμβάλλει στην αποδοτικότητα του Συστήματος καθώς ο χρόνος ανάκτησης των δεδομένων του μειώνεται δραστικά όταν το Σύστημα διαβάζει συνεχόμενους τομείς του δίσκου για να το προσπελάσει.

Θετικό στοιχείο των συνεχών αρχείων σε σχέση με την Ψηφιακή Εγκληματολογία είναι ότι αυτά είναι πλήρως ανακτήσιμα ακόμα και σε περίπτωση επανεγγραφής της καταχώρισης τους στο $MFT, καθώς οι τεχνικές χάραξης δύναται να τα επανακτήσουν από τον μη κατανεμημένο χώρο με απλές διαδικασίες και ελάχιστη πιθανότητα αποτυχίας.

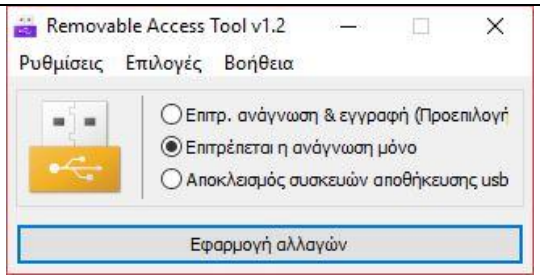
Γενικές Πληροφορίες Εξέτασης

Τα αρχεία που θα ανακτηθούν στο παρόν κεφάλαιο περιλαμβάνονται σε φορητό μέσο αποθήκευσης διασύνδεσης USB, το οποίο αποτελεί το ψηφιακό πειστήριο της πρακτικής αυτής υπόθεσης. Προς τήρηση των βασικών αρχών της Ψηφιακής Εγκληματολογίας το πειστήριο θα πρέπει να προστατευτεί από τυχόν αλλοίωση κατά τη διενέργεια της εργαστηριακής εξέτασης και επομένως ένα εγκληματολογικό αντίγραφο θα πρέπει να αποκτηθεί από αυτό.

Η μέθοδος φραγμού που θα επιλεχθεί κατά τη διαδικασία απόκτησης αντιγράφου είναι ο φραγμός εγγραφής λογισμικού, με τη χρήση του προγράμματος Ratool v.1.2. Τα εγκληματολογικά λογισμικά που θα χρησιμοποιηθούν για τη διενέργεια της εξέτασης είναι τα FTK Imager (x86) v.3.1.0.1514 της εταιρείας AccessData και Winhex (x86) v.18.3 της εταιρείας X-Ways Forensics.

1ο Στάδιο Εξέτασης

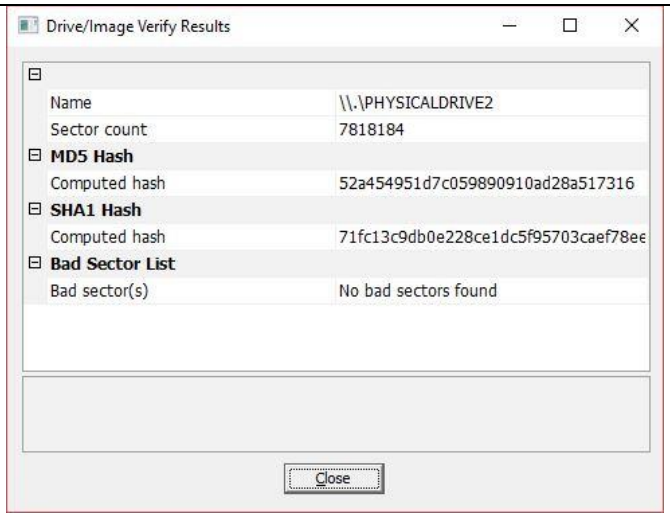
Στο πρώτο μέρος της εξέτασης θα ενεργοποιηθεί ο φραγμός εγγραφής λογισμικού στον υπολογιστή και πριν από οποιαδήποτε ενέργεια επί του πειστηρίου θα δοκιμαστεί η εύρυθμη λειτουργία του σε εργαστηριακό υλικό δοκιμών.



Ενεργοποίηση κατάστασης λειτουργίας Read Only

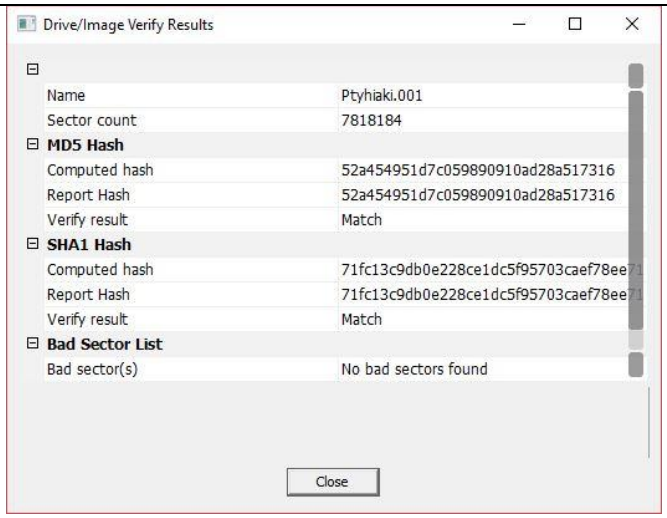
Αφού διαπιστωθεί η λειτουργικότητα του φραγμού επί του υλικού δοκιμής του εργαστηρίου, το πειστήριο θα μπορέσει να εισαχθεί με ασφάλεια στον υπολογιστή εξέτασης. Σε αυτό το σημείο το λογισμικό φραγμού θα μπλοκάρει οποιαδήποτε ενέργεια εγγραφής εκτελεί αυτοματοποιημένα το Λειτουργικό Σύστημα (όπως της ανάθεσης αναγνωριστικού ID στο μέσο, πληροφορία η οποία εγγράφεται άμεσα στον τομέα εκκίνησής του κατά τη διαδικασία mount), και θα διατηρήσει την ακεραιότητα του αρχικού πειστηρίου.

Εν συνεχεία θα ενεργηθεί επί του πειστηρίου η πρώτη διαδικασία ψηφιακής επικύρωσης. Το εγκληματολογικό εργαλείο FTK Imager θα χρησιμοποιηθεί ώστε να εξαχθεί η μοναδική αλφαριθμητική ταυτότητα του μέσου. Θα χρησιμοποιηθούν δύο διαφορετικοί αλγόριθμοι ψηφιακής επικύρωσης, οι MD5 και SHA1.



Αποτέλεσμα ψηφιακής επικύρωσης του πειστηρίου αποθηκευτικού μέσου

Επόμενο βήμα είναι η δημιουργία του εγκληματολογικού αντιγράφου του πειστηρίου. Το εγκληματολογικό εργαλείο FTK Imager θα χρησιμοποιηθεί ώστε να αποκτηθεί η εικόνα (image) του μέσου. Ο τύπος αντιγράφου που επιλέχθηκε είναι της μορφής επίπεδου αρχείου (“flat file”) ακατέργαστων δεδομένων (raw) dd με επέκταση αρχείου .001. Το εγκληματολογικό λογισμικό υπολόγισε εκ νέου την ψηφιακή ταυτότητα του αποκτημένου αρχείου αντιγράφου και επιβεβαίωσε την αντιστοιχία με την αρχικά υπολογισμένη τιμή του πειστηρίου. Το αποτέλεσμα αυτό αποδεικνύει ότι το αποκτηθέν αντίγραφο αποτελεί πανομοιότυπη εικόνα του αρχικού πειστηρίου και συνεπώς κάθε εξέταση επί αυτού ισοδυναμεί με την εξέταση του αυθεντικού πειστηρίου μέσου.



Αποτέλεσμα ψηφιακής επικύρωσης του αποκτηθέντος εγκληματολογικού αντιγράφου

Σε αυτό το σημείο το προκαταρκτικό στάδιο της εξέτασης έχει περαιωθεί. Το αρχικό πειστήριο μπορεί να αποθηκευτεί σύμφωνα με τις προδιαγραφές που χαρακτηρίζουν κάθε εργαστήριο. Κάθε περαιτέρω ανάλυση θα λάβει χώρα επί του εγκληματολογικού αντιγράφου, προσφέροντας έτσι ασφάλεια στο αρχικώς κατασχεθέν μέσο από οποιαδήποτε αλλοίωση τυχόν συμβεί κατά τη διάρκεια της εξέτασης.

Σημειώνεται ότι το 1^{ο} στάδιο εξέτασης είναι το ίδιο για όλα τα μέρη του παρόντος κεφαλαίου και συνεπώς η παρουσίαση του δε θα επαναληφθεί στη συνέχεια για κάθε διαφορετικό μέρος.

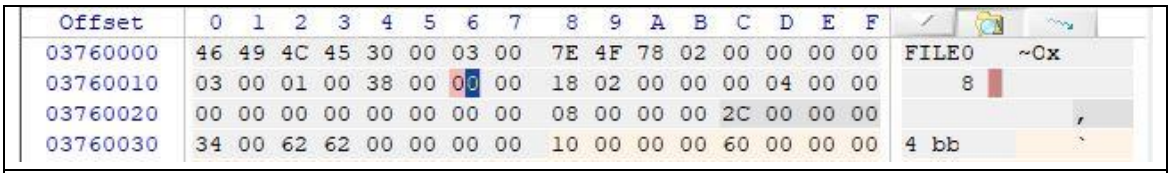
2ο Στάδιο Εξέτασης

Στο παρόν στάδιο θα λάβει χώρα η πραγματική ψηφιακή εγκληματολογική εξέταση επί των δεδομένων του υπό εξέταση μέσου.

Το αρχείο που επιθυμούμε να ανακτήσουμε έχει διαγραφεί από τον χρήστη (ολική διαγραφή Shift+Delete και όχι ανακύκλωση). Η εγγραφή αρχείου του όμως παραμένει άθικτη στον πίνακα $MFT, γεγονός που συνεπάγεται ότι εφόσον και τα clusters που είχαν κατανεμηθεί αρχικά σε αυτό δεν έχουν επαναχρησιμοποιηθεί για την αποθήκευση έτερου αρχείου, το αρχείο είναι πλήρως ανακτήσιμο. Ας αναλύσουμε εκτενέστερα την εγγραφή αρχείου του στο $MFT.

Επικεφαλίδα Εγγραφής Αρχείου

Η επικεφαλίδα της εγγραφής αρχείου δίνεται στην παρακάτω εικόνα:



Επικεφαλίδα Εγγραφής Αρχείου για το αρχείο PHOTO.jpg

Διακρίνονται οι πιο σημαντικές τιμές:

- Υπογραφή – 0x46 49 4C 45 = FILE

- Θέση διάταξης Fix-up – 0x30 (Little Endian) = 48 bytes από την αρχή του χαρακτηριστικού
- Αριθμός καταχωρίσεων στη διάταξη Fix-up – 0x03 = τρεις διαφορετικές τιμές μεγέθους δύο bytes η καθεμία στη διάταξη Fix-up
- Άθροισμα διαδοχής – 0x03 = Η εγγραφή αρχείου έχει χρησιμοποιηθεί τρείς φορές συνολικά
- Θέση πρώτου χαρακτηριστικού – 0x38 = 56 bytes από την αρχή της εγγραφής (ουσιαστικά το μήκος της επικεφαλίδας εγγραφής)
- Σημαία (Flag) κατάστασης χρήσης – 0x00 = διαγραμμένο αρχείο
- Λογικό μέγεθος της εγγραφής αρχείου (Logical) – 0x0218 = 536 bytes χρησιμοποιούνται συνολικά από την εγγραφή
- Αριθμός εγγραφής αρχείου στο $MFT – 0x2C = καταχώριση υπ' αριθμόν 44 στον πίνακα $MFT

Η υπογραφή "FILE" της εγγραφής υποδηλώνει ότι πρόκειται για μία καταχώρηση του πίνακα $MFT χωρίς σφάλματα και συνεπώς μπορεί να αξιοποιηθεί για επιπλέον ανάλυση. Αυτό μπορεί να επαληθευθεί και μέσω της εξέτασης της διάταξης Fix-up.

Η διάταξη Fix-up εντοπίζεται 48 bytes από την αρχή του χαρακτηριστικού, στη θέση 0x03760030. Η επικεφαλίδα της εγγραφής δηλώνει ότι στη διάταξη υπάρχουν τρείς τιμές. Κάθε τιμή φέρει μήκος δύο bytes και η πρώτη καταχώριση είναι πάντοτε η τιμή ελέγχου που έχει δημιουργήσει αυτομάτως το Σύστημα.

Στο παράδειγμα η τιμή ελέγχου είναι η 0x34 00. Αυτή η τιμή έχει αντικαταστήσει τα δύο τελευταία bytes κάθε τομέα της εγγραφής και δεδομένου ότι η εγγραφή φέρει μέγεθος δύο τομέων (1024 bytes), στη συνέχεια της διάταξη υπάρχουν οι άλλες δύο τιμές που έχουν αντικατασταθεί από την τιμή ελέγχου. Η πρώτη εξ' αυτών είναι η τιμή 0x62 62 ενώ η δεύτερη είναι η κενή τιμή 0x00 00. Αυτό δικαιολογείται εάν ληφθεί υπόψη το λογικό μέγεθος της εγγραφής που φέρει μήκος 536 bytes. Συνεπώς ένας τομέας (512 bytes) έχει χρησιμοποιηθεί πλήρως και τα τελευταία δύο byte του με τιμή 0x62 62 έχουν αντικατασταθεί από την τιμή ελέγχου και μεταφερθεί στη διάταξη. Από τον δεύτερο τομέα χρησιμοποιούνται μόλις τα πρώτα 24 bytes και συνεπώς τα μηδενικής τιμής δύο τελευταία του έχουν επίσης αντικατασταθεί από την τιμή ελέγχου και μεταφερθεί στη διάταξη.

Η σημαία κατάστασης χρήσης στη θέση 0x03760016 φέρει την τιμή 0x00 00 και υποδηλώνει ότι η εγγραφή ανήκει σε ένα διαγεγραμμένο αρχείο. Ακολουθώντας την πληροφόρηση της επικεφαλίδας θα πλοηγηθούμε στο πρώτο κατά σειρά χαρακτηριστικό για να συνεχίσουμε την ανάλυση της εγγραφής.

Χαρακτηριστικό $Standard_Information

Το χαρακτηριστικό $Standard_Information αποτελεί το πρώτο κατά σειρά χαρακτηριστικό της εγγραφής. Ακολουθεί την επικεφαλίδα τηρώντας το όριο θέσης του πολλαπλάσιου των 8 byte. Αρχίζει στη θέση 0x03760038 και η δομή του απεικονίζεται παρακάτω:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03760030	34	00	62	62	00	00	00	00	10	00	00	00	60	00	00	00
03760040	00	00	00	00	00	00	00	00	48	00	00	00	18	00	00	00
03760050	85	43	C8	88	A6	EB	D3	01	EF	3C	0B	8A	E6	EC	D3	01
03760060	97	69	DD	60	00	FF	D3	01	86	DA	08	8A	E6	EC	D3	01
03760070	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
03760080	00	00	00	00	0D	01	00	00	00	00	00	00	00	00	00	00
03760090	C8	F1	00	00	00	00	00	00	30	00	00	00	70	00	00	00

χαρακτηριστικό $Standard_Information για το αρχείο PHOTO.jpg

Διακρίνονται οι πιο σημαντικές τιμές:

- Κωδικός χαρακτηριστικού – 0x10 00 00 00 = $Standard_Information
- Μήκος του χαρακτηριστικού – 0x60 (Little Endian) = 96 bytes συνολικά (συμπεριλαμβανομένου της επικεφαλίδας χαρακτηριστικού, της επικεφαλίδας τοπικού περιεχομένου και του περιεχομένου)
- Σημαία μη τοπικού περιεχομένου – 0x00 = τοπικό περιεχόμενο
- Σημαίες αρχείου (Flags) – 0x00 = Το αρχείο δεν αποτελεί συμπιεσμένο, κρυπτογραφημένο ή αραιό αρχείο.
- Θέση του περιεχομένου – 0x18 = 24 bytes από την αρχή του χαρακτηριστικού
- Χρονοσφραγίδα δημιουργίας – 0x01 D3 EB A6 88 C8 43 85 = FILETIME σε UTC
- Χρονοσφραγίδα τροποποίησης – 0x01 D3 EC E6 8A 0B 3C EF = FILETIME σε UTC
- Χρονοσφραγίδα τροποποίησης εγγραφής $MFT – 0x01 D3 FF 00 60 DD 69 97= FILETIME σε UTC
- Σημαίες είδους αρχείου – 0x20 = αρχείο έτοιμο προς αρχειοθέτηση (archive)

Το περιεχόμενο του χαρακτηριστικού είναι πάντοτε τοπικό. Η εξέτασή του δίνει πληροφόρηση για το χρονολόγιο διάδρασης μεταξύ του χρήστη και του αρχείου.

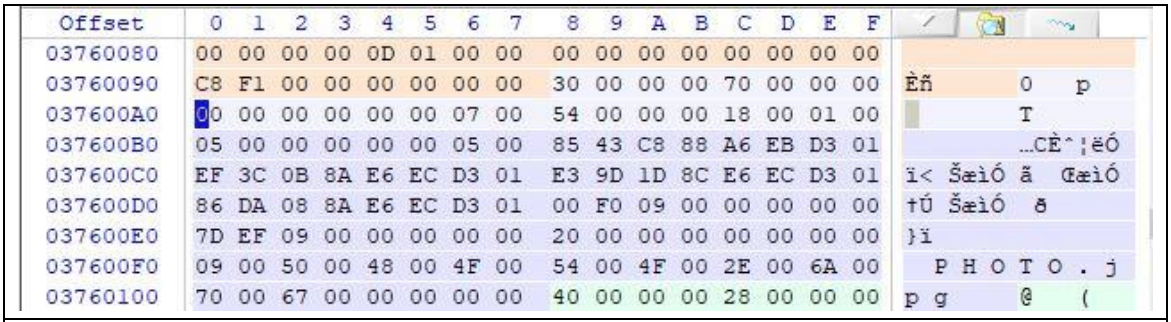
Η χρονοσφραγίδα δημιουργίας φέρει την τιμή 0x01 D3 EB A6 88 C8 43 85 με μορφή FILETIME σε UTC. Η μετάφραση της δίνει την τιμή 14-05-2018, 17:11:01 που είναι η τιμή που το αρχείο εικόνας δημιουργήθηκε στον παρών τόμο. Η χρονοσφραγίδα τροποποίησης φέρει την τιμή 0x01 D3 EC E6 8A 0B 3C EF με μορφή FILETIME σε UTC. Η μετάφρασή της δίνει την τιμή 16-05-2018, 07:21:42. Η διαφορά μεταξύ των δύο χρονικών στιγμών δεν μπορεί να δώσει σίγουρα αποτελέσματα για το αν το αρχείο δημιουργήθηκε αρχικά σε αυτόν τον τόμο ή έχει μεταφερθεί από κάποιον άλλον. Απορρέει παρόλαυτα χρήση του αρχείου σε μεταγενέστερο χρόνο από τη δημιουργία του στον τόμο. Ο συνδυασμός έτερων ευρημάτων στο μέσο (αρχεία LNK, JumpLists κλπ) θα μπορούσε δυνητικά να προσφέρει πληροφόρηση για το ποιο πρόγραμμα χρησιμοποιήθηκε για να τροποποιηθεί το αρχείο και ενδεχομένως ποιο ακριβώς ήταν το είδος της τροποποίησης.

Η χρονοσφραγίδα τροποποίησης εγγραφής $MFT φέρει την τιμή 0x01 D3 FF 00 60 DD 69 97 με μορφή FILETIME σε UTC. Η μετάφραση της δίνει την τιμή 08-06-2018, 08:12:01 που σηματοδοτεί την ημεροχρονολογία και ώρα που το αρχείο διεγράφη καθώς η τροποποίηση της σημαίας κατάστασης χρήσης στην επικεφαλίδα της εγγραφής προκειμένου να δείχνει το αρχείο ως διαγραμμένο, είναι και η τελευταία χρονικά αλλαγή που επήλθε στην εγγραφή $MFT του αρχείου.

Αν και οι παραπάνω πληροφορίες δεν είναι απαραίτητες για την ανάκτηση του αρχείου παίζουν καίριο ρόλο σε μία εγκληματολογική εξέταση και συνεπώς συμπεριλήφθησαν στο παρόν πρακτικό μέρος. Γίνεται μνεία ότι οι ανωτέρω χρονοσφραγίδες πρέπει να μετατραπούν στην τοπική ώρα στην οποία είναι ρυθμισμένη η συσκευή προκειμένου να γίνει σωστή ερμηνεία τους.

Χαρακτηριστικό $File_Name

Το χαρακτηριστικό $File_Name αποτελεί το δεύτερο κατά σειρά χαρακτηριστικό της εγγραφής. Ακολουθεί την επικεφαλίδα τηρώντας το όριο θέσης του πολλαπλάσιου των 8 byte. Αρχίζει στη θέση 0x03760098 και η δομή του απεικονίζεται παρακάτω:



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03760080	00	00	00	00	0D	01	00	00	00	00	00	00	00	00	00	00
03760090	C8	F1	00	00	00	00	00	00	30	00	00	00	70	00	00	00
037600A0	00	00	00	00	00	00	07	00	54	00	00	00	18	00	01	00
037600B0	05	00	00	00	00	00	05	00	85	43	C8	88	A6	EB	D3	01
037600C0	EF	3C	0B	8A	E6	EC	D3	01	E3	9D	1D	8C	E6	EC	D3	01
037600D0	86	DA	08	8A	E6	EC	D3	01	00	F0	09	00	00	00	00	00
037600E0	7D	EF	09	00	00	00	00	00	20	00	00	00	00	00	00	00
037600F0	09	00	50	00	48	00	4F	00	54	00	4F	00	2E	00	6A	00
03760100	70	00	67	00	00	00	00	00	40	00	00	00	28	00	00	00

χαρακτηριστικό $File_Name για το αρχείο PHOTO.jpg

Διακρίνονται οι πιο σημαντικές τιμές:

- Κωδικός χαρακτηριστικού – 0x30 00 00 00 = $File_Name
- Μήκος του χαρακτηριστικού – 0x70 (Little Endian) = 112 bytes συνολικά (συμπεριλαμβανομένου της επικεφαλίδας χαρακτηριστικού, της επικεφαλίδας τοπικού περιεχομένου και του περιεχομένου)
- Σημαία μη τοπικού περιεχομένου – 0x00 = τοπικό περιεχόμενο
- Σημαίες αρχείου (Flags) – 0x00 = Το αρχείο δεν αποτελεί συμπιεσμένο, κρυπτογραφημένο ή αραιό αρχείο
- Θέση του περιεχομένου – 0x18 = 24 bytes από την αρχή του χαρακτηριστικού
- Αριθμός εγγραφής αρχείου $MFT του γονικού καταλόγου – 0x05 = #5 καταχώριση στον πίνακα $MFT για τον γονικό κατάλογο του αρχείου
- Αριθμός διαδοχής του γονικού καταλόγου – 0x05= Η εγγραφή του γονικού καταλόγου στο $MFT έχει χρησιμοποιηθεί πέντε συνολικά φορές
- Χρονοσφραγίδα τροποποίησης εγγραφής $MFT – 0x01 D3 EC E6 8C 1D 9D E3 = FILETIME σε UTC
- Μέγεθος ονόματος αρχείου – 0x09 = Εννέα χαρακτήρες Unicode
- Τύπος ονόματος αρχείου – 0x00 = ονομασία τύπου Posix
- Ονομασία αρχείου – 0x50 00 48 00 4F 00 54 00 4F 00 2E 00 6A 00 70 00 67 00 = “PHOTO.jpg”

Ο αριθμός εγγραφής αρχείου $MFT του γονικού καταλόγου ισούται με 5 και επομένως το αρχείο εικόνας που εξετάζουμε βρισκόταν αποθηκευμένο στον ριζικό κατάλογο Root.

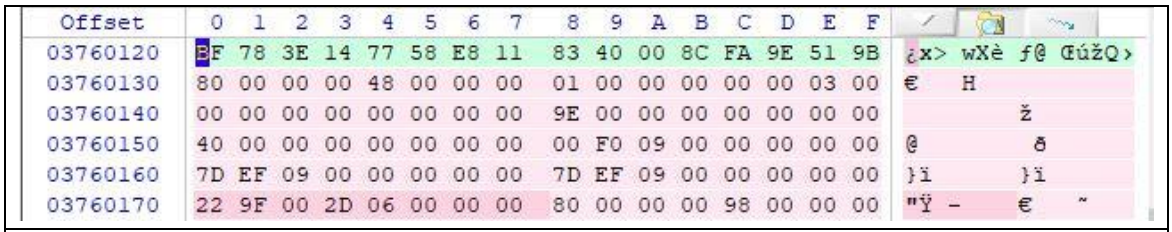
Μεταβαίνοντας στην εγγραφή του ριζικού καταλόγου και αναλύοντας την επικεφαλίδα της διαπιστώνουμε ότι το άθροισμα διαδοχής του Root ισούται με τον αριθμό 5. Κατά συνέπεια ισούται και με τον αριθμό που δίνει το χαρακτηριστικό $File_Name του αρχείου εικόνας για αριθμό διαδοχής του γονικού του καταλόγου. Συμπερασματικά η σχέση γονέα-παιδιού είναι ενεργή και το αρχείο εικόνας θα μπορούσε να τοποθετηθεί στη σωστή θέση της ιεραρχικής δομής του Συστήματος Αρχείων αν επιδιώκαμε κάτι τέτοιο.

Η χρονοσφραγίδα τροποποίησης εγγραφής $MFT μεταφράζεται στην τιμή 16-05-2018, 07:21:46 η οποία διαφέρει από την αντίστοιχη τιμή που φέρει το χαρακτηριστικό $Standard_Information. Αποδεικνύεται ότι η χρονοσήμανση του χαρακτηριστικού $File_Name δεν είναι αξιόπιστη και ακολουθεί την τελευταία τροποποίηση του αρχείου μη λαμβάνοντας υπόψη την ενέργεια διαγραφής του που έλαβε χώρα σε μεταγενέστερο χρόνο.

Το μέγεθος του ονόματος του αρχείου είναι εννέα χαρακτήρες Unicode και άρα αποθηκεύεται στο μέσο χρησιμοποιώντας συνολικά 18 bytes, δύο για την απόδοση του κάθε χαρακτήρα. Ο χώρος ονομασίας (namespace) είναι POSIX λόγω της μίξης κεφαλαίων και μικρών γραμμάτων στο όνομα του αρχείου, το οποίο ξεκινάει στη θέση 0x037600F2 με τιμή “PHOTO.jpg”.

Χαρακτηριστικό $Data

Το χαρακτηριστικό $Data αποτελεί το τέταρτο κατά σειρά χαρακτηριστικό της εγγραφής. Αρχίζει στη θέση 0x03760130 και η δομή του απεικονίζεται παρακάτω:



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03760120	BF	78	3E	14	77	58	E8	11	83	40	00	8C	FA	9E	51	9B
03760130	80	00	00	00	48	00	00	00	01	00	00	00	00	00	03	00
03760140	00	00	00	00	00	00	00	00	9E	00	00	00	00	00	00	00
03760150	40	00	00	00	00	00	00	00	00	F0	09	00	00	00	00	00
03760160	7D	EF	09	00	00	00	00	00	7D	EF	09	00	00	00	00	00
03760170	22	9F	00	2D	06	00	00	00	80	00	00	00	98	00	00	00

χαρακτηριστικό $Data για το αρχείο PHOTO.jpg

Διακρίνονται οι πιο σημαντικές τιμές:

- Κωδικός χαρακτηριστικού – 0x80 00 00 00 = $File_Name
- Μήκος του χαρακτηριστικού – 0x48 (Little Endian) = 72 bytes συνολικά
- Σημαία μη τοπικού περιεχομένου – 0x01 = μη τοπικό περιεχόμενο
- Αριθμός έναρξης εικονικού cluster (VCN) της λίστας data run – 0x00 = Το πρώτο εικονικό cluster φέρει τον αριθμό 0
- Αριθμός λήξης εικονικού cluster (VCN) της λίστας data run – 0x9E = Το τελευταίο εικονικό cluster φέρει τον αριθμό 158
- Θέση της λίστας data runs – 0x40 = 64 bytes από την αρχή του χαρακτηριστικού
- Κατανεμημένο μέγεθος του αρχείου – 0x09 F0 00 = 651.264 bytes
- Πραγματικό μέγεθος του αρχείου – 0x09 EF 7D = 651.133 bytes
- Αρχικοποιημένο μέγεθος του αρχείου – 0x09 EF 7D = 651.133 bytes
- Λίστα data runs – 0x22 9F 00 2D 06

Το περιεχόμενο του χαρακτηριστικού είναι μη τοπικό οπότε δεδομένου ότι το αρχείο είναι διαγεγραμμένο, αυτό θα πρέπει να ανακτηθεί από τον μη κατανεμημένο χώρο του τόμου με τη βοήθεια της λίστας data runs.

Επίσης επειδή το περιεχόμενο του αρχείου είναι μη τοπικό, έχουν αποδοθεί από το Σύστημα εικονικοί αριθμοί στα clusters που το απαρτίζουν, ώστε να επιτυγχάνεται η σωστή επανασύνδεσή τους κάθε φορά που το αρχείο πρέπει να προσπελαστεί. Στη συγκεκριμένη περίπτωση το αρχείο είναι συνεχές, ήτοι καταλαμβάνει διαδοχικά clusters στο μέσο για την αποθήκευσή του.

Ο αριθμός έναρξης και λήξης των εικονικών clusters μας δίνουν τα συνολικά cluster που καταλαμβάνει το αρχείο στον τόμο. Ο αριθμός έναρξης είναι μηδέν και ο αριθμός λήξης είναι 158, που μεταφράζεται ως 159 συνολικά κατανεμημένα clusters στο αρχείο αφού το υπ' αριθμόν 0 cluster αποτελεί μία επιπλέον μονάδα η οποία προστίθεται στα υπόλοιπα 158 για να δώσει το σύνολο.

Η επαλήθευση αυτού του μεγέθους γίνεται από τη σύγκριση του κατανεμημένου μεγέθους του αρχείου στο δίσκο (physical size). Αφού το αρχείο καταλαμβάνει 159 συνολικά clusters και το μέγεθος του κάθε cluster είναι 4096 bytes (όπως αυτό προκύπτει από την ανάλυση του τομέα εκκίνησης), τότε 159 x 4096 = 651.264 bytes, αριθμός που συμφωνεί με το κατανεμημένο μέγεθος του αρχείου.

Το πραγματικό μέγεθος (logical size) του αρχείου στον τόμο είναι μικρότερο του κατανεμημένου, υποδεικνύοντας την ύπαρξη χώρου υπολείμματος αρχείου στο τέλος αυτού. Τέλος το πραγματικό μέγεθος και το αρχικοποιημένο μέγεθος ισούνται γεγονός που αποδεικνύει ότι το αρχείο βρίσκεται αποθηκευμένο στην ολότητά του στον τόμο.

Λίστα Data Runs

Το τελευταίο στάδιο για να ανακτήσουμε το αρχείο είναι ο εντοπισμός του στον μη κατανεμημένο χώρο του τόμου. Η run list του χαρακτηριστικού $Data παρέχει όλη την απαραίτητη πληροφορία για την επίτευξη αυτού του σκοπού. Ακολουθεί η ανάλυσή της:

22 9F 00 2D 06 00. Η επικεφαλίδα 0x22 δίνει το μήκος της καταχώρισης: 2+2= 4 bytes.

- Το περιεχόμενο της καταχώρισης είναι 9F 00 2D 06 00

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 2

- Άρα το μέγεθος του fragment είναι 0x9F 00 (Little Endian) = 159 clusters. Συμπεραίνουμε λοιπόν από αυτόν τον αριθμό, ότι το αρχείο βρίσκεται ολόκληρο σε αυτό το κομμάτι διαδοχικών clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη θέση 0x2D 06 (Little Endian) = 1.581 clusters, που ταυτίζεται με το λογικό cluster του τόμου (LCN) αφού εξετάζουμε την πρώτη καταχώριση της λίστας

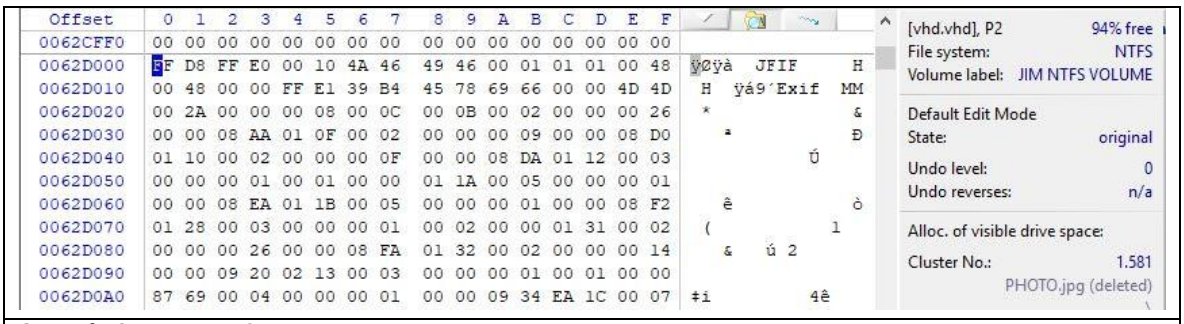
Το τελευταίο cluster που καταλαμβάνει αυτό το fragment καθορίζεται αν στη θέση έναρξης του προστεθεί ο αριθμός των clusters που το απαρτίζουν και αφαιρεθεί μία μονάδα. Αφαιρούμε μία μονάδα αφού το αρχικό cluster προσμετρείται και αυτό στο ολικό μέγεθος. Ήτοι, 1.581 + 159 – 1 = 5.980 cluster είναι το τελικό cluster που καταλαμβάνει η τρέχουσα καταχώριση της Run List. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
159	1.581	1.581	1.739
Καταχώριση 1 στη λίστα Data Runs			

Το επόμενο byte στην λίστα είναι το 0x00, δηλαδή η μηδενική επικεφαλίδα η οποία υποδηλώνει το τέλος της λίστας data runs.

Συνοψίζοντας όλα τα ανωτέρω, για να ανακτηθεί το αρχείο θα πρέπει να μεταβούμε στο λογικό cluster 1.581 του τόμου και να το επιλέξουμε μαζί με τα επόμενα 158 διαδοχικά clusters τα οποία θα αποτελούν και το σύνολο του αρχείου.

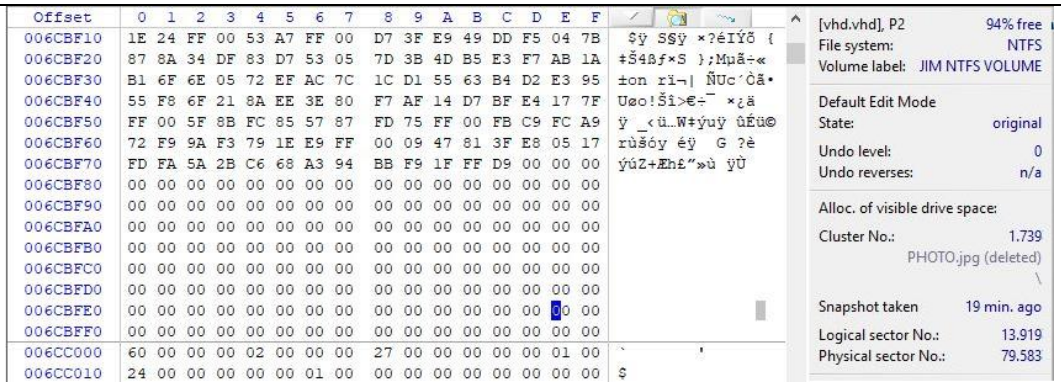
Χρησιμοποιώντας το εγκληματολογικό εργαλείο που έχουμε στη διάθεσή μας μεταβαίνουμε στο λογικό cluster 1.581 του τόμου όπου και διαπιστώνουμε στην αρχή του πρώτου του τομέα την υπογραφή των αρχείων εικόνας jpeg 0xFF D8.



Αρχικό cluster του fragment

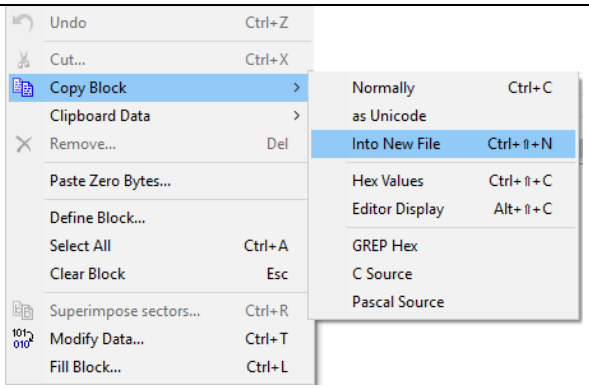
Δίνοντας στο εγκληματολογικό εργαλείο εντολή επιλογής 158 clusters επιπλέον από το αρχικό σημείο, μεταβαίνουμε στο λογικό cluster 1.739 του τόμου το οποίο αποτελεί και το τελευταίο cluster του αρχείου.

Η διαφορά του πραγματικού από τον κατανεμημένο χώρο, 651.264 – 651.133 = 131 bytes υποδεικνύει ότι το αρχείο φέρει μόνο RAM Slack στον τελευταίο τομέα του τελευταίου cluster του που ισούται με 131 bytes. Αυτό το γεγονός επαληθεύεται και από τα bytes που εντοπίζονται στη θέση 0x006CBF7B με τιμή 0xFF D9 που αποτελούν την σημαία λήξης (footer) ενός αρχείου εικόνας jpeg και τα οποία αφήνουν υπόλοιπο ακριβώς 131 bytes μέχρι το τέλος του τομέα. Τα bytes αυτά περιέχουν την κενή τιμή 0x00 με την οποία έχει γεμίσει τον χώρο υπολείμματος του αρχείου το Σύστημα. Στον επόμενο τομέα στη θέση 0x006CC000 ξεκινάει νέο cluster και παρατηρείται η ύπαρξη έτερου αρχείου.



Τελικό cluster του fragment

Αφού όλα τα clusters του αρχείου έχουν εντοπιστεί και επιλεγεί, εν συνεχεία απομένει μόνο η αντιγραφή αυτού του επιλεγμένου μπλοκ και η εξαγωγή του σε ένα νέο αρχείο στον υπολογιστή εξέτασης ως ανακτημένο πλέον αρχείο.



Αντιγραφή μπλοκ επιλεγμένων bytes σε νέο αρχείο

Τα αρχεία εικόνας jpeg έχουν την ιδιότητα να μην “διαβάζουν” οποιαδήποτε άλλη πληροφορία μετά το footer τους. Συνεπώς δεν υπάρχει καμία διαφορά στη μορφή του τελικού εξαχθέντος αρχείου είτε εάν επιλεχθούν όλα τα bytes του τελευταίου τομέα και τελικά ανακτηθεί το πραγματικό μέγεθος ή αν επιλεχθούν τα bytes μέχρι την τιμή footer και ανακτηθεί το φυσικό μέγεθός του.

Τελικό ανακτημένο αρχείο PHOTO.jpg

3.2 Ανάκτηση Κατακερματισμένου Αρχείου (Fragmented File Recovery)

Αρχεία μεγάλου μεγέθους που τείνουν να τροποποιούνται αρκετές φορές μετά τη δημιουργία τους και τα οποία συνυπάρχουν στον ίδιο τόμο με το Λειτουργικό Σύστημα είναι πολύ πιθανό να κατακερματιστούν. Το περιεχόμενο των κατακερματισμένων αρχείων κατανέμεται σε διαφορετικά μπλοκ διαδοχικών clusters τα οποία ονομάζονται κομμάτια fragments. Η ανάκτηση κατακερματισμένων αρχείων συνίσταται στην προσπάθεια εντοπισμού όλων των διαφορετικών του fragments εντός του τόμου και η επανασύνδεσή τους με τη σωστή σειρά ώστε να αναδημιουργηθεί το αρχικό αρχείο.

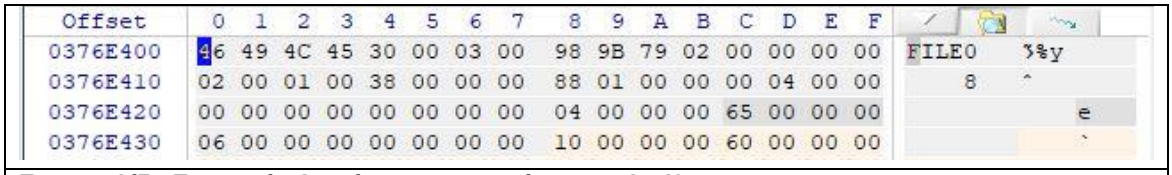
Αυτού του είδους τα αρχεία παρουσιάζουν μεγάλες δυσκολίες στην ανάκτησή τους καθώς η ταυτόχρονη κατάληψη πλήθους διαφορετικών τοποθεσιών του τόμου μπορεί ευκολότερα να οδηγήσει στην επανεγγραφή κάποιου από τα μέρη τους και συνεπώς να καταστήσει το αρχείο μερικώς ανακτήσιμο ή ακόμα και ολικά μη ανακτήσιμο αναλόγως του μέρους που έχει απολέσει. Επιπροσθέτως, σε περίπτωση που η εγγραφή τους στον πίνακα $MFT έχει χρησιμοποιηθεί ξανά από το Σύστημα, η χάραξή τους από τον μη κατανεμημένο χώρο είναι εξίσου δύσκολη καθώς οι περισσότερες βασικές τεχνικές αυτής αποτυγχάνουν στην ανάκτηση κατακερματισμένων αρχείων.

Στο πρακτικό παράδειγμά που ακολουθεί θα ανακτηθεί ένα διαγεγραμμένο (ολική διαγραφή με χρήση Shift+Delete) αρχείο κειμένου διαμόρφωσης pdf του οποίου όμως διασώζεται ακόμα η εγγραφή στον πίνακα $MFT. Θα παρουσιαστεί επίσης η έννοια του ορφανού αρχείου καθώς ο γονικός κατάλογος του αρχείου έχει επίσης διαγραφεί και η εγγραφή αρχείου του στο $MFT έχει αντικατασταθεί από άλλον κατάλογο.

Υπενθυμίζεται ότι τα προκαταρκτικά στάδια της εξέτασης για την απόκτηση του εγκληματολογικού αντιγράφου και την ψηφιακή επιβεβαίωση αυτού, είναι τα ίδια που έλαβαν χώρα στο μέρος 3.1 του παρόντος κεφαλαίου και ως αυτού η παρουσίασή τους εκ νέου παραλείπεται.

Επικεφαλίδα Εγγραφής Αρχείου

Η επικεφαλίδα της εγγραφής αρχείου δίνεται στην παρακάτω εικόνα:



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0376E400	46	49	4C	45	30	00	03	00	98	9B	79	02	00	00	00	00
0376E410	02	00	01	00	38	00	00	00	88	01	00	00	00	04	00	00
0376E420	00	00	00	00	00	00	00	00	04	00	00	00	65	00	00	00
0376E430	06	00	00	00	00	00	00	00	10	00	00	00	60	00	00	00

Επικεφαλίδα Εγγραφής Αρχείου για το αρχείο manual.pdf

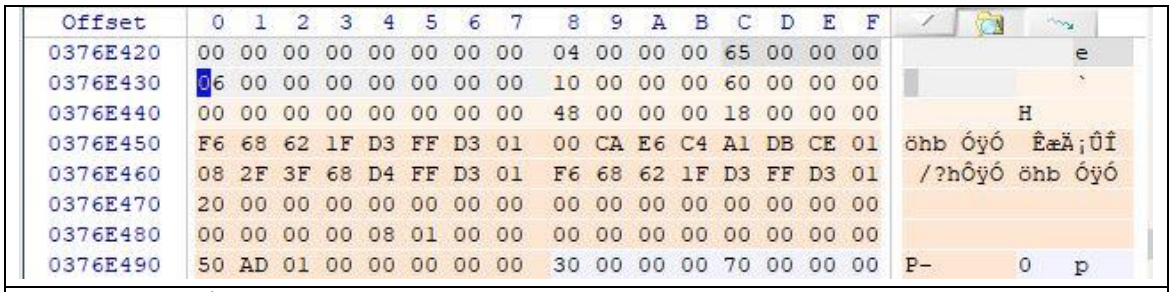
Διακρίνονται οι πιο σημαντικές τιμές:

- Υπογραφή – 0x46 49 4C 45 = FILE
- Θέση διάταξης Fix-up – 0x30 (Little Endian) = 48 bytes από την αρχή του χαρακτηριστικού
- Αριθμός καταχωρίσεων στη διάταξη Fix-up – 0x03 = τρεις διαφορετικές τιμές μεγέθους δύο bytes η καθεμία στη διάταξη Fix-up
- Άθροισμα διαδοχής – 0x02 = Η εγγραφή αρχείου έχει χρησιμοποιηθεί δύο φορές συνολικά
- Θέση πρώτου χαρακτηριστικού – 0x38 = 56 bytes από την αρχή της εγγραφής (ουσιαστικά το μήκος της επικεφαλίδας εγγραφής)
- Σημαία (Flag) κατάστασης χρήσης – 0x00 = διαγραμμένο αρχείο
- Λογικό μέγεθος της εγγραφής αρχείου (Logical) – 0x01 88 = 392 bytes χρησιμοποιούνται συνολικά από την εγγραφή
- Αριθμός εγγραφής αρχείου στο $MFT – 0x65 = καταχώριση υπ' αριθμόν 101 στον πίνακα $MFT

Η σημαία κατάστασης χρήσης στη θέση 0x0376E416 φέρει την τιμή 0x00 00 και υποδηλώνει ότι η εγγραφή ανήκει σε ένα διαγεγραμμένο αρχείο. Ακολουθώντας την πληροφόρηση της επικεφαλίδας θα πλοηγηθούμε στο πρώτο κατά σειρά χαρακτηριστικό για να συνεχίσουμε την ανάλυση της εγγραφής.

Χαρακτηριστικό $Standard_Information

Το χαρακτηριστικό $Standard_Information αποτελεί το πρώτο κατά σειρά χαρακτηριστικό της εγγραφής. Ακολουθεί την επικεφαλίδα τηρώντας το όριο θέσης του πολλαπλάσιου των 8 byte. Αρχίζει στη θέση 0x0376E438 και η δομή του απεικονίζεται παρακάτω:



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0376E420	00	00	00	00	00	00	00	00	04	00	00	00	65	00	00	00
0376E430	06	00	00	00	00	00	00	00	10	00	00	00	60	00	00	00
0376E440	00	00	00	00	00	00	00	00	48	00	00	00	18	00	00	00
0376E450	F6	68	62	1F	D3	FF	D3	01	00	CA	E6	C4	A1	DB	CE	01
0376E460	08	2F	3F	68	D4	FF	D3	01	F6	68	62	1F	D3	FF	D3	01
0376E470	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0376E480	00	00	00	00	08	01	00	00	00	00	00	00	00	00	00	00
0376E490	50	AD	01	00	00	00	00	00	30	00	00	00	70	00	00	00

χαρακτηριστικό $Standard_Information για το αρχείο manual.pdf

Διακρίνονται οι πιο σημαντικές τιμές:

- Κωδικός χαρακτηριστικού – 0x10 00 00 00 = $Standard_Information
- Μήκος του χαρακτηριστικού – 0x60 (Little Endian) = 96 bytes συνολικά (συμπεριλαμβανομένου της επικεφαλίδας χαρακτηριστικού, της επικεφαλίδας τοπικού περιεχομένου και του περιεχομένου)

- Σημαία μη τοπικού περιεχομένου – 0x00 = τοπικό περιεχόμενο
- Σημαίες αρχείου (Flags) – 0x00 = Το αρχείο δεν αποτελεί συμπιεσμένο, κρυπτογραφημένο ή αραιό αρχείο.
- Θέση του περιεχομένου – 0x18 = 24 bytes από την αρχή του χαρακτηριστικού
- Χρονοσφραγίδα δημιουργίας – 0x01 D3 FF D3 1F 62 68 F6 = FILETIME σε UTC
- Χρονοσφραγίδα τροποποίησης – 0x01 CE DB A1 C4 E6 CA 00 = FILETIME σε UTC
- Χρονοσφραγίδα τροποποίησης εγγραφής $MFT – 0x01 D3 FF D4 68 3F 2F 08= FILETIME σε UTC
- Σημαίες είδους αρχείου – 0x20 = αρχείο έτοιμο προς αρχειοθέτηση (archive)

Το περιεχόμενο του χαρακτηριστικού είναι πάντοτε τοπικό. Η εξέτασή του δίνει πληροφόρηση για το χρονολόγιο διάδρασης μεταξύ του χρήστη και του αρχείου.

Η χρονοσφραγίδα δημιουργίας φέρει την τιμή 0x01 D3 FF D3 1F 62 68 F6 με μορφή FILETIME σε UTC. Η μετάφραση της δίνει την τιμή 09-06-2018, 09:20:35 που είναι η τιμή που το αρχείο εικόνας δημιουργήθηκε στον παρόν τόμο. Η χρονοσφραγίδα τροποποίησης φέρει την τιμή 0x01 CE DB A1 C4 E6 CA 00 με μορφή FILETIME σε UTC. Η μετάφρασή της δίνει την τιμή 07-11-2013, 10:11:48. Επειδή ο χρόνος τροποποίησης του αρχείου είναι προγενέστερος του χρόνου δημιουργίας του, συμπεραίνουμε ότι το αρχείο είχε αρχικά δημιουργηθεί σε διαφορετικό τόμο ή μέσο και κατόπιν μετακινήθηκε στον παρόντα.

Η χρονοσφραγίδα τροποποίησης εγγραφής $MFT φέρει την τιμή 0x01 D3 FF D4 68 3F 2F 08 με μορφή FILETIME σε UTC. Η μετάφραση της δίνει την τιμή 09-06-2018, 09:27:49 που σηματοδοτεί την ημεροχρονολογία και ώρα που το αρχείο διεγράφη καθώς η τροποποίηση της σημαίας κατάστασης χρήσης στην επικεφαλίδα της εγγραφής προκειμένου να εμφανίζεται το αρχείο ως διαγραμμένο, είναι και η τελευταία χρονικά αλλαγή που επήλθε στην εγγραφή $MFT του αρχείου.

Χαρακτηριστικό $File_Name

Το χαρακτηριστικό $File_Name αποτελεί το δεύτερο κατά σειρά χαρακτηριστικό της εγγραφής. Ακολουθεί την επικεφαλίδα τηρώντας το όριο θέσης του πολλαπλάσιου των 8 byte. Αρχίζει στη θέση 0x0376E498 και η δομή του απεικονίζεται παρακάτω:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0376E480	00	00	00	00	08	01	00	00	00	00	00	00	00	00	00	00
0376E490	50	AD	01	00	00	00	00	00	30	00	00	00	70	00	00	00
0376E4A0	00	00	00	00	00	00	03	00	56	00	00	00	18	00	01	00
0376E4B0	3F	00	00	00	00	00	01	00	F6	68	62	1F	D3	FF	D3	01
0376E4C0	00	CA	E6	C4	A1	DB	CE	01	66	75	3C	7E	D7	F1	D3	01
0376E4D0	F6	68	62	1F	D3	FF	D3	01	00	60	3E	02	00	00	00	00
0376E4E0	D8	5F	3E	02	00	00	00	00	20	00	00	00	00	00	00	00
0376E4F0	0A	00	4D	00	61	00	6E	00	75	00	61	00	6C	00	2E	00
0376E500	70	00	64	00	66	00	00	00	80	00	00	00	78	00	00	00

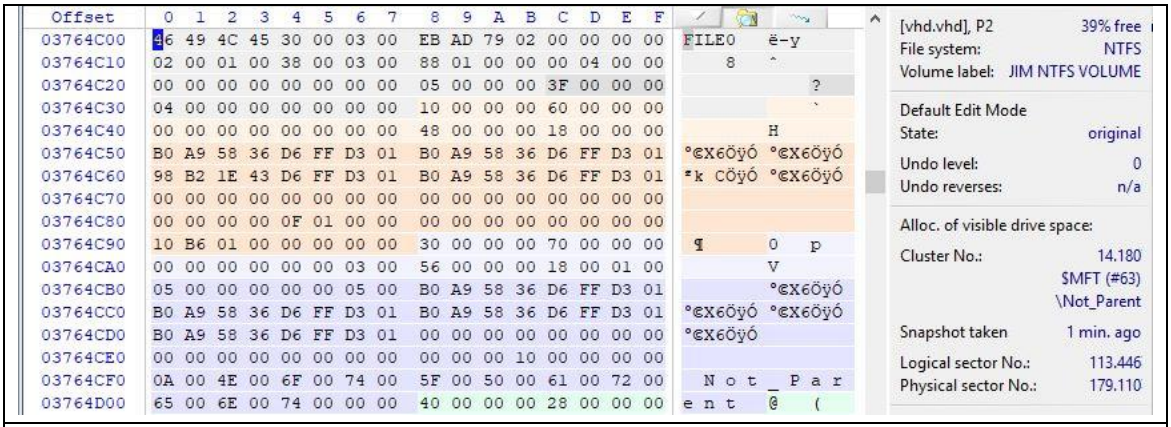
χαρακτηριστικό $File_Name για το αρχείο manual.pdf

Διακρίνονται οι πιο σημαντικές τιμές:

- Κωδικός χαρακτηριστικού – 0x30 00 00 00 = $File_Name
- Μήκος του χαρακτηριστικού – 0x70 (Little Endian) = 112 bytes συνολικά (συμπεριλαμβανομένου της επικεφαλίδας χαρακτηριστικού, της επικεφαλίδας τοπικού περιεχομένου και του περιεχομένου)

- Σημαία μη τοπικού περιεχομένου – 0x00 = τοπικό περιεχόμενο
- Σημαίες αρχείου (Flags) – 0x00 = Το αρχείο δεν αποτελεί συμπιεσμένο, κρυπτογραφημένο ή αραιό αρχείο
- Θέση του περιεχομένου – 0x18 = 24 bytes από την αρχή του χαρακτηριστικού
- Αριθμός εγγραφής αρχείου $MFT του γονικού καταλόγου – 0x3F = #63 καταχώριση στον πίνακα $MFT για τον γονικό κατάλογο του αρχείου
- Αριθμός διαδοχής του γονικού καταλόγου – 0x01= Η εγγραφή του γονικού καταλόγου στο $MFT έχει χρησιμοποιηθεί μία συνολικά φορές
- Χρονοσφραγίδα τροποποίησης εγγραφής $MFT – 0x01 D3 F1 D7 7E 3C 75 66= FILETIME σε UTC
- Μέγεθος ονόματος αρχείου – 0x0A = Δέκα χαρακτήρες Unicode
- Τύπος ονόματος αρχείου – 0x00 = ονομασία τύπου POSIX
- Ονομασία αρχείου – 0x4D 00 61 00 6E 00 75 00 61 00 6C 00 2E 00 70 00 64 00 66 00 = “Manual.pdf”

Ο αριθμός εγγραφής $MFT του γονικού καταλόγου του αρχείου pdf ισούται με 63 και το άθροισμα διαδοχής του με 1. Μεταβαίνοντας στην εν λόγω εγγραφή του γονικού καταλόγου και αναλύοντας την επικεφαλίδα της διαπιστώνουμε ότι το άθροισμα διαδοχής εντοπίζεται στη θέση 0x03764C10 και ισούται με τον αριθμό 2. Μέχρι αυτό το σημείο η σχέση μεταξύ γονέα-παιδιού ακόμα ερευνάται καθώς το άθροισμα διαδοχής αυξάνεται κατά μία μοναδιαία τιμή σε κάθε διαγραφή της εγγραφής αρχείου οπότε αυτός ο αριθμός διαδοχής θα μπορούσε απλά να δείχνει ότι και ο γονικός κατάλογος του αρχείου έχει διαγραφεί.



χαρακτηριστικό $File_Name για το αρχείο manual.pdf

Η σημαία κατάστασης χρήσης όμως για την παρούσα εγγραφή εντοπίζεται στην θέση 0x03764C16 και φέρει την τιμή 0x03 που μεταφράζεται ως κατανεμημένος κατάλογος. Ένεκα αυτών συμπεραίνεται ότι η σχέση γονέα-παιδιού δεν είναι ενεργή καθώς προκύπτει ότι ο νέος κατάλογος που καταλαμβάνει την εγγραφή υπ’ αριθμόν 63 του πίνακα $MFT με όνομα “Not_Parent”, έχει εγγραφεί επί του αρχικού γονικού καταλόγου του αρχείου.

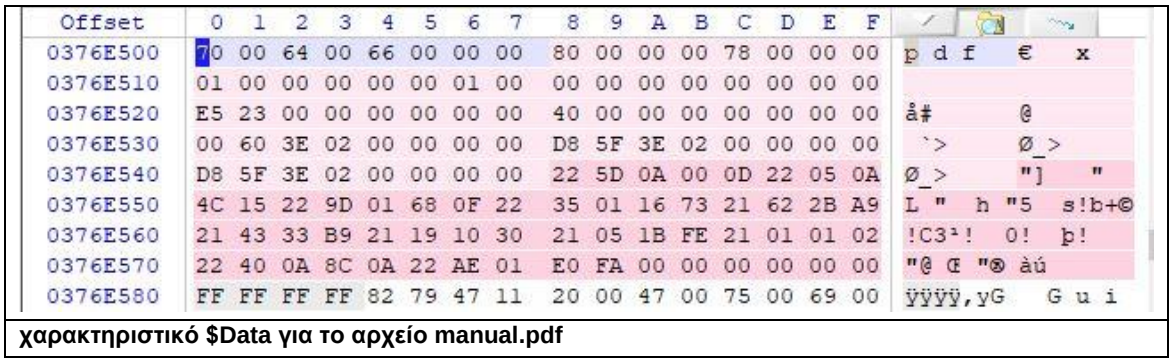
Αντιθέτως το αρχείο κειμένου pdf φέρει την εγγραφή αρχείου του στο $MFT αναλλοίωτη και συνεπώς, εάν κανένα από τα fragments του περιεχομένου του στον τόμο δεν έχει επαναγραφεί, μπορεί ακόμα να ανακτηθεί πλήρως. Δεν θα μπορούσε ποτέ όμως να τοποθετηθεί στη σωστή θέση της ιεραρχικής δομής του Συστήματος Αρχείων, δεδομένου ότι αποτελεί ένα ορφανό αρχείο και δεν υπάρχει καμία περίπτωση πλέον να συσχετιστεί με κάποιο κατάλογο-γονέα.

Το μέγεθος του ονόματος του αρχείου είναι δέκα χαρακτήρες Unicode και άρα αποθηκεύεται στο μέσο χρησιμοποιώντας συνολικά 20 bytes, δύο για την απόδοση του κάθε χαρακτήρα. Ο

χώρος ονομασίας (namespace) είναι POSIX λόγω της μίξης κεφαλαίων και μικρών γραμμάτων στο όνομα του αρχείου, το οποίο ξεκινάει στη θέση 0x0376E4F2 με τιμή “Manual.pdf”.

Χαρακτηριστικό $Data

Το χαρακτηριστικό $Data αποτελεί το τρίτο κατά σειρά χαρακτηριστικό της εγγραφής. Αρχίζει στη θέση 0x0376E508 και η δομή του απεικονίζεται παρακάτω:



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0376E500	70	00	64	00	66	00	00	00	80	00	00	00	78	00	00	00
0376E510	01	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00
0376E520	E5	23	00	00	00	00	00	00	40	00	00	00	00	00	00	00
0376E530	00	60	3E	02	00	00	00	00	D8	5F	3E	02	00	00	00	00
0376E540	D8	5F	3E	02	00	00	00	00	22	5D	0A	00	0D	22	05	0A
0376E550	4C	15	22	9D	01	68	0F	22	35	01	16	73	21	62	2B	A9
0376E560	21	43	33	B9	21	19	10	30	21	05	1B	FE	21	01	01	02
0376E570	22	40	0A	8C	0A	22	AE	01	E0	FA	00	00	00	00	00	00
0376E580	FF	FF	FF	FF	82	79	47	11	20	00	47	00	75	00	69	00

χαρακτηριστικό $Data για το αρχείο manual.pdf

Διακρίνονται οι πιο σημαντικές τιμές:

- Κωδικός χαρακτηριστικού – 0x80 00 00 00 = $File_Name
- Μήκος του χαρακτηριστικού – 0x78 (Little Endian) = 120 bytes συνολικά
- Σημαία μη τοπικού περιεχομένου – 0x01 = μη τοπικό περιεχόμενο
- Αριθμός έναρξης εικονικού cluster (VCN) της λίστας data run – 0x00 = Το πρώτο εικονικό cluster φέρει τον αριθμό 0
- Αριθμός λήξης εικονικού cluster (VCN) της λίστας data run – 0x23 E5 = Το τελευταίο εικονικό cluster φέρει τον αριθμό 9.189
- Θέση της λίστας data runs – 0x40 = 64 bytes από την αρχή του χαρακτηριστικού
- Κατανεμημένο μέγεθος του αρχείου – 0x02 3E 60 00 = 37.642.240 bytes
- Πραγματικό μέγεθος του αρχείου – 0x02 3E 5F D8 = 37.642.200 bytes
- Αρχικοποιημένο μέγεθος του αρχείου – 0x02 3E 5F D8 = 37.642.200 bytes
- Λίστα data runs – 0x22 5D 0A 00 0D 22 05 0A 4C 15 22 9D 01 68 0F 22 35 01 16 73 21 62 2B A9 21 43 33 B9 21 19 10 30 21 05 1B FE 21 01 01 02 22 40 0A 8C 0A 22 AE 01 E0 FA 00

Το περιεχόμενο του χαρακτηριστικού είναι μη τοπικό οπότε δεδομένου ότι το αρχείο είναι διαγεγραμμένο, αυτό θα πρέπει να ανακτηθεί από τον μη κατανεμημένο χώρο του τόμου με τη βοήθεια της λίστας data runs.

Επίσης επειδή το περιεχόμενο του αρχείου είναι μη τοπικό, έχουν αποδοθεί από το Σύστημα εικονικοί αριθμοί στα clusters που το απαρτίζουν, ώστε να επιτυγχάνεται η σωστή επανασύνδεσή τους κάθε φορά που το αρχείο πρέπει να προσπελαστεί. Στη συγκεκριμένη περίπτωση το αρχείο είναι κατακερματισμένο, ήτοι καταλαμβάνει πλήθος κομματιών διαδοχικών clusters στο μέσο για την αποθήκευσή του.

Ο αριθμός έναρξης και λήξης των εικονικών clusters μας δίνουν τα συνολικά cluster που καταλαμβάνει το αρχείο στον τόμο. Ο αριθμός έναρξης είναι μηδέν και ο αριθμός λήξης είναι 9.189, που μεταφράζεται ως 9.190 συνολικά κατανεμημένα clusters στο αρχείο αφού το υπ’ αριθμόν 0 cluster αποτελεί μία επιπλέον μονάδα η οποία προστίθεται στα υπόλοιπα 9.189 για να δώσει το σύνολο.

Η επαλήθευση αυτού του μεγέθους γίνεται από τη σύγκριση του κατανεμημένου μεγέθους του αρχείου στο δίσκο (physical size). Αφού το αρχείο καταλαμβάνει 9.190 συνολικά clusters και το μέγεθος του κάθε cluster είναι 4096 bytes (όπως αυτό προκύπτει από την ανάλυση του τομέα εκκίνησης), τότε 9.190 x 4096 = 37.642.240 bytes, αριθμός που συμφωνεί με το κατανεμημένο μέγεθος του αρχείου.

Το πραγματικό μέγεθος (logical size) του αρχείου στον τόμο είναι μικρότερο του κατανεμημένου, υποδεικνύοντας την ύπαρξη χώρου υπολείμματος αρχείου στο τέλος αυτού. Τέλος το πραγματικό μέγεθος και το αρχικοποιημένο μέγεθος ισούνται γεγονός που αποδεικνύει ότι το αρχείο βρίσκεται αποθηκευμένο στην ολότητά του στον τόμο.

Λίστα Data Runs

Το τελευταίο στάδιο για να ανακτήσουμε το αρχείο είναι ο εντοπισμός όλων των διαφορετικών κομματιών του στον μη κατανεμημένο χώρο του τόμου και η επανασύνδεσή τους. Η λίστα Data Runs του χαρακτηριστικού $Data παρέχει όλη την απαραίτητη πληροφορία για την επίτευξη αυτού του σκοπού. Η run list του υπό εξέταση αρχείου είναι η 22 5D 0A 00 0D 22 05 0A 4C 15 22 9D 01 68 0F 22 35 01 16 73 21 62 2B A9 21 43 33 B9 21 19 10 30 21 05 1B FE 21 01 01 02 22 40 0A 8C 0A 22 AE 01 E0 FA 00. Ακολουθεί η ανάλυσή της:

Καταχώριση 1η

22 5D 0A 00 0D. Η επικεφαλίδα 0x22 δίνει το μήκος της καταχώρισης: 2+2= 4 bytes.

- Το περιεχόμενο της καταχώρισης είναι 5D 0A 00 0D

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 2

- Άρα το μέγεθος του fragment είναι 0x5D 0A (Little Endian) = 2.653 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη θέση 0x00 0D (Little Endian) = 3.328 που ταυτίζεται με το λογικό cluster του τόμου (LCN) αφού εξετάζουμε το πρώτο fragment του αρχείου

Το τελευταίο cluster που καταλαμβάνει αυτό το fragment καθορίζεται αν στη θέση έναρξης του προστεθεί ο αριθμός των clusters που το απαρτίζουν και αφαιρεθεί μία μονάδα. Αφαιρούμε μία μονάδα αφού το αρχικό cluster προσμετρείται και αυτό στο ολικό μέγεθος. Ήτοι, 3.328 + 2.653 – 1 = 5.980 cluster είναι το τελικό cluster που καταλαμβάνει η τρέχουσα καταχώριση της Run List. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
2.653	3.328	3.328	5.980
Καταχώριση 1 στη λίστα Data Runs			

Καταχώριση 2η

22 05 0A 4C 15. Η επικεφαλίδα 0x22 δίνει το μήκος της καταχώρισης: 2+2= 4 bytes.

- Το περιεχόμενο της καταχώρισης είναι 05 0A 4C 15

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 2

- Άρα το μέγεθος του fragment είναι 0x05 0A (Little Endian) = 2.565 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x4C 15 (Little Endian) = 5.452

Επειδή πρόκειται για τη δεύτερη κατά σειρά καταχώριση στη Run List η θέση έναρξης δεν ταυτίζεται με το λογικό cluster του τόμου όπως στην πρώτη. Από αυτή την καταχώριση και εφεξής, κάθε θέση έναρξης πρέπει να προστίθεται στο λογικό cluster έναρξης (LCN) της προηγούμενης καταχώρισης, ώστε τελικά να εντοπίζεται η πραγματική του τοποθεσία συναρτήσει των λογικών clusters του τόμου. Επομένως για την τρέχουσα καταχώριση το αρχικό LCN δίνεται από τη σχέση «αρχικό LCN προηγούμενης καταχώρισης + Θέση έναρξης τρέχουσας καταχώρισης = αρχικό LCN τρέχουσας καταχώριση». Δηλαδή 3.328 + 5.452 =8.780 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 8.780 + 2.565 -1 = 11.344 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
2.565	5.452	8.780	11.344
Καταχώριση 2 στη λίστα Data Runs			

Καταχώριση 3η

22 9D 01 68 0F. Η επικεφαλίδα 0x22 δίνει το μήκος της καταχώρισης: 2+2= 4 bytes.

- Το περιεχόμενο της καταχώρισης είναι 9D 01 68 0F

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 2

- Άρα το μέγεθος του fragment είναι 0x9D 01 (Little Endian) = 413 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x4C 15 (Little Endian) = 3.944

Επομένως το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 8.780 + 3.944 = 12.724 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 12.724 + 413 -1 = 13.136 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
413	3.944	12.724	13.136
Καταχώριση 3 στη λίστα Data Runs			

Καταχώριση 4η

22 35 01 16 73. Η επικεφαλίδα 0x22 δίνει το μήκος της καταχώρισης: 2+2= 4 bytes.

- Το περιεχόμενο της καταχώρισης είναι 35 01 16 73

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 2

- Άρα το μέγεθος του fragment είναι 0x35 01 (Little Endian) = 309 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x16 73 (Little Endian) = 29.462

Επομένως το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 12.724 + 29.462 = 42.186 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 42.186 + 309 – 1 = 13.136 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
309	29.462	42.186	42.494
Καταχώριση 4 στη λίστα Data Runs			

Καταχώριση 5η

21 62 2B A9. Η επικεφαλίδα 0x21 δίνει το μήκος της καταχώρισης: 2+1= 3 bytes.

- Το περιεχόμενο της καταχώρισης είναι 62 2B A9

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 1

- Άρα το μέγεθος του fragment είναι 0x62 (Little Endian) = 98 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x2B A9 (Little Endian) = –22.229

Στην τρέχουσα καταχώριση ο αριθμός σχετικής θέσης προκύπτει αρνητικός αριθμός καθώς οι Run Lists υλοποιούνται με signed integers που δύναται να πάρουν και αρνητικές τιμές. Με αυτόν τον τρόπο το Σύστημα μπορεί να δημιουργεί δείκτες (pointers) που δείχνουν και σε μικρότερους αριθμούς λογικών clusters ώστε να επιτυγχάνεται η πλήρης χαρτογράφηση του τόμου.

Άρα το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 42.186 + (-22.229) = 19.957 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 19.957 + 98 – 1 = 20.054 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
98	–22.229	19.957	20.054
Καταχώριση 5 στη λίστα Data Runs			

Καταχώριση 6η

21 43 33 B9. Η επικεφαλίδα 0x21 δίνει το μήκος της καταχώρισης: 2+1= 3 bytes.

- Το περιεχόμενο της καταχώρισης είναι 43 33 B9

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 1

- Άρα το μέγεθος του fragment είναι 0x43 (Little Endian) = 67 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x33 B9 (Little Endian) = –18.125

Άρα το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 19.957 + (–18.125) = 1.832 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 1.832 + 67 – 1 = 1.898 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
67	–18.125	1.832	1.898
Καταχώριση 6 στη λίστα Data Runs			

Καταχώριση 7η

21 19 10 30. Η επικεφαλίδα 0x21 δίνει το μήκος της καταχώρισης: 2+1= 3 bytes.

- Το περιεχόμενο της καταχώρισης είναι 19 10 30

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 1

- Άρα το μέγεθος του fragment είναι 0x19 (Little Endian) = 25 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x10 30(Little Endian) = 12.304

Άρα το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 1.832 + (12.304) = 14.136 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 14.136 + 25 – 1 = 14.160 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
25	12.304	14.136	14.160
Καταχώριση 7 στη λίστα Data Runs			

Καταχώριση 8η

21 05 1B FE. Η επικεφαλίδα 0x21 δίνει το μήκος της καταχώρισης: 2+1= 3 bytes.

- Το περιεχόμενο της καταχώρισης είναι 05 1B FE

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 1

- Άρα το μέγεθος του fragment είναι 0x05 (Little Endian) = 5 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x1B FE (Little Endian) = – 485

Άρα το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 14.136 + (–485) = 13.651 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 13.651 + 5 – 1 = 13.655 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
5	– 485	13.651	13.655
Καταχώριση 8 στη λίστα Data Runs			

Καταχώριση 9η

21 01 01 02. Η επικεφαλίδα 0x21 δίνει το μήκος της καταχώρισης: 2+1= 3 bytes.

- Το περιεχόμενο της καταχώρισης είναι 01 01 02

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 1

- Άρα το μέγεθος του fragment είναι 0x01 (Little Endian) = 1 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x01 02 (Little Endian) = 513

Άρα το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 13.651 + 513 = 14.164 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 14.164 + 1 – 1 = 14.164 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
1	513	14.164	14.164
Καταχώριση 9 στη λίστα Data Runs			

Καταχώριση 10η

22 40 0A 8C 0A. Η επικεφαλίδα 0x22 δίνει το μήκος της καταχώρισης: 2+2= 4 bytes.

- Το περιεχόμενο της καταχώρισης είναι 40 0A 8C 0A

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 2

- Άρα το μέγεθος του fragment είναι 0x40 0A (Little Endian) = 2.624 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0x8C 0A(Little Endian) = 2.700

Άρα το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 14.164 + 2.700 = 16.864 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 16.864 + 2.624 – 1 = 19.487 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
2.624	2700	16.864	19.487
Καταχώριση 10 στη λίστα Data Runs			

Καταχώριση 11η

22 AE 01 E0 FA. Η επικεφαλίδα 0x22 δίνει το μήκος της καταχώρισης: 2+2= 4 bytes.

- Το περιεχόμενο της καταχώρισης είναι AE 01 E0 FA

Το δεξί nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για το μέγεθος του fragment = 2

- Άρα το μέγεθος του fragment είναι 0xAE 01(Little Endian) = 430 clusters

Το αριστερό nibble της επικεφαλίδας δείχνει πόσα bytes θα χρησιμοποιηθούν για τη θέση έναρξης του fragment = 2

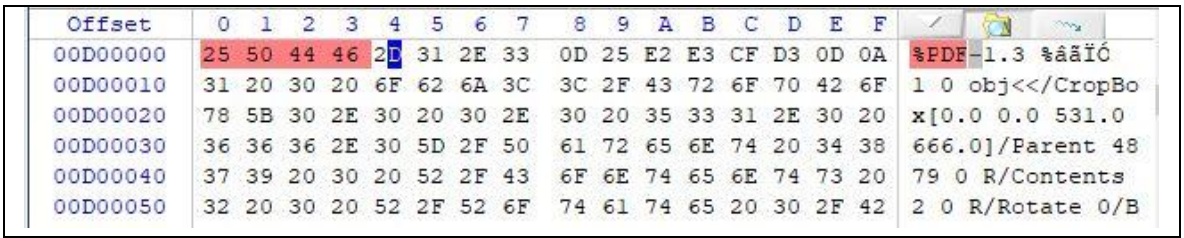
- Άρα το υπό εξέταση fragment αρχίζει από τη σχετική θέση 0xE0 FA (Little Endian) = –1.312

Άρα το αρχικό LCN της τρέχουσας καταχώρισης είναι το υπ' αριθμόν 16.864 + (–1.312) = 15.552 λογικό cluster του τόμου. Το τελευταίο cluster που καταλαμβάνει αυτό το fragment είναι το υπ' αριθμόν 15.552 + 430 – 1 = 15.981 λογικό cluster του τόμου. Συνοψίζοντας για την καταχώριση:

Μήκος σε Clusters	Σχετική θέση έναρξης	Αρχικό LCN	Τελικό LCN
430	–1.312	15.552	15.981
Καταχώριση 11 στη λίστα Data Runs			

Το επόμενο byte στην λίστα είναι το 0x00, δηλαδή η μηδενική επικεφαλίδα η οποία υποδηλώνει το τέλος της λίστας data runs.

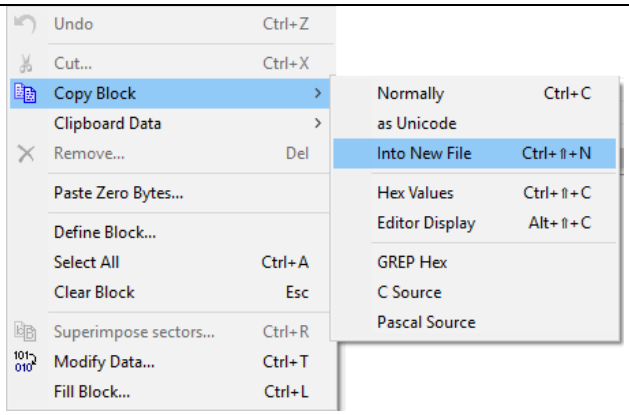
Συνοψίζοντας όλα τα ανωτέρω, για να ανακτηθεί το αρχείο θα πρέπει να περιηγηθούμε, διαδοχικά με τη σειρά που παρείχε η Run List, σε όλα τα κομμάτια αυτού (fragments) στον τόμο. Το πρώτο κομμάτι σύμφωνα με τη λίστα ξεκινά στο λογικό cluster 3.328 του τόμου. Χρησιμοποιώντας το εγκληματολογικό εργαλείο που έχουμε στη διάθεσή μας μεταβαίνουμε στο συγκεκριμένο λογικό cluster όπου και διαπιστώνουμε στην αρχή του πρώτου του τομέα την τιμή υπογραφής (file signature) των αρχείων κειμένου pdf 0x25 50 44 46.



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00D00000	25	50	44	46	2D	31	2E	33	0D	25	E2	E3	CF	D3	0D	0A	%PDF-1.3 %âãÏÓ
00D00010	31	20	30	20	6F	62	6A	3C	3C	2F	43	72	6F	70	42	6F	1 0 obj<</CropBo
00D00020	78	5B	30	2E	30	20	30	2E	30	20	35	33	31	2E	30	20	x[0.0 0.0 531.0
00D00030	36	36	36	2E	30	5D	2F	50	61	72	65	6E	74	20	34	38	666.0]/Parent 48
00D00040	37	39	20	30	20	52	2F	43	6F	6E	74	65	6E	74	73	20	79 0 R/Contents
00D00050	32	20	30	20	52	2F	52	6F	74	61	74	65	20	30	2F	42	2 0 R/Rotate 0/B

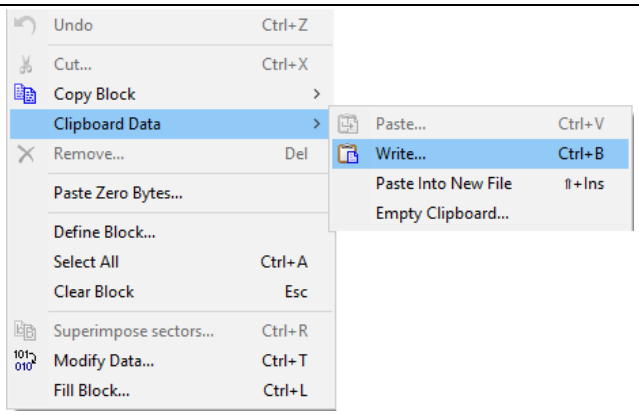
Αρχικό cluster του 1ου fragment

Δίνοντας στο εγκληματολογικό εργαλείο εντολή επιλογής αυτού και των επόμενων 2.652 clusters καταλήγουμε στο LCN 5.980 το οποίο αποτελεί και το τελικό cluster αυτού του κομματιού. Εν συνεχεία με την εντολή Copy Block -> Into New File επιτυγχάνεται η αντιγραφή αυτού του επιλεγμένου μπλοκ bytes, σε ένα νέο αρχείο στον υπολογιστή εξέτασης. Τη δεδομένη χρονική στιγμή αυτό το εξαγόμενο αρχείο είναι ημιτελές και δεν δύναται να προσπελαστεί από οικεία προγράμματα ανάγνωσης αρχείων pdf.



Αντιγραφή μπλοκ επιλεγμένων bytes σε νέο αρχείο

Στη συνέχεια μεταβαίνουμε στο λογικό cluster 8.780 που αποτελεί το δεύτερο κομμάτι του αρχείου σύμφωνα με την Run List. Δίνοντας στο εγκληματολογικό εργαλείο εντολή επιλογής του και των επόμενων 2.564 clusters καταλήγουμε στο LCN 11.344 το οποίο αποτελεί και το τελικό cluster αυτού του κομματιού. Εν συνεχεία με την εντολή Clipboard Data -> Write επιτυγχάνεται η προσάρτηση αυτού του επιλεγμένου μπλοκ bytes, στο τέλος του προηγούμενου κομματιού που έχει ήδη ανακτηθεί στο προηγούμενο βήμα.

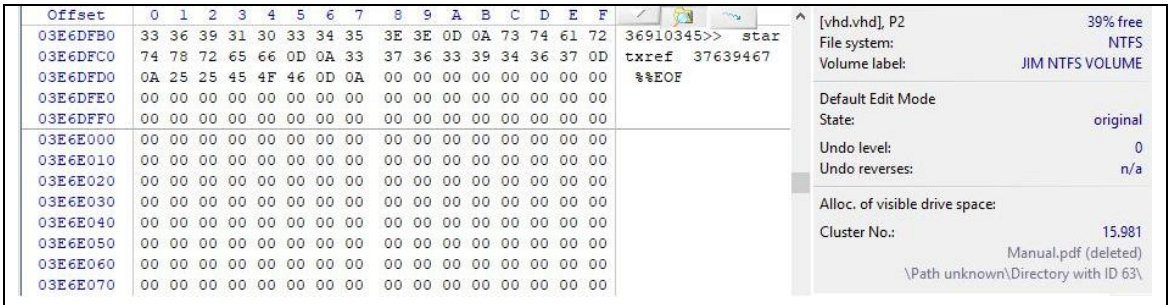


Προσάρτηση μπλοκ επιλεγμένων bytes σε ήδη υπάρχων αρχείο

Η διαδικασία ανάκτησης συνεχίζεται με τον ίδιο τρόπο για κάθε επόμενο κομμάτι της λίστας. Κάθε κομμάτι επιλέγεται στο σύνολο του και προσαρτάται στο τέλος του προϋπάρχοντος αρχείου, μεγαλώνοντας το μέγεθός του με κάθε εντολή εγγραφής write.

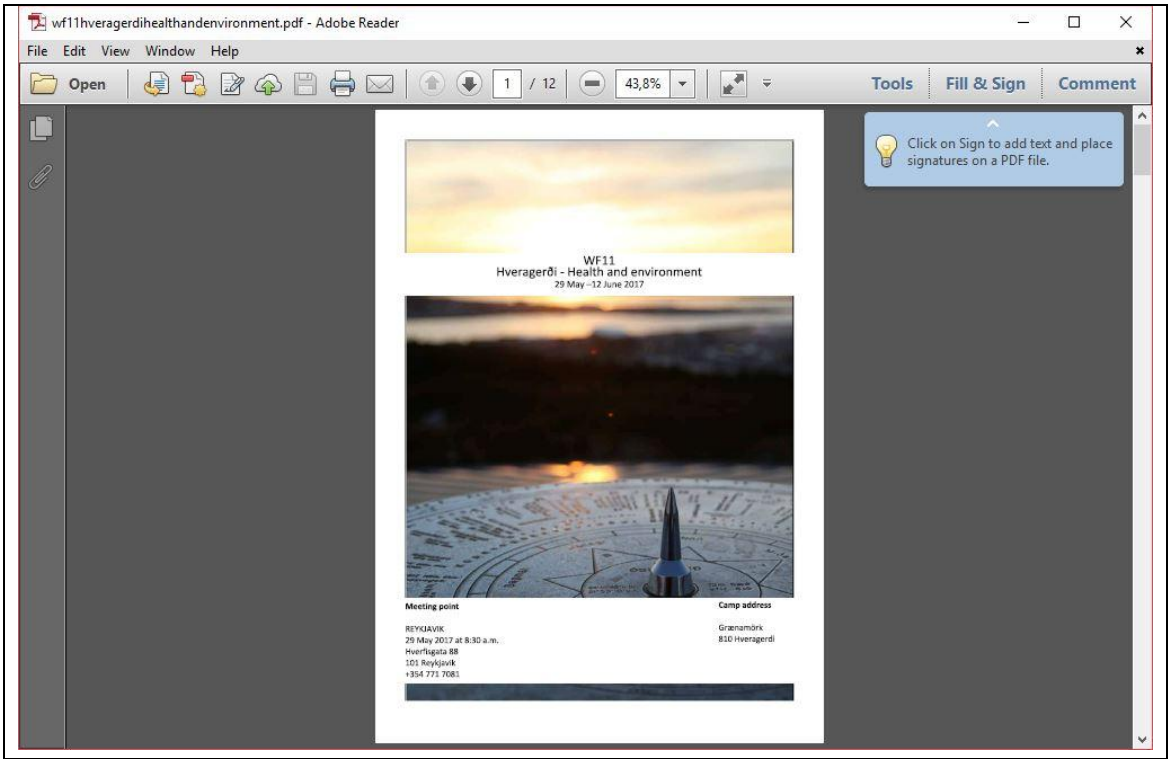
Φτάνοντας στο 11^{ο} κομμάτι του αρχείου επιλέγουμε το σύνολο των clusters που το αποτελούν και καταλήγουμε στο λογικό cluster 15.981 το οποίο είναι και το τελικό cluster του αρχείου. Περιηγούμαστε στον τελευταίο τομέα του cluster όπου και διαπιστώνουμε στη θέση 0x03E6DFCF την τιμή κατάληξης (footer) των αρχείων κειμένου pdf 0x0D 0A 25 25 45 4F 46 0D 0A.

Η διαφορά του πραγματικού από τον κατανεμημένο χώρο, 37.642.240 – 37.642.200 = 40 bytes υποδεικνύει ότι το αρχείο φέρει μόνο RAM Slack στον τελευταίο τομέα του τελευταίου cluster του που ισούται με 40 bytes. Αυτό επαληθεύεται και από την τιμή κατάληξης (footer) η οποία αφήνει υπόλοιπο ακριβώς 40 bytes μέχρι το τέλος του τομέα. Τα bytes αυτά περιέχουν την κενή τιμή 0x00 με την οποία έχει γεμίσει τον χώρο υπολείμματος του αρχείου το Σύστημα. Στον επόμενο τομέα ξεκινάει νέο cluster και παρατηρείται ότι δεν περιέχει δεδομένα.



Τιμή κατάληξης αρχείου pdf στον τελικό τομέα του αρχείου

Η ανεύρεση και πλήρης επανασύνδεση όλων των κομματιών του αρχείου οδηγεί στην εγκληματολογική του ανάκτηση καίτοι αυτό διαγράφηκε ολικά από τον χρήστη όπως επίσης και ο γονικός κατάλογος που το περιείχε. Το ανακτηθέν αρχείο είναι πλήρως λειτουργικό.



Τελικό ανακτηθέν αρχείο κειμένου διαμόρφωσης pdf

3.3 Data Carving Συνεχούς Αρχείου

Στην περίπτωση ολικής διαγραφής ενός αρχείου και επαναχρησιμοποίησης της εγγραφής αρχείου του στον πίνακα $MFT από το Σύστημα, καθίσταται αδύνατη η ανάκτησή του με τις ανωτέρω περιγραφόμενες διαδικασίες. Το αρχείο παραμένει άθικτο στον μη κατανεμημένο χώρο του τόμου για όσο χρονικό διάστημα τα clusters που καταλαμβάνει δεν κατανεμηθούν σε άλλο αρχείο, αλλά δύναται να ανακτηθεί μόνο με ειδικές διαδικασίες χάραξης δεδομένων (Data Carving).

Η ακριβή του τοποθεσία στην ιεραρχική δομή του Συστήματος δεν μπορεί να ανοικοδομηθεί, ενώ επίσης κανένα από τα μεταδεδομένα Συστήματος που το συνόδευαν δεν είναι πλέον ανακτήσιμα ώστε να προσφέρουν επιπλέον πληροφόρηση για αυτό. Το μόνο που διασώζεται είναι το ίδιο το περιεχόμενο του αρχείου, γεγονός που αποδυναμώνει την εγκληματολογική έρευνα γύρω από αυτό, καθώς κανένα περαιτέρω ασφαλές συμπέρασμα δεν μπορεί να συναχθεί, άλλο από την απλή προηγούμενη ύπαρξη του αρχείου στον τόμο σε κάποια παρελθοντική χρονική στιγμή.

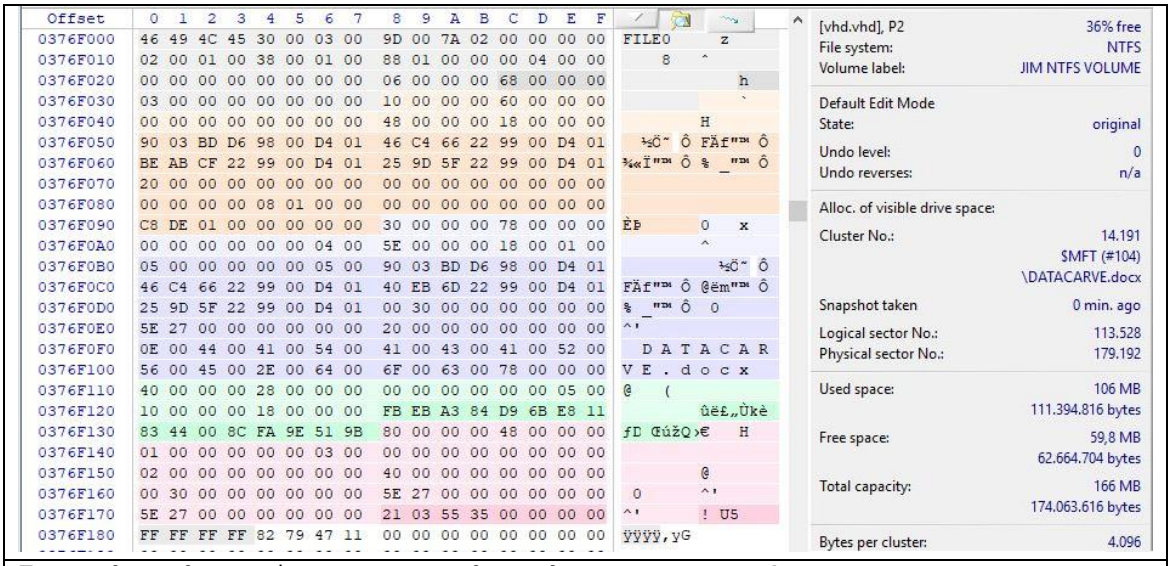
Από τις διάφορες τεχνικές χάραξης που υπάρχουν, η βασική μπορεί να πραγματοποιηθεί με χειροκίνητο τρόπο. Οι λοιπές τεχνικές, ως πιο εξειδικευμένες και περίπλοκες στην υλοποίησή τους, πραγματοποιούνται από ειδικά λογισμικά τα οποία χρησιμοποιούν μαθηματικούς αλγορίθμους για την επίτευξη του στόχου τους και απομακρύνονται του σκοπού της παρούσης.

Συνεπώς στο πρακτικό παράδειγμα που ακολουθεί θα παρουσιαστεί η χειροκίνητη ανάκτηση ενός αρχείου κειμένου διαμόρφωσης docx με τη χρήση της βασικής τεχνικής χάραξης

δεδομένων. Τα αρχεία αυτής της διαμόρφωσης αποτελούν σύνθετα αρχεία (Compound Files) που περιέχουν ένα συνδυασμό δεδομένων στο περιεχόμενο τους όπως κείμενο, αρχεία εικόνας, λογιστικά φύλλα κλπ. Τα αρχεία αυτής της μορφής φέρουν την χαρακτηριστική τιμή επικεφαλίδας (file header) “0x50 4B 03 04 14 00 06 00” και την χαρακτηριστική τιμή κατάληξης (file footer) “0x50 4B 05 06 + ακόμα 18 τυχαία bytes”, μεταξύ των οποίων περικλείεται το σύνολο του περιεχομένου του αρχείου.

Η βασική τεχνική χάραξης λειτουργεί αναζητώντας τις δύο αυτές τιμές, αποθηκευμένες διαδοχικά στον μη κατανεμημένο χώρο. Εφόσον αυτές εντοπιστούν θεωρείται ότι οριοθετούν το αναζητούμενο αρχείο το οποίο και κατόπιν εξάγεται (χαράσσεται) από το ευρύτερο μπλοκ δεδομένων που αποτελεί ο μη κατανεμημένος χώρος. Η αναζήτηση των τιμών αυτών ενδέχεται να επιφέρει ψευδώς θετικά αποτελέσματα (false positives) στην περίπτωση που η συγκεκριμένη αλληλουχία bytes εντοπίζεται σε άλλο μέρος αρχείου, εκτός της επικεφαλίδας ή της κατάληξης, αποθηκευμένη με την ίδια ακριβώς διαδοχή. Τα αρχεία που θα εξαχθούν σε αυτή την περίπτωση δεν θα είναι προσπελάσιμα. Ακολουθεί η περιγραφή της διαδικασίας χάραξης:

Δημιουργείται ένα αρχείο κειμένου διαμόρφωσης docx στον τόμο. Η εγγραφή αρχείου του καταλαμβάνει τη θέση #104 στον πίνακα $MFT. Όπως προκύπτει από την εγγραφή, το αρχείο ονομάζεται “DATACARVE.docx” έχει ημεροχρονολογία και ώρα δημιουργίας την 10-06-2018, 08:55:53, γονικό κατάλογο τον Root, αριθμό διαδοχής 2 και λογικό μέγεθος 10.078 bytes. Το περιεχόμενό του είναι μη τοπικό και η run list του χαρακτηριστικού $Data υποδεικνύει ότι τη δεδομένη στιγμή αυτό καταλαμβάνει τρία διαδοχικά clusters, τα 13.653 ως 13.655 LCN, ήτοι το αρχείο είναι συνεχές.



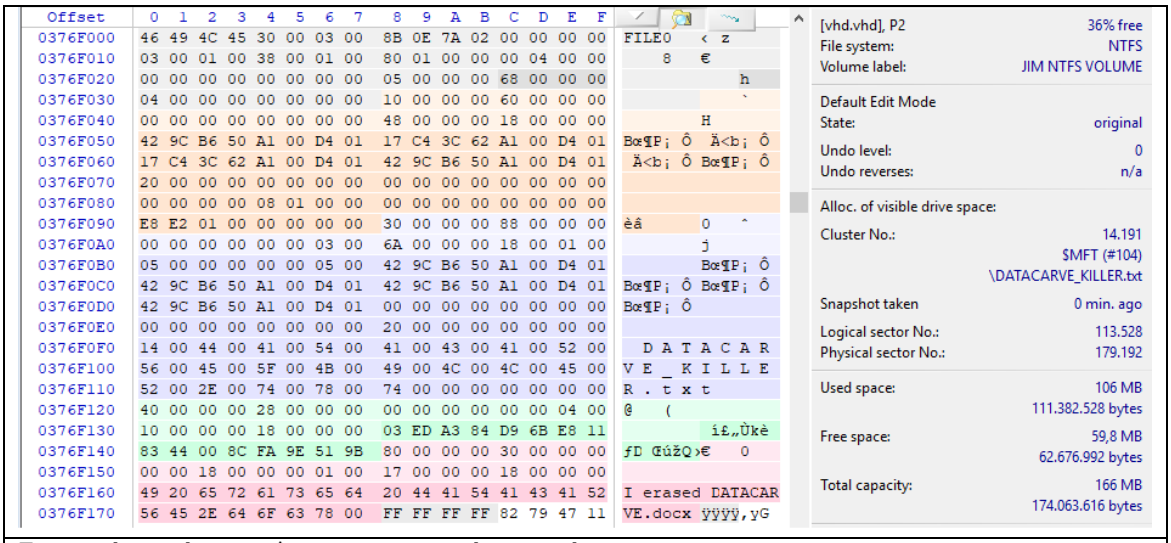
Εγγραφή αρχείου στο $MFT για το αρχείο κειμένου DATACARVE.docx

Στη συνέχεια το αρχείο διαγράφεται ολικά με τη χρήση των πλήκτρων Shift+Delete. Τα βήματα που περιγράφονται στο κεφάλαιο 2.3 λαμβάνουν χώρα. Η εγγραφή του αρχείου στο $MFT υπάρχει ακόμα, με μόνη διαφορά ότι το άθροισμα διαδοχής της έχει αυξηθεί κατά μία μονάδα και η σημαία κατάστασής χρήσης φέρει πλέον την τιμή 0x00 - διαγεγραμμένο αρχείο.

Ένα αρχείο σημειώματος txt θα δημιουργηθεί τώρα στον τόμο. Ο αλγόριθμος “top down - first available” που χρησιμοποιεί το Σύστημα Αρχείων NTFS για αποφυγή κατακερματισμού του πίνακα $MFT θα οδηγήσει στην ανακατάληψη της εγγραφής υπ’ αριθμόν #104 η οποία είναι η μόνη ανενεργή αυτή τη στιγμή στον πίνακα. Ως αυτού δε δημιουργείται νέα εγγραφή στο $MFT

και συνεπώς αυξάνεται η αποδοτικότητα του Συστήματος, χάνεται όμως κάθε πληροφορία που υπήρξε ποτέ στον τόμο για το αρχείο κειμένου DATACARVE.docx.

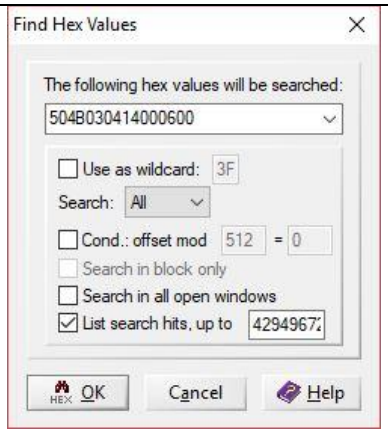
Όπως προκύπτει από την εγγραφή, το αρχείο ονομάζεται “DATACARVE_KILLER.txt” έχει ημεροχρονολογία και ώρα δημιουργίας την 10-06-2018, 09:56:34, γονικό κατάλογο τον Root, αριθμό διαδοχής 3 και τοπικό περιεχόμενο μεγέθους 23 bytes που αναγράφει «I erased DATACARVE.docx»



Εγγραφή αρχείου στο $MFT για το αρχείο σημειώματος DATACARVE_KILLER.txt

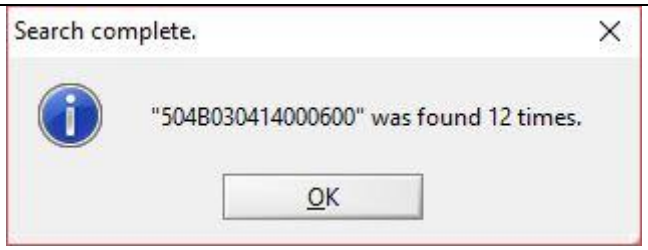
Κανένας σύνδεσμος δεν υπάρχει πλέον μεταξύ του συστήματος και του αρχείο DATACARVE.docx που καταλάμβανε την εγγραφή #104 πριν τη δημιουργία του αρχείου txt. Ο εξεταστής λοιπόν θα πρέπει να περιηγηθεί στον μη κατανεμημένο χώρο του τόμου και να αναζητήσει εκεί το αρχείο. Στο χρησιμοποιούμενο εγκληματολογικό λογισμικό του παρόντος πρακτικού, ο μη κατανεμημένος χώρος αναφέρεται ως Free Space και δεν φέρει καμία δομή οργάνωσης. Δεν υπάρχουν τομείς ή clusters αλλά το σύνολό του χώρου που καταλαμβάνει είναι χωρισμένο σε σελίδες μεγέθους 560 bytes ακατέργαστων δεδομένων (raw data).

Η λάξευση του αρχείου μέσα από αυτόν τον αδόμητο χώρο θα πραγματοποιηθεί με τη βοήθεια των τιμών επικεφαλίδας και κατάληξης που αυτό φέρει. Ενώ βρισκόμαστε στο Free space του τόμου θα ενεργήσουμε αναζήτηση της δεκαεξαδικής τιμής 0x50 4B 03 04 14 00 06 00 ζητώντας από το λογισμικό να δημιουργήσει μία λίστα των θέσεων που θα εντοπίσει την τιμή αυτή.



Αναζήτηση τιμής επικεφαλίδας για το αναζητούμενο αρχείο κειμένου

Το λογισμικό εκτελεί την αναζήτηση και παράγει μήνυμα εύρεσης της αναζητούμενης τιμής. Αυτή ανευρέθη δώδεκα φορές στον μη κατανεμημένο χώρο.



Αποτελέσματα αναζήτησης τιμής επικεφαλίδας

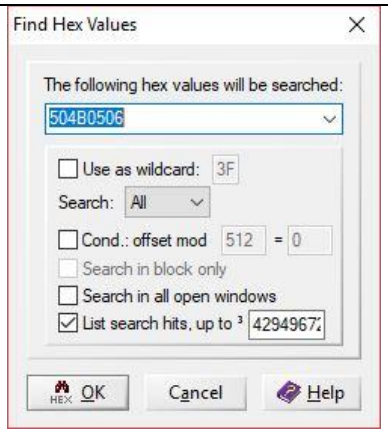
Κατόπιν ο διαχειριστής θέσης του εργαλείου (Position Manager) κατατάσσει τις θέσεις σύμφωνα με τη διαδοχική σειρά με την οποία εντοπίζονται στο Free Space.

Position Manager (General)

Offset	Search hits	Path	Time
25C9000	504B030414000600		10/06/2018 13:50:26
25C939F	504B030414000600		10/06/2018 13:50:26
25C96C3	504B030414000600		10/06/2018 13:50:26
25C98FF	504B030414000600		10/06/2018 13:50:26
25C9B46	504B030414000600		10/06/2018 13:50:26
25CA20F	504B030414000600		10/06/2018 13:50:26
25CA534	504B030414000600		10/06/2018 13:50:26
25CA6E9	504B030414000600		10/06/2018 13:50:26
25CA7D6	504B030414000600		10/06/2018 13:50:26
25CAA96	504B030414000600		10/06/2018 13:50:26
25CAD41	504B030414000600		10/06/2018 13:50:26
25D1E59	504B030414000600		10/06/2018 13:50:26
3763900	H@ □□!Q5		06/06/2018 19:45:41
3763948	X□□8Zone.Identifier[ZoneTransfer] ZoneId=3		06/06/2018 19:46:10

Κατάταξη θέσεων των αποτελεσμάτων της αναζήτησης

Στη συνέχεια θα αναζητηθεί η χαρακτηριστική τιμή κατάληξης (file footer) 0x50 4B 05 06 με την ίδια επιλογή κατάταξης των αποτελεσμάτων.



Αναζήτηση τιμής κατάληξης για το αναζητούμενο αρχείο κειμένου

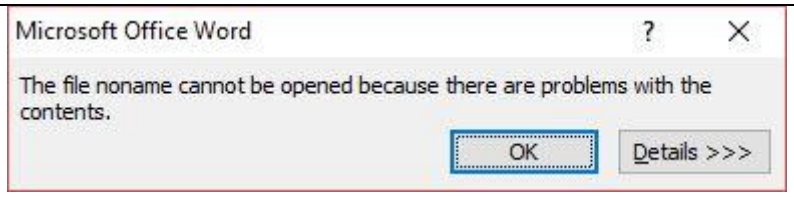
Σημειώνεται ότι η τιμή κατάληξης των αρχείων κειμένου διαμόρφωσης docx, φέρει συνολικά 22 bytes εκ των οποίων τα αρχικά τέσσερα είναι τα προαναφερθέντα και τα υπόλοιπα δεκαοκτώ μετά από αυτά είναι τυχαία κάθε φορά για κάθε διαφορετικό αρχείο κειμένου docx. Συνεπώς αναζητείται η ανωτέρω σταθερή τιμή και κατά την τελική επιλογή του περιεχομένου του αρχείου θα γίνει η απαιτούμενη αναπροσαρμογή στα bytes του. Το εργαλείο παράγει μήνυμα εύρεσης της τιμής μόλις μία φορά, γεγονός από το οποίο συμπεραίνεται ότι μόλις ένα είναι το αρχείο κειμένου διαμόρφωσης pdf που βρίσκεται στον μη κατανεμημένο χώρο και τα υπόλοιπα έντεκα επιπλέον αποτελέσματα της τιμής επικεφαλίδας είναι ψευδώς θετικά.

Αποτελέσματα αναζήτησης τιμής κατάληξης

Η χειροκίνητη βασική διαδικασία χάραξης δεδομένων συνεχίζει με την περιήγηση στην θέση κάθε ανευρεθείσας τιμής επικεφαλίδας, και την επιλογή όλων των bytes που περικλείονται από αυτή και την τιμή κατάληξης. Στο συγκεκριμένο παράδειγμα λόγω του ότι γνωρίζουμε το αρχικό μέγεθος του αρχείου πριν τη διαγραφή του θα μπορούσε αυτή η διαδικασία να απλοποιηθεί. Σε κάθε διαφορετική όμως περίπτωση θα πρέπει να τηρείται στην ολότητά της.

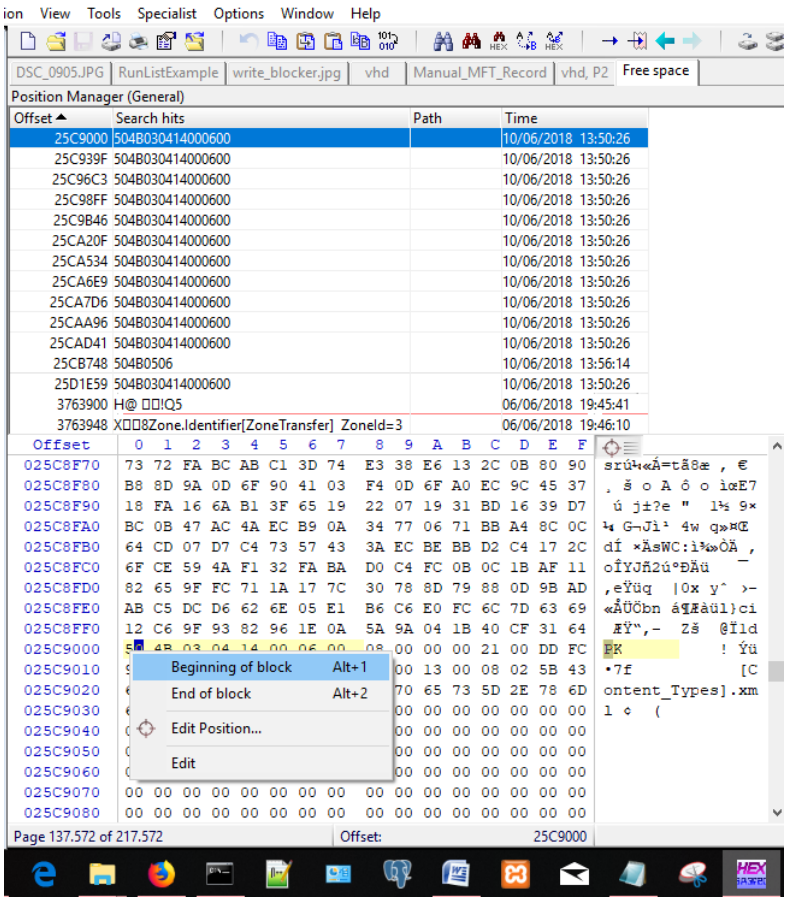
Συνεχίζοντας τη διαδικασία παρατηρούμε ότι τρείς τιμές κατάληξης βρίσκονται μετά την τιμή footer και ως αυτού, αμέσως αποκλείονται γιατί δεν θα μπορούσαν ποτέ να περιγράφουν ένα συνεχές αρχείο στον τόμο. Ένα σημείο που μπορεί να παρατηρηθεί επίσης είναι το γεγονός ότι όλες οι εναπομείνασες θέσεις τιμών επικεφαλίδας βρίσκονται μεταξύ της αρχικής και της τιμής κατάληξης. Δεδομένου ότι τα αρχεία διαμόρφωσης docx ως σύνθετα αρχεία φέρουν εσωτερική δομή διαμέρισης θα μπορούσαμε να υποθέσουμε ότι όλα τα ενδιάμεσα σημεία περιγράφουν τα επιμέρους δοχεία (containers) του κυρίως αρχείου. Αυτή η προσέγγιση μπορεί να παράσχει μία πρώτη ένδειξη για το ποια από όλες τις θέσεις header είναι η σωστή, άλλα θα πρέπει να επιβεβαιωθεί εργαστηριακά με τη δοκιμή εξαγωγής όλων των δυνητικών αρχείων που προσδιορίζονται από τις ανευρεθείσες τιμές.

Κάθε ενδιάμεση τιμή θα οδηγήσει στην εξαγωγή ενός μη λειτουργικού αρχείου το οποίο δεν μπορεί να προσπελαστεί από το οικείο πρόγραμμα.



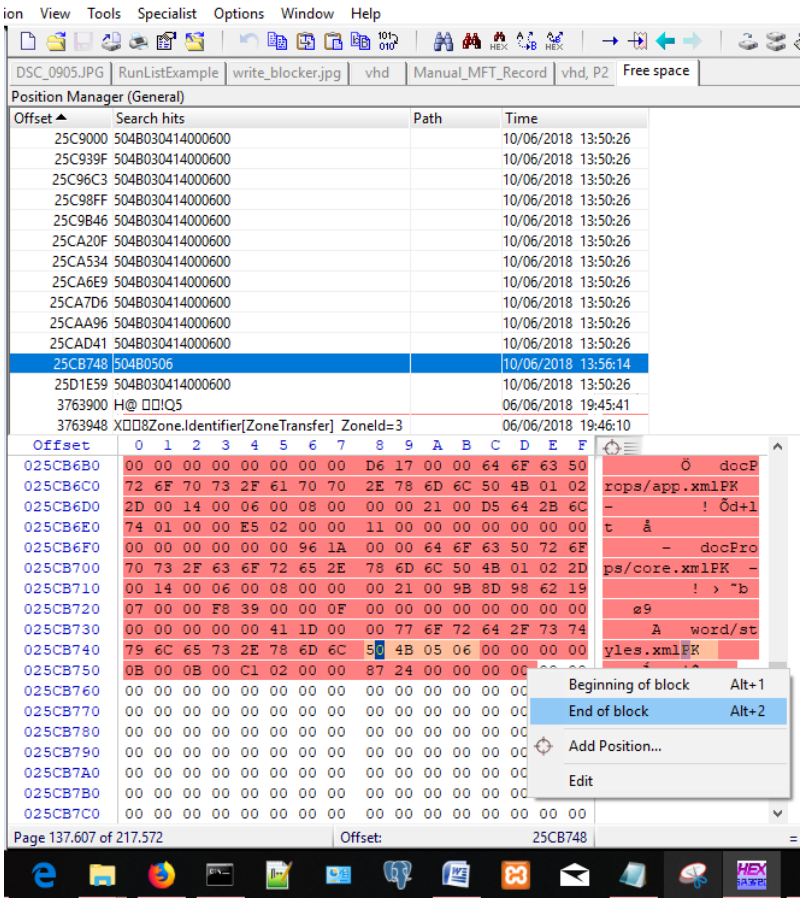
Αποτέλεσμα προσπέλασης ημιτελούς ανακτημένου αρχείου

Μεταβαίνοντας στη θέση 0x25C9000 όπου έχει ανευρεθεί η πρώτη επικεφαλίδα σημαίνουμε το πρώτο byte αυτής ως αρχή του μπλοκ.



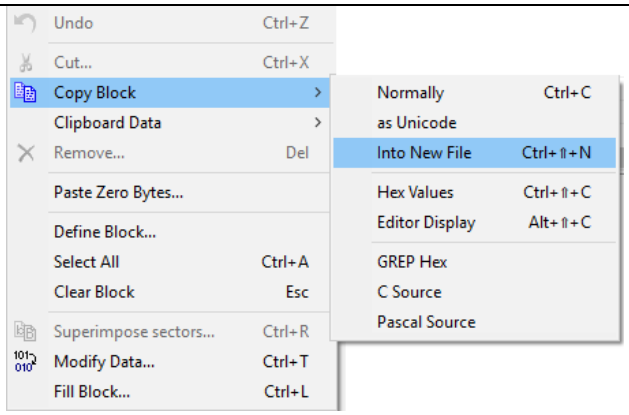
Οριοθέτηση αρχείου με ορισμό σημείου έναρξής του

Στη συνέχεια, επιλέγοντας στον διαχειριστή θέσης την τιμή footer, μεταφερόμαστε στη θέση 0x25CB748 του μη κατανεμημένου χώρου. Σε αυτό το σημείο βρίσκεται η αρχή της τιμής κατάληξης. Φροντίζουμε να επιλέξουμε το byte που βρίσκεται δέκα οκτώ θέσεις μετά το τέλος αυτής ούτως ώστε να περικλείσουμε ολόκληρη την τιμή footer και κατ' επέκταση το σύνολο του αρχείου. Σημαίνουμε αυτό το byte ως τέλος του μπλοκ.



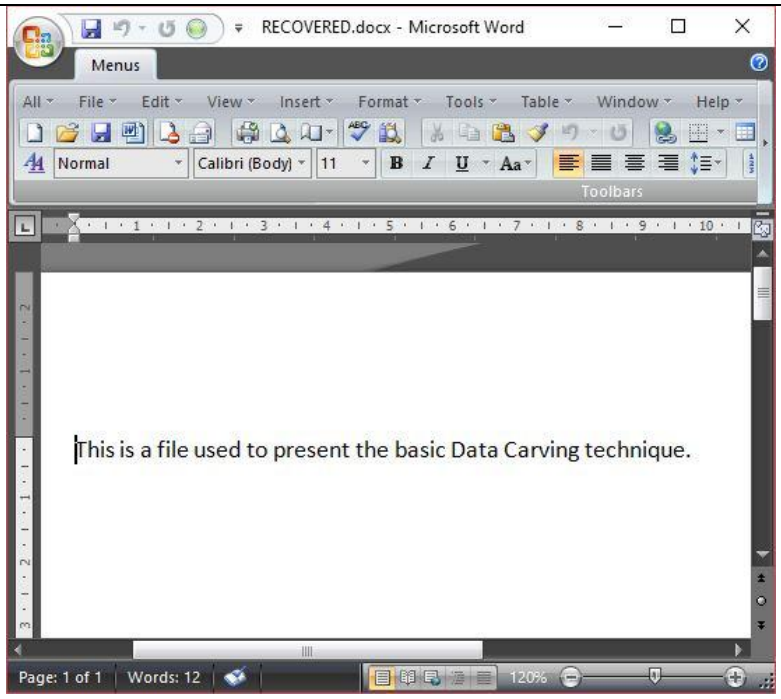
Οριοθέτηση αρχείου με ορισμό σημείου λήξης του

Εν συνεχεία με την εντολή Copy Block -> Into New File επιτυγχάνεται η εξαγωγή αυτού του επιλεγμένου μπλοκ bytes, σε ένα νέο αρχείο στον υπολογιστή εξέτασης.



Αντιγραφή μπλοκ επιλεγμένων bytes σε νέο αρχείο

Το ανακτηθέν αρχείο ονομάζεται RECOVERED και του δίνεται η επέκταση [.docx]. Είναι πλήρως λειτουργικό καθώς όλα τα clusters που καταλάμβανε πριν τη διαγραφή του παρέμειναν άθικτα στον μη κατανεμημένο χώρο μέχρι την πραγματοποίηση της διαδικασίας χάραξης.



Τελικό ανακτηθέν αρχείο

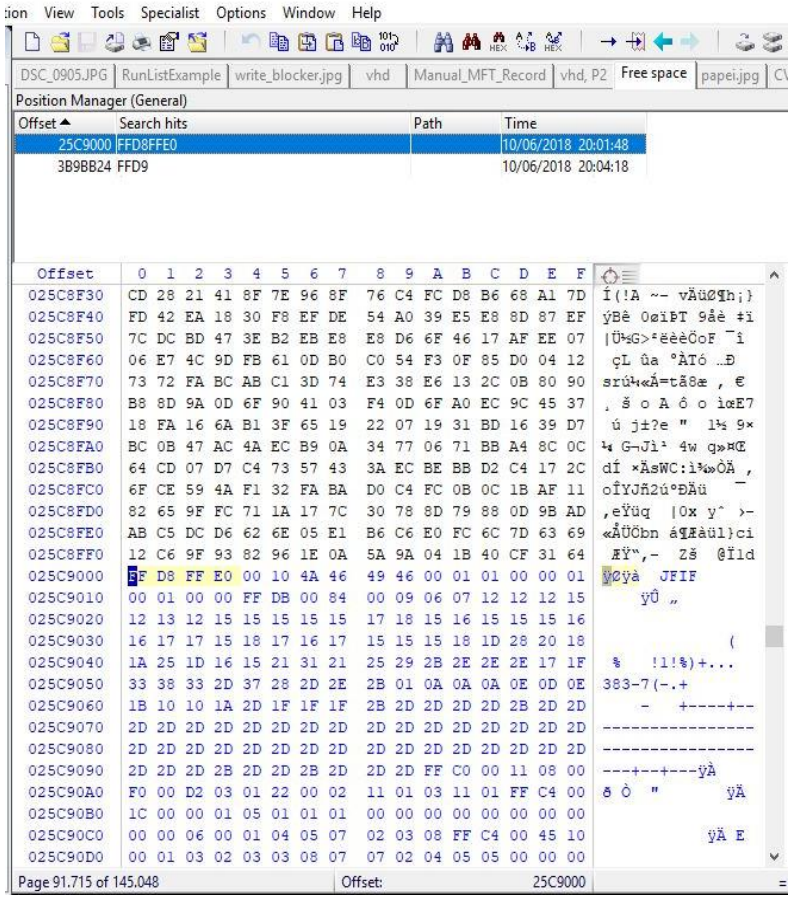
3.4 Data Carving Κατακερματισμένου Αρχείου

Το κύριο μειονέκτημα που έχει η βασική τεχνική χάραξης δεδομένων είναι ότι δεν δύναται να ανακτήσει κατακερματισμένα αρχεία. Το περιεχόμενο του αρχείου πρέπει να είναι κατανεμημένο σε συνεχόμενα, διαδοχικά clusters του τόμου ώστε να μπορέσουν να λαξευτούν μέσα από τον μη κατανεμημένο χώρο. Σε κάθε διαφορετική περίπτωση η τεχνική εξαγωγής που ακολουθείται θα δημιουργήσει αρχεία που φέρουν επιπλέον άχρηστη πληροφορία, καταστρέφοντας τη μορφή του αρχικού συνόλου δεδομένων.

Ορισμένες κατηγορίες αρχείων μπορούν να ανακτηθούν μερικώς και ένα μόλις κομμάτι των δεδομένων τους να προσπελαστεί. Σε αυτές τις περιπτώσεις το ανακτηθέν αρχείο δε μπορεί να έχει καμία εγκληματολογική αξία αλλά ενδεχομένως αρκεί για να ενισχύσει υποθετικά σενάρια που έχουν δημιουργηθεί κατά το προκαταρκτικό στάδιο της εξέτασης.

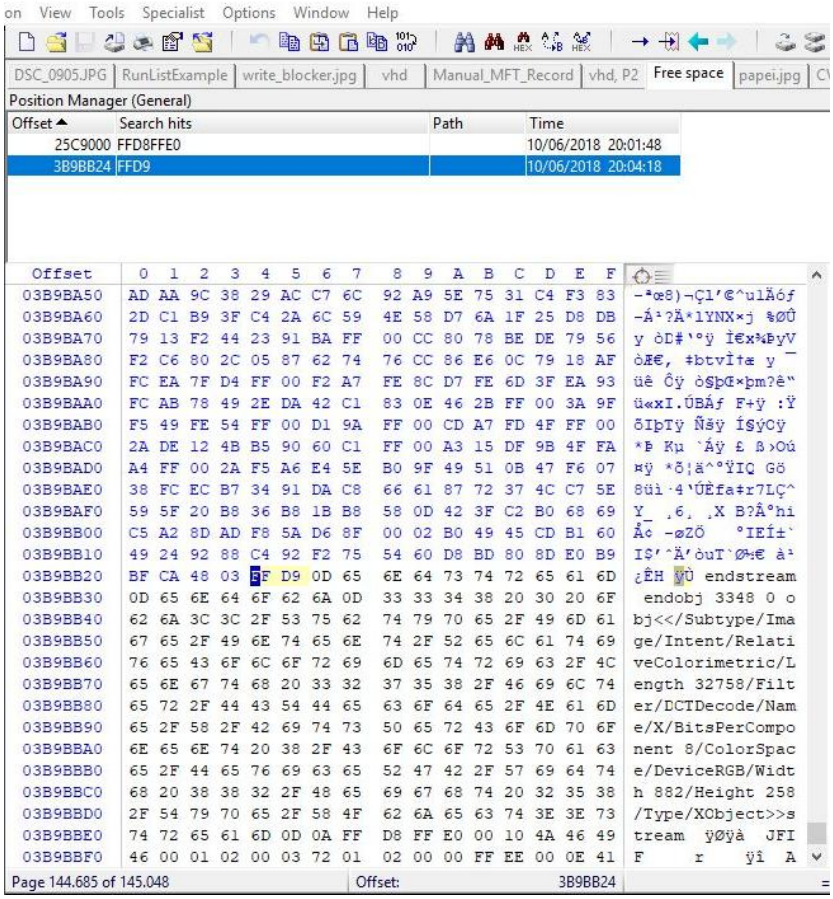
Το παρόν πρακτικό θα παρουσιάσει την μερική ανάκτηση ενός αρχείου εικόνας jpeg, το οποίο αποτελεί μία εκ των κατηγοριών που αναφέρθηκαν. Η διαδικασία χάραξης του από τον μη κατανεμημένο χώρο δεν διαφέρει καθόλου από αυτή που αναπτύχθηκε στο μέρος 3.3 του παρόντος κεφαλαίου.

Η αναζήτηση των τιμών επικεφαλίδας και κατάληξης έχουν αποφέρει από ένα αποτέλεσμα για την κάθε κατηγορία. Οι θέσεις των τιμών αποτυπώνονται στον διαχειριστή θέσης (Position Manager) του εγκληματολογικού εργαλείου. Στη θέση 0x25C9000 παρατηρούμε το header του αρχείου εικόνας με την χαρακτηριστική τιμή 0xFF D8 FF E0.



Θέση τιμής επικεφαλίδας για το αρχείο εικόνας jpeg

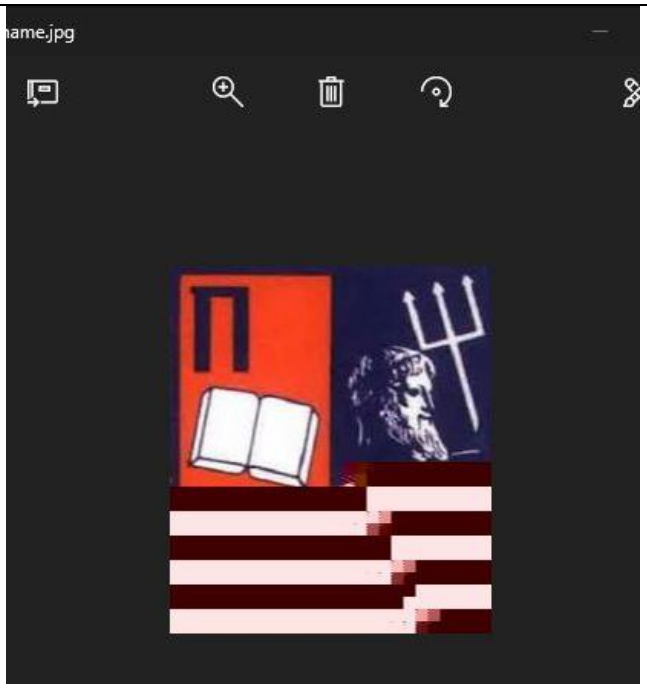
Στη θέση 0x3B9BB24 παρατηρούμε το footer του αρχείου εικόνας με την χαρακτηριστική τιμή 0xFF D9.



Θέση τιμής κατάληξης για το αρχείο εικόνας jpeg

Δεν μένει παρά να επιλεγούν όλα τα ενδιάμεσα bytes που περικλείονται από τις δύο ανευρεθείσες τιμές και να εξαχθεί το νέο αρχείο στον εργαστηριακό υπολογιστή που λαμβάνει χώρα η έρευνα.

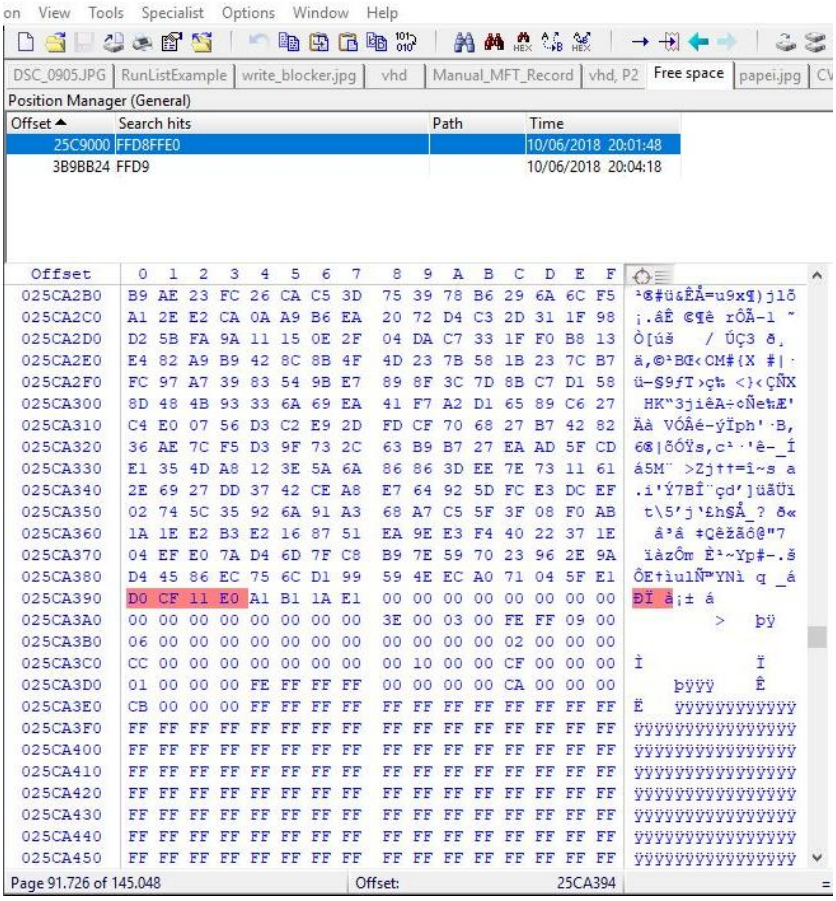
Αφού πραγματοποιηθούν όλα τα απαραίτητα βήματα όπως αναλύθηκαν στο μέρος 3.3, εξάγεται το τελικό αρχείο εικόνας από τον μη κατανεμημένο χώρο του τόμου. Η προσπάθεια να προσπελαστεί το αρχείο με το οικείο πρόγραμμα, δεν παράγει κανένα μήνυμα λάθους παρόλαυτα η εικόνα που εμφανίζεται έχει την ακόλουθη μορφή:



Τελικό ανακτηθέν αρχείο

Αυτή η εικόνα αποτελεί χαρακτηριστικό δείγμα ημιτελούς ανακτημένου αρχείου jpeg. Το μέρος των δεδομένων που έχουν διαβαστεί από το Σύστημα μέχρι το πρώτο σημείο κατακερματισμού του αρχείου, παρουσιάζεται άθικτο στον χρήστη. Το επόμενο κομμάτι των δεδομένων δεν έχει τοποθετηθεί στη σωστή θέση, καίτοι αυτό βρίσκεται ανάμεσα στο σύνολο των ανακτηθέντων δεδομένων. Η σωστή σειρά επανασύνδεσης των κομματιών του αρχείου είναι απαραίτητο στοιχείο ώστε η βασική τεχνική χάραξης να αποδώσει το ορθό αποτέλεσμα.

Διατρέχοντας το περιεχόμενο του ανακτηθέντος αρχείου διαπιστώνουμε ότι στη θέση 0x025CA390, θέση που εντοπίζεται πριν την εύρεση της τιμής footer του αρχείου jpeg, υπάρχει η δεκαεξαδική τιμή 0xD0 CF 11 E0. Αυτή η τιμή αποτελεί τιμή επικεφαλίδας των σύνθετων αρχείων OLE (Object Linking and Embedding – doc, xls, ppt κλπ) και αποδεικνύει ότι το αρχείο που επιχειρείται να ανακτηθεί είναι κατακερματισμένο, αφού άλλο αρχείο παρεμβάλλεται μεταξύ του συνόλου των δεδομένων του.



Αρχείο doc που παρεμβάλλεται μεταξύ των clusters του ανακτηθέντος αρχείου

Η έτερη χειροκίνητη μέθοδος χάραξης, η μέθοδος επικεφαλίδας/μέγιστου μεγέθους, αποτυγχάνει και πάλι σε περιπτώσεις κατακερματισμένων αρχείων αφού η προβληματική της ύπαρξης ενδιάμεσων κομματιών δεδομένων στο σύνολο του αρχείου δεν μπορεί και πάλι να επιλυθεί. Πιο εξειδικευμένες τεχνικές χάραξης δεδομένων μπορούν να ανακτήσουν κατακερματισμένα αρχεία (π.χ. τεχνική SmartCarving), υλοποιούν όμως πολύπλοκους αλγορίθμους αναγνώρισης οι οποίοι απαιτούν αυτοματοποιημένα λογισμικά για να εφαρμοστούν.

Συμπεράσματα - Περίληψη

Στην επιστήμη των υπολογιστών πάντα ακολουθείται μία αφαιρετική προσέγγιση για την ανάλυση των δομών της. Κάθε εσωτερικό επίπεδό αναλύεται με απλουστευμένες διαδικασίες συναρτήσει κάποιου ανώτερου και για κάθε θεμελιώδη έννοια γίνεται προσπάθεια προσέγγισης με αφηρημένο τρόπο. Αυτή η αφαιρετική φιλοσοφία ενυπάρχει ακόμα και στον ίδιο τον πυρήνα δημιουργίας των διάφορων Λειτουργικών Συστημάτων, των Πρωτοκόλλων Μεταφοράς Δεδομένων και Πληροφοριών, των Δικτύων Υπολογιστών κλπ.

Ο τομέας της Ψηφιακής Εγκληματολογίας αντιστρέφει όλη αυτή την παραδοσιακή προσέγγιση. Το ζητούμενο της είναι ακριβώς η βαθύτερη ανάλυση των δομών, η επικέντρωση

στο εσωτερικό επίπεδο πληροφορίας και η παραγωγή πλήρως αιτιολογημένων αναφορών σχετικά με το τι ακριβώς συνέβη στον πυρήνα ενός υπολογιστικού συστήματος. Μόνο η βαθύτερη ανάλυση των ευρημάτων και η υποστήριξή τους με αδιαμφισβήτητα αποδεικτικά στοιχεία μπορούν να θωρακίσουν τα αποτελέσματα μίας ψηφιακής εγκληματολογικής έρευνας στις δικαστικές αίθουσες. Και για να ανευρεθούν αδιαμφισβήτητα αποδεικτικά στοιχεία ο εξεταστής θα πρέπει να φτάσει στο βαθύτερο επίπεδο του εξεταζόμενου συστήματος, τηρώντας όλες τις θεμελιώδεις αρχές της Ψηφιακής Εγκληματολογίας. Σε αυτό το επίπεδο ο χρήστης δε δύναται να αλλοιώσει καμία πληροφορία αυτοβούλως και συνεπώς το ίδιο το υπό εξέταση μέσο δίνει αδιάψευστες απαντήσεις στα ερωτήματα της έρευνας.

Αυτό το επίπεδο είναι το επίπεδο του Συστήματος Αρχείων. Της ιεραρχικής δομής που αποτελεί τη ραχοκοκαλιά του μέσου και δημιουργεί τις συνθήκες εύρυθμης λειτουργίας του υπολογιστικού συστήματος. Μία πλειάδα διαφορετικών Συστημάτων Αρχείων υπάρχει για κάθε διαφορετικό είδος Λειτουργικού Συστήματος. Η γενική φιλοσοφία η ίδια, μα η υλοποίηση από κάθε εταιρεία τελείως διαφορετική.

Από την δημιουργία του πρώτου Συστήματος Αρχείων, σχεδόν μισό αιώνα πριν, μέχρι και σήμερα, η εξέλιξη που έχει σημειωθεί είναι αξιοσημείωτη. Από τη βασική τους λειτουργικότητα, ως απλώς συστήματα αποθήκευσης και εύρεσης των αρχείων μέσα στον τόμο, τα Συστήματα Αρχείων έχουν πλέον εξελιχθεί σε τέτοιο βαθμό ώστε να προσφέρουν αυξημένες δυνατότητες ασφάλειας και ιδιωτικότητας, κρυπτογράφησης, αποδοτικότητας αποθηκευτικού χώρου κ.α.

Ένα από τα πλέον εξελιγμένα Συστήματα Αρχείων είναι και το Σύστημα Αρχείων NTFS. Προεπιλεγμένο Σύστημα διαμόρφωσης των μέσων για την πλέον επιτυχημένη εμπορικά εταιρεία παγκοσμίως, το NTFS αποτελεί το πιο ευρέως διαδεδομένο Σύστημα Αρχείων τη δεδομένη χρονική στιγμή στον κόσμο. Ως αυτού, η εγκληματολογική εξέταση του αποτελεί ένα μόνιμο θέρετρο επιχειρήσεων για τον εγκληματολόγο εξεταστή. Η αυξημένη δημοτικότητά του συμβαδίζει αναλόγως με την αυξημένη πολυπλοκότητά του.

Ένα πλήθος εσωτερικών δομών καθιστά το NTFS ένα Σύστημα ασφαλές, αποδοτικό, σταθερό και επεκτάσιμο. Κάθε δομή πληροφορίας αντιμετωπίζεται ως αρχείο, χωρίς διαχωρισμό μεταξύ αρχείων χρήστη ή συστήματος. Τα αρχεία συστήματος ονομάζονται αρχεία μεταδεδομένων και αποτελούν τη βάση της λειτουργίας του NTFS. Σημαντικότερα αρχεία μεταδεδομένων είναι ο τομέας εκκίνησης, που παρέχει όλη τη πληροφορία με την οποία ο τόμος μπορεί να αναλυθεί από το Λειτουργικό Σύστημα και ο πίνακας $MFT που περιέχει εγγραφές για όλα τα αρχεία που περιέχονται στον τόμο. Και τα δύο αυτά αρχεία φέρουν αντίγραφο ασφαλείας σε περίπτωση αλλοίωσης του αρχικού, συμβάλλοντας με αυτό τον τρόπο στην αξιοπιστία του Συστήματος.

Η περιγραφή των λοιπών αρχείων του τόμου, λαμβάνει χώρα εντός του πίνακα $MFT με τη βοήθεια των χαρακτηριστικών αρχείων, που είναι δομές αυστηρά ορισμένες οι οποίες προσφέρουν πληροφορία για το κάθε αρχείο. Η πληροφόρηση που παρέχει το Σύστημα Αρχείων NTFS για κάθε αρχείο αποθηκευμένο σε αυτό είναι πολύπλευρη και συμβάλλει στην αυξημένη ασφάλεια του Συστήματος.

Τρία εκ των χαρακτηριστικών αυτών, κατέχουν τον σημαντικότερο ρόλο στην πιο συχνή παραγγελία στην οποία καλείται να παράσχει απαντήσεις μία εγκληματολογική εξέταση, την ανάκτηση διαγεγραμμένων αρχείων. Η λειτουργικότητα του Συστήματος Αρχείων NTFS είναι προσανατολισμένη στην ταχύτητα και ως αυτού η διαγραφή ενός αρχείου τροποποιεί μόνο τις απολύτως απαραίτητες δομές ώστε το Σύστημα να δίνει την εντύπωση στον χρήστη ότι το αρχείο διαγράφηκε. Στην πραγματικότητα τα δεδομένα ενός αρχείου παραμένουν στον τόμο πολύ καιρό μετά τη διαγραφή του. Η ανάλυση των τριών χαρακτηριστικών που περιγράφουν το αρχείο στον πίνακα $MFT οδηγεί στην ανάκτηση του περιεχομένου του και στην επιτυχή ολοκλήρωση μίας εγκληματολογικής έρευνας ανάλογου αιτήματος. Αυτά τα χαρακτηριστικά είναι τα $Standard_Information, $File_Name και $Data τα οποία συνδυάζονται για να επιτευχθεί η διαδικασία ανάκτησης του διαγραμμένου αρχείου.

Σε ορισμένες περιπτώσεις, όμως, τα ίδια τα χαρακτηριστικά που συνόδευαν το αρχείο έχουν διαγραφεί και αυτά. Η απώλεια των τριών προαναφερθέντων χαρακτηριστικών, λοιπόν, οδηγεί στη διαφορετική προσέγγιση για την ανάκτηση του αρχείου, με τη χρήση τεχνικών χάραξης δεδομένων από τον μη κατανεμημένο χώρο του τόμου. Αυτή η τεχνοτροπία αποφέρει μειωμένα αποτελέσματα σε σχέση με την ανάκτηση δεδομένων μέσω των χαρακτηριστικών του αρχείου, αλλά δεν παύει να παρέχει απαντήσεις σε έναν ικανό αριθμό εγκληματολογικών εξετάσεων.

Συνοψίζοντας πρέπει να τονιστεί η εξαιρετική σημασία της ψηφιακής πληροφορίας στη σύγχρονη εποχή. Βασικοί πυλώνες κάθε κοινωνίας, όπως η Οικονομία, η Ασφάλεια ή ο Πολιτισμός αποτελούν πλέον έννοιες συνυφασμένες με τον ψηφιακό κόσμο. Και επειδή κάθε πτυχή αυτού του κόσμου είναι το ίδιο ευάλωτη σε κακόβουλες επιθέσεις με απρόβλεπτα, καταστροφικά αποτελέσματα, γεννάται η ανάγκη δημιουργίας δικλείδων ασφαλείας. Μία εξ' αυτών αποτελεί η Ψηφιακή εγκληματολογία.

Η βαθιά γνώση της δομής των Πληροφοριακών Συστημάτων και η άριστη εφαρμογή των θεμελιωδών αρχών της Ψηφιακής εγκληματολογίας αποτελούν τον απαραίτητο συνδυασμό για την επιτυχή ολοκλήρωση μίας ψηφιακής εγκληματολογικής έρευνας, η οποία αποτελεί και το βασικό συστατικό μέρος αυτής της ασφαλιστικής δικλείδας.

Βιβλιογραφία

Bunting, S. (2012). *EnCase Computer Forensics: The Official EnCase Certified Examiner study guide third edition*. ΗΠΑ: John Wiley & Sons Inc.

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

Casey, E. (2011). *Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet 3rd edition*. ΗΠΑ: Academic Press Inc.

Chow, K. & Law, F. & Kwan, M. & Lai, K. (2007). *The Rules of Time on NTFS File System*. Κίνα: The University of Hong Kong, Academic Paper

Chow, K. & Shenoi, S. (2010). *Advances in Digital Forensics VI*. ΗΠΑ: Springer Publishing

IACIS (2017). *Basic Computer Forensic Examiner Book 1 & 2*. Βέλγιο: OLAF

Merola, A. (2008). *Data Carving Concepts*, ΗΠΑ: SANS Institute

Rusbarsky, K. (2012). *A Forensic Comparison of NTFS and FAT32 File Systems*. ΗΠΑ: Marshall University, Research Paper

Sammons, J. (2012). *The Basics of Digital Forensics*. ΗΠΑ: Syngress Publishing

Sammons, J. (2015). *Digital Forensics Threatscape and Best Practices*. ΗΠΑ: Syngress Publishing

Tanenbaum, A. (2009). *Modern Operating Systems*. ΗΠΑ: Pearson Education Inc.

File Systems Technologies (2009). Ανακτήθηκε από https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc778296(v=ws.10)

Hurlbut, D. (2015). *Orphans in the NTFS World*. Ανακτήθηκε από https://www.syntricate.com/files/Orphans in the NTFS World.pdf

Zatyko, K. (2008). *Defining Digital Forensics*. Ανακτήθηκε από https://www.forensicmag.com/article/2008/12/ defining-digital-forensics