# Database Forensics

Σωτήριος – Άγγελος Δ. Λένας [1] Παύλος Σ. Εφραιμίδης[2] Βασίλειος Κάτος[2]

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Δημοκρίτειο Πανεπιστήμιο Θράκης
Κτίριο Α, Πανεπιστημιούπολη, 67100 Ξάνθη, τηλ: 2541079781
{sl2568, pefraimi, vkatos}@ee.duth.gr

1. Προπτυχιακός φοιτητής
2. Μέλος ΔΕΠ

## Περίληψη

*Οι βάσεις δεδομένων αποτελούν ένα από τα πιο κύρια τμήματα των σύγχρονων πληροφοριακών συστημάτων. Λόγω της κρισιμότητας των πληροφοριών που διατηρούν η εξασφάλιση ενός υψηλού επιπέδου ασφάλειας κρίνεται αναγκαία. Αρκετές φορές όμως είτε λόγω ανεπαρκούς επιπέδου ασφάλειας είτε εξαιτίας άλλων παραγόντων το πληροφοριακό σύστημα παραβιάζεται με αποτέλεσμα την μη νόμιμη απόκτηση πρόσβασης και την πιθανότητα αλλοίωσης τόσο του συστήματος διαχείρισης της βάσης όσο και των δεδομένων που είναι αποθηκευμένα σε αυτή. Ο γενικός κλάδος της πληροφορικής ο οποίος ανιχνεύει και μελετά την παραβίαση ασφάλειας ενός πληροφοριακού συστήματος με νομικώς αποδεκτές μεθόδους ονομάζεται Computer Forensics (ψηφιακή σήμανση). Ως προέκταση αυτού, τα τελευταία χρόνια, έχει εμφανιστεί ο κλάδος των Database Forensics (ψηφιακή σήμανση Βάσεων Δεδομένων), ο οποίος χρησιμοποιεί εξειδικευμένες τεχνικές και εργαλεία ώστε να γίνει αποτελεσματικά και με επιτυχία η εξέταση της εκάστοτε παραβιασμένης βάσης δεδομένων. Στο παρόν άρθρο παρουσιάζεται η forensics ανάλυση των βάσεων δεδομένων με επίκεντρο την διαδικασία που πρέπει να ακολουθηθεί από την στιγμή που διαπιστώθηκε παραβίαση της βάσης δεδομένων, γνωστή και ως Live Response. Επίσης γίνεται αναφορά σε δύο θέματα τα οποία συνδέονται άμεσα με την forensics ανάλυση και είναι αυτά της ανίχνευσης παραβίασης – αλλοίωσης της βάσης δεδομένων και της παραβίασης της ιδιωτικότητας. Τέλος γίνεται μια αναφορά στα συνήθη προβλήματα τα οποία καλείται να αντιμετωπίσει ο εκάστοτε αναλυτής forensics και παρουσιάζονται τα συμπεράσματα της εργασίας.*

## Κύρια περιοχή έρευνας

Ασφάλεια Βάσεων Δεδομένων

## Λέξεις Κλειδιά

Database Forensics, Βάσεις Δεδομένων, Computer Forensics, Ασφάλεια Υπολογιστών.

## 1. Εισαγωγή

Οι βάσεις δεδομένων (ΒΔ) αποτελούν ένα βασικό συστατικό των σύγχρονων πληροφοριακών συστημάτων καθώς σε αυτές φυλάσσονται κρίσιμα δεδομένα. Αυτό δημιουργεί την ανάγκη για αυξημένη ασφάλεια των ΒΔ. Παρόλο που λαμβάνονται μέτρα ασφάλειας για τις ΒΔ, στην πράξη, υπάρχουν περιπτώσεις μη νόμιμης απόκτησης πρόσβασης σε αυτές.

Ο γενικός κλάδος της πληροφορικής ο οποίος εξετάζει το θέμα της μη νόμιμης απόκτησης πρόσβασης και αλλοίωσης τόσο της δομής όσο και των αποθηκευμένων δεδομένων ενός πληροφοριακού συστήματος με την χρήση νομικώς αποδεκτών μεθόδων ονομάζεται Computer Forensics (ψηφιακή σήμανση). Με το πέρασμα των χρόνων δημιουργήθηκε η ανάγκη μιας πιο εξειδικευμένης προσέγγισης στο θέμα της παραβίασης μιας ΒΔ η οποία ονομάστηκε Database Forensics (ψηφιακή σήμανση Βάσεων Δεδομένων).

Στο παρόν άρθρο γίνεται μια παρουσίαση της forensics ανάλυσης των ΒΔ, τόσο σε επίπεδο λειτουργικού συστήματος όσο και σε επίπεδο συστήματος διαχείρισης της βάσης, δίνοντας όμως έμφαση στο λεγόμενο Live Response. Με τον όρο Live Response αναφερόμαστε στα βήματα που πρέπει να ακολουθηθούν από την στιγμή που διαπιστώθηκε παραβίαση της ΒΔ. Για το πειραματικό μέρος της εργασίας αυτής επιλέξαμε ένα από τα πιο ευρέως διαδεδομένα Συστήματα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΣ), την Oracle. Παρόμοια στρατηγική αλλά και τεχνικές μπορούν να εφαρμοστούν και σε άλλα δημοφιλή ΣΔΒΔ.

Η ενασχόληση βέβαια με το θέμα της forensics ανάλυσης εγείρει κάποια ενδιαφέροντα ερωτήματα. Ένα από αυτά αφορά την ανίχνευση παραβίασης – αλλοίωσης της ΒΔ. Η παραβίαση – αλλοίωση μιας ΒΔ δεν είναι τόσο προφανής σε αντίθεση με άλλα συστατικά μέρη ενός πληροφοριακού συστήματος όπου η ανίχνευση παραδείγματος χάρη κακόβουλου λογισμικού μπορεί να αποτελέσει ένδειξη παραβίασης. Για παράδειγμα ας θεωρήσουμε τη ΒΔ μιας τράπεζας όπου η αλλαγή της

τιμής μιας εγγραφής μπορεί να έχει καταστροφικές συνέπειες. Για τον λόγο αυτό γίνεται μια σύντομη αναφορά σε μια τεχνική πρόληψης – ενημέρωσης παραβίασης της ΒΔ. Ένα δεύτερο ερώτημα που προκύπτει είναι αυτό της παραβίασης της ιδιωτικότητας. Η παραβίαση του ιδιωτικού απόρρητου τόσο με την χρήση των ΒΔ όσο και με την forensics ανάλυση αποτελεί ένα υπαρκτό σενάριο του οποίου οι πτυχές σχολιάζονται.

Στο τέλος του παρόντος άρθρου γίνεται μια αναφορά στα συνήθη προβλήματα τα οποία καλείται να αντιμετωπίσει ο εκάστοτε αναλυτής forensics και παρουσιάζονται τα συμπεράσματα της εργασίας.

## 2. Computer forensics

Με βάση τον ορισμό των Farmer και Venema [2], ο όρος Computer Forensics αναφέρεται στη συγκέντρωση και ανάλυση δεδομένων με τέτοιο τρόπο ώστε να αποφευχθεί η διαστρέβλωση στοιχείων και να καταστεί δυνατή η ανακατασκευή των δεδομένων ή η εξακρίβωση του τι συνέβη στο παρελθόν σε ένα σύστημα.

Τα βήματα τα οποία ακολουθεί γενικά ένας αναλυτής forensics είναι τα εξής [2]:

- Εκκίνηση μιας τεκμηριωμένης υπόδειξης - καταγραφής ως προς το χρόνο, των βασισμένων σε υπολογιστή γεγονότων.
- Ανίχνευση και προσδιορισμός του περιστατικού παραβίασης.
- Δημιουργία πλάνου αντίδρασης.
- Δημιουργία αντιγράφων ασφαλείας των ηλεκτρονικών αρχείων ως αποδεικτικά στοιχεία ώστε να είναι νομικώς αποδεκτά.
- Ανάκτηση υπηρεσιών και διαστρεβλωμένων δεδομένων (αλλοίωση ή διαγραφή).
- Συλλογή και ταξινόμηση των ηλεκτρονικών μεταδεδομένων ως προς τον χρόνο.
- Ενσωμάτωση όλων των πληροφοριών στην χρονική εξέλιξη των γεγονότων. Αυτή η διαδικασία συμπεριλαμβάνει και την συλλογή – παρουσίαση των αρχείων log.
- Ανάλυση του χρόνου εξέλιξης των μεταδεδομένων.
- Λεπτομερής εξέταση των block - δεδομένων σε χαμηλό επίπεδο.
- Λεπτομερής καταγραφή της όλης διαδικασίας ούτως ώστε να μπορεί να επαναληφτεί.
- Παρουσίαση των αποδείξεων σε ένα νομικό πλαίσιο.

Από τα παραπάνω βήματα λοιπόν γίνεται κατανοητό πως στην forensics ανάλυση ένα από τα πιο σημαντικά στοιχεία είναι ο χρόνος και το επόμενο είναι η αντίδραση. Ο χρόνος αποτελεί σημαντικό παράγοντα τόσο ως προς την παρουσίαση εξέλιξης των γεγονότων όσο και ως προς τον περιορισμό της ζημιάς αφού όσο πιο σύντομα διαπιστωθεί η παραβίαση και γίνουν οι κατάλληλες ενέργειες τόσο καλύτερα αποτελέσματα αναμένονται. Τι εννοούμε όμως λέγοντας αντίδραση;

Πολλές φορές σε περίπτωση παραβίασης η πρώτη κίνηση είναι το τράβηγμα της πρίζας ή η αποσύνδεση του συστήματος από το δίκτυο. Αυτό αποτρέπει τις περαιτέρω εισβολές και την κλοπή δεδομένων για αυτό και φαίνεται ως μια λογική αντίδραση. Όμως συχνά η κίνηση αυτή είναι λανθασμένη διότι με την λήψη αυτών των μέτρων, χρήσιμα στοιχεία όπως τα δεδομένα που βρίσκονται στην μνήμη μπορούν να χαθούν. Αυτό το παράδειγμα μας δίνει μια γεύση για το τι σημαίνει αντίδραση αλλά εισάγει επίσης και το πρώτο δίλημμα· σε περίπτωση παραβίασης ποια θα είναι η πρώτη μας κίνηση, τερματίζουμε την βάση ή όχι; Η forensics ανάλυση περιλαμβάνει σε κάθε της βήμα τέτοια διλήμματα (trade-offs) κάτι που την καθιστά ακόμα πιο ενδιαφέρουσα.

Τέλος ένα ακόμη σημείο στο οποίο πρέπει να σταθούμε είναι ότι ο εκάστοτε αναλυτής forensics φροντίζει να περιορίσει την εργασία του πάνω στο πραγματικό σύστημα στο ελάχιστο δυνατό. Για αυτό το λόγο μια από τις πρώτες του κινήσεις είναι η δημιουργία αντίγραφων ασφαλείας ώστε η εκτέλεση των περαιτέρω ενεργειών της ανάλυσης να γίνει πάνω σε αυτά. Επιπλέον τα αντίγραφα ασφάλειας μπορούν να φανούν χρήσιμα και σε πιθανό αίτημα επανάληψης της όλης διαδικασίας.

## 3. Database forensics

Database Forensics είναι η επιστήμη η οποία έχει αναλάβει την διαδικασία συγκέντρωσης και ανάλυσης στοιχείων από μια ΒΔ με τρόπο τέτοιο ώστε αυτά να είναι νομικώς αποδεκτά και φυσικά προσπαθώντας κατά την διάρκεια της διαδικασίας αυτής να εξακριβώσει το τι συνέβη στο παρελθόν σε αυτή τη ΒΔ. Αυτό επιτυγχάνεται με εργαλεία και τεχνικές που αφορούν μόνο τη ΒΔ αλλά και με την εφαρμογή της κλασσικής επιστήμης των Computer Forensics.

Όσο αφορά το Live Response, είναι αδύνατο για έναν αναλυτή forensics να μην αφήσει κανένα ίχνος στο σύστημα το οποίο ερευνά. Σκοπός του λοιπόν είναι να καταστήσει αυτό το ίχνος όσο το δυνατόν μικρότερο και να είναι σε θέση να υποστηρίξει τις αποφάσεις – ενέργειες του που είχαν ως συνέπεια την μεταβολή του υπό εξέταση συστήματος. Παραδείγματος χάριν, η πράξη της σύνδεσης από μια κονσόλα στον υπολογιστή προς εξέταση θα προκαλέσει ενδεχομένως την εγγραφή κάποιων δεδομένων στο ίχνος ελέγχου (audit trail) του συστήματος, επίσης ένα κέλυφος θα ανοίξει κάτι που θα έχει επιπτώσεις στη μνήμη και θα προκαλέσει τροποποιήσεις στο αρχείο σελιδοποίησης. Αλλαγές βέβαια όπως αυτές είναι αναπόφευκτες αλλά θεωρούνται αποδεκτές. Αυτό που δεν είναι αποδεκτό παραδείγματος χάρη είναι η δημιουργία νέων αρχείων στο σύστημα.

Αυτό θα μπορούσε ενδεχομένως να προκαλέσει την αλλοίωση κάποιων block δεδομένων ενός δίσκου που περιείχε τα διαγραμμένα δεδομένα τα οποία βέβαια αν αυτό δεν είχε συμβεί θα ήταν ανακτήσιμα κατά τη διάρκεια εξέτασης του συστήματος. Επίσης όλοι οι έξοδοι από τα εργαλεία που χρησιμοποιούνται για το Live Response θα πρέπει να αποθηκεύονται σε κάποια άλλη τοποθεσία είτε μέσω δικτύου είτε μέσω κάποιας άλλης περιφερειακής συσκευής.[1]

Η διαδικασία του Live Response χωρίζεται σε δύο τμήματα. Το πρώτο τμήμα περιλαμβάνει τα στάδια της διαδικασίας τα οποία συνδέονται γενικά με το λειτουργικό σύστημα ενώ το δεύτερο τμήμα περιλαμβάνει τα στάδια τα οποία συνδέονται ειδικά με την εκάστοτε ΒΔ και γίνονται μέσω αυτής. Χρονικά προηγούνται τα βήματα που σχετίζονται με το λειτουργικό σύστημα και ακολουθούν αυτά που αφορούν ειδικά τη ΒΔ.

### 3.1 Live Response λειτουργικού συστήματος.

Τα κυριότερα βήματα του Live Response που αφορούν το λειτουργικό σύστημα είναι τα εξής [1]:

a. *Ημερομηνία και ώρα συστήματος.*

Η πρώτη κίνηση του αναλυτή forensics είναι η καταγραφή της ώρας και της ημερομηνίας του ως προς εξέταση συστήματος.

*b. Καταγραφή των χρηστών που είναι συνδεδεμένοι.*

Είναι πολύ χρήσιμο (ανάλογα με το πόσο σύντομη είναι η αντίδραση) να ξέρουμε ποιοι χρήστες είναι συνδεδεμένοι στο σύστημα, από ποια γεωγραφική περιοχή και πόση ώρα είναι συνδεδεμένοι.

*c. Συλλογή λίστας με όλους τους χρήστες και τα γκρουπ του συστήματος με αναλυτικές πληροφορίες για το καθένα.*

*d. Ανοιχτές θύρες (ports) και συνδέσεις.*

Μια από τις πιο σημαντικές ενέργειες του αναλυτή forensics κατά την διάρκεια εξέτασης του λειτουργικού συστήματος είναι η καταγραφή των ανοικτών θυρών και συνδέσεων του συστήματος και αυτό διότι πολλές φορές οι πληροφορίες που γίνονται γνωστές από την καταγραφή όλων των ανοιχτών θυρών και συνδέσεων του συστήματος επιτρέπουν την εξακρίβωση του πώς ένας επιτιθέμενος έχει κατορθώσει να αποκτήσει ή να διατηρήσει πρόσβαση στο σύστημα. Αξιοσημείωτο είναι ότι μια ή περισσότερες από αυτές τις συνδέσεις θα είναι η σύνδεση του αναλυτή στο προς ανάλυση σύστημα. Μεγάλη προσοχή επίσης πρέπει να δοθεί στις συνδέσεις που αφορούν τον server της εκάστοτε ΒΔ.

Ένα σημαντικό ερώτημα που προκύπτει συνήθως και σχετίζεται ως ένα βαθμό με την παραπάνω ενέργεια είναι η αποσύνδεση του παραβιασμένου συστήματος από το δίκτυο ή όχι. Εάν η απάντηση είναι καταφατική, υπάρχει η πιθανότητα με μερικές εκδόσεις λειτουργικών, ο τερματισμός των ενεργών συνδέσεων να διαγράψει και όλες τις πληροφορίες που σχετίζονται με αυτές. Στην περίπτωση βέβαια που δεν αποσυνδεθεί από το δίκτυο και ο επιτιθέμενος είναι ακόμα συνδεδεμένος τότε το σενάριο της επέκτασης της ζημιάς είναι υπαρκτό. Από την άλλη πλευρά, παρέχει στον αναλυτή την δυνατότητα να συγκεντρώσει στοιχεία για την επίθεση την στιγμή που αυτή συμβαίνει. Εάν η αξία των στοιχείων που υποκλέπτονται είναι πολύ μεγάλη και η διαρροή τους είναι καταστροφική τότε το σενάριο της αποσύνδεσης αποκτά ένα πλεονέκτημα, από την άλλη όμως και η αποσύνδεση ενός συστήματος το οποίο εξυπηρετεί χιλιάδες αιτήματα το λεπτό είναι και αυτό μια δαπανηρή ενέργεια. Η εξισορρόπηση των παραπάνω πλεονεκτημάτων – μειονεκτημάτων και η λήψη της καταλληλότερης απόφασης είναι ευθύνη του αναλυτή forensics.

*e. Συλλογή λίστας με όλες τις τρέχουσες διαδικασίες του λειτουργικού.*

Από εδώ ο αναλυτής μπορεί μετά από ανάλυση να καταλάβει αν η παραβίαση του συστήματος ή της βάσης έγινε από ένα exploit υπερχείλισης.

*f. Δημιουργία ενός πλήρους αντιγράφου της μνήμης RAM.*

*g. Πλήρης ανάκτηση φακέλων και αρχείων του σκληρού δίσκου καθώς και τις ιδιότητες – άδειες αυτών. Δημιουργία ενός ένα προς ένα αντιγράφου του φυσικού μέσου αποθήκευσης.*

*h. Τέλος εντοπισμός και δημιουργία αντιγράφου των log files και των μηνυμάτων σφάλματος που υπάρχουν σε κάθε λειτουργικό.*

### 3.2 Live Response βάσης δεδομένων.

Η παρουσίαση των βημάτων για την συλλογή στοιχείων από την ίδια τη ΒΔ γίνεται με χρήση της Oracle (έκδοση 11), οπότε τα διάφορα αρχεία, διαδρομές αρχείων και εντολές που αναφέρονται αντιστοιχούν σε αυτά που χρησιμοποιεί η Oracle. Με βάση το άρθρο [1] και για τους σκοπούς του παρόντος άρθρου δοκιμάστηκαν οι παρακάτω ενέργειες για την συλλογή στοιχείων από την ίδια τη ΒΔ:

*a. Συλλογή όλων των αρχείων που σχετίζονται άμεσα με την Oracle.*

Η Oracle αποθηκεύει σε διάφορους - γνωστούς καταλόγους πληροφορίες που σχετίζονται άμεσα με αυτή. Έτσι οι πιο σημαντικοί είναι:

*Audit_file_dest:* Εάν ο έλεγχος είναι ενεργοποιημένος και έχει ρυθμιστεί έτσι ώστε να κρατά στο σύστημα αρχείων του λειτουργικού συστήματος τα αποτελέσματά του, τότε τα log files θα βρίσκονται σε αυτόν τον κατάλογο.

*Background_dump_dest:* Αυτός ο κατάλογος περιέχει το alert.log και τα ίχνη των εργασιών που εκτελούνται στο παρασκήνιο.

*Core_dump_dest:* Τα αρχεία του Oracle server core dumps γράφονται σε αυτή την τοποθεσία. Τα Core αρχεία μπορούν να υποδείξουν μια πιθανή προσπάθεια παραβίασης του συστήματος χρησιμοποιώντας κάποιου είδους υπερχείλισης (buffer overflow).

*Db_recovery_file_dest:* Αυτή είναι η τοποθεσία όπου βρίσκεται η περιοχή του flash recovery area και περιέχει τα archived redo logs.

*Db_create_file_dest, Db_create_online_log_dest_n , Log_archive_dest, log_archive_dest_n and log_archive_duplex_dest:* Τοποθεσίες όπου μπορεί κανείς να βρει redo log αρχεία.

*Αρχεία δεδομένων της Oracle:* Αυτά τα αρχεία περιλαμβάνουν τα πραγματικά δεδομένα της βάσης. Το πρόβλημα που προκύπτει όμως είναι ότι συνήθως το μέγεθος αυτών των αρχείων είναι δυνατό να φτάνει από μερικά terabyte έως ένα petabyte, με αποτέλεσμα να καθιστά την διαδικασία αντιγραφής εξαιρετικά δύσκολη. Για αυτό τον λόγο αποθηκεύουμε τα βασικά αρχεία δεδομένων τα οποία περιλαμβάνουν αυτά του SYSTEM, SYSAUX, TEMP and UNDO tablespaces.

*Συλλογή των Listener log files:* Αυτά μπορούν να βρεθούν στην διαδρομή ORACLE_HOME/network/log. Αν δεν υπάρχουν εκεί τότε η διαδρομή στην οποία αποθηκεύονται μπορεί να βρεθεί από το αρχείο listener.ora το οποίο βρίσκεται στην διαδρομή ORACLE_HOME/network/admin/.

*b. Συλλογή πληροφοριών με την χρήση SQL ερωτημάτων.*

Το τελευταίο μέρος της έρευνας του αναλυτή, και αφού έχουν προηγηθεί όλα τα παραπάνω, είναι να συνδεθεί με τη ΒΔ. Ο αναλυτής forensics πρέπει να συνδεθεί στην βάση με δικαιώματα SYS διότι αυτό του επιτρέπει να αποκτήσει πρόσβαση σε πληροφορίες κρίσιμες για την ανάλυση της βάσης. Βέβαια οι ενέργειες που πρέπει να αποφύγει οπωσδήποτε είναι αυτές που είναι ικανές να προκαλέσουν κάποια αλλαγή στην βάση όπως INSERT, DELETE, UPDATE και ALTER.

*Συλλογή ερωτημάτων sql που έχουν ήδη γίνει προς τον server*

```
SQL> SELECT LAST_ACTIVE_TIME,
PARSING_USER_ID, SQL_TEXT FROM V$SQL
  ORDER BY LAST_ACTIVE_TIME ASC;
```

Αυτή η εντολή θα επιστρέψει τις τελευταίες εντολές SQL που εκτελέστηκαν, την ώρα που αυτό έγινε και από ποιόν. Επίσης στην παραπάνω εντολή οι επιστρεφόμενες τιμές έχουν κάποιο όριο, περίπου 2500, και μάλιστα είναι κυκλικές. Αυτό σημαίνει πως η εκτέλεση νεότερων εντολών, θα διαγράψει τις εγγραφές που δημιούργησαν παλαιότερες εντολές.

*Συλλογή του AUDIT αν αυτό δεν αποθηκευόταν στο λειτουργικό*

```
SQL> SELECT * FROM AUD$;
```

*Συλλογή πληροφοριών για τα Logons*

```
SQL> SELECT USER_ID, SESSION_ID,
SAMPLE_TIME FROM
  SYS.WRH$_ACTIVE_SESSION_HISTORY;
```

*Πλήρης λίστα χρηστών και ρόλων*

```
SQL> SELECT USER#, NAME, ASTATUS,
PASSWORD, CTIME, PTIME, LTIME FROM
  SYS.USER$ WHERE TYPE#=1;
```

*Πλήρης λίστα όλων των δικαιωμάτων του συστήματος*

```
SQL> SELECT U.NAME AS "GRANTEE", S.NAME
AS "PRIVILEGE" FROM SYS.USER$
  U, SYS.SYSAUTH$ A,
SYS.SYSTEM_PRIVILEGE_MAP S WHERE
U.USER# =
  A.GRANTEE# AND PRIVILEGE# =
S.PRIVILEGE ORDER BY U.NAME;
```

*Πλήρης λίστα όλων των αντικειμένων του συστήματος και σε ποιους αυτά ανήκουν*

```
SQL> SELECT OBJ#, OWNER#, NAME,
TYPE#, CTIME, MTIME, STIME FROM
  SYS.OBJ$ ORDER BY CTIME ASC;
```

*Λίστα με τους πίνακες που έχουν διαγραφεί*

Το παρακάτω ερώτημα δίνει τους διαγραμμένους πίνακες του συστήματος μόνο εφόσον αυτοί υπάρχουν ακόμη στο recycle bin της βάσης.

```
SQL> SELECT U.NAME, R.ORIGINAL_NAME,
R.OBJ#, R.DROPTIME, R.DROPSCN
  FROM SYS.RECYCLEBIN$ R, SYS.USER$ U
WHERE R.OWNER#=U.USER#;
```

*Έκδοση του server και επίπεδο ενημέρωσης*

```
SQL> SELECT BANNER FROM V$VERSION;
```

*Συλλογή πληροφοριών για τα triggers της βάσης*

Τα Triggers μπορούν να χρησιμοποιηθούν από τους επιτιθεμένους ως ένας μηχανισμός backdoor ή μιας λογικής βόμβας έτσι πρέπει να ελεγχθούν προσεκτικά, ειδικά αυτά που ενεργοποιούνται με το ξεκίνημα ή το κλείσιμο της βάσης ή με την σύνδεση κάποιου συγκεκριμένου χρήστη.

```
SQL> SELECT U.NAME AS "OWNER",
O.NAME AS "ENABLED_TRIGGER_NAME",
  DECODE(T.TYPE#, 0, 'BEFORE',2,
'AFTER','NOTSET') AS "WHEN" FROM
  SYS.OBJ$ O, SYS.TRIGGER$ T,
SYS.USER$ U WHERE O.OBJ#=T.OBJ# AND
  O.OWNER# = U.USER# AND ENABLED=1;
```

*Συλλογή πληροφοριών για τα views της βάσης*

Τα Views μπορούν να χρησιμοποιηθούν από τους επιτιθέμενους ώστε να κρύψουν τα ίχνη τους. Ειδική προσοχή πρέπει να δοθεί στα views που ξεκινούν με ¨DBA_¨ όπως DBA_VIEWS, DBA_USERS, DBA_ROLE_PRIVS, DBA_TAB_PRIVS και DBA_JOBS.

```
  SQL> SELECT U.NAME AS "OWNER",
O.NAME AS "VIEW", V.TEXT FROM
  SYS.VIEW$ V, SYS.OBJ$ O, SYS.USER$ U
WHERE O.OBJ#=V.OBJ# AND
  O.OWNER#=U.USER# ORDER BY U.NAME;
```

Εκτελώντας όλες τις παραπάνω ενέργειες καθώς και αυτές που αναφέρονται στην ενότητα 3.1 ο αναλυτής forensics θα έχει μια ολοκληρωμένη εικόνα τόσο του συστήματος στο οποίο έτρεχε η βάση όσο και της ίδιας της ΒΔ. Προχωρώντας λοιπόν τόσο στην ανάλυση σε βάθος των παραπάνω στοχείων όσο και στην διασταύρωση αυτών θα μπορεί να εξάγει συμπέρασμα για το τι, πως και πότε πήγε στραβά και αν τα στοιχεία επαρκούν ίσως προσδιορίσει και την ταυτότητα του εισβολέα.

## 4. Προβλήματα

Για την πλειοψηφία των ερευνητών ένα από τα πρώτα βήματα για τη συλλογή στοιχείων είναι η δημιουργία ενός ένα προς ένα αντιγράφου του φυσικού μέσου αποθήκευσης που χρησιμοποιεί ένα υπολογιστικό σύστημα. Αυτή η προσέγγιση γίνεται όλο και περισσότερο δύσκολο να εφαρμοστεί αλλά και μη πρακτική, λόγω των ακόλουθων τεχνολογικών προκλήσεων [3]:

- Η χωρητικότητα των σκληρών δίσκων στην αγορά το 2007 έχει φτάσει το 1TB (terabyte) σε κοινώς χρησιμοποιούμενες αρχιτεκτονικές υπολογιστών και 20GB σε μορφή microdrive. Η υψηλή αυτή χωρητικότητα προκαλεί πρακτικά προβλήματα: η διαδικασία της αντιγραφής γίνεται μια υπερβολικά χρονοβόρα διαδικασία ενώ η αναζήτηση στοιχείων είναι ακόμη πιο αργή.
- Τα συστήματα αρχείων που χρησιμοποιούνται στα διάφορα λειτουργικά συστήματα επιτρέπουν την απόκρυψη αρχείων από ένα κανονικό χρήστη και μπορούν να εμφανιστούν μόνο με την χρήση ειδικών εργαλείων.
- Οι τεχνολογίες Storage Virtualisation που έχουν αναπτυχθεί τα τελευταία χρόνια επιτρέπουν την αποθήκευση δεδομένων σε φυσικές τοποθεσίες διαφορετικές από εκεί που βρίσκεται το κυρίως σύστημα, όπου πιθανότατα να υπόκεινται σε άλλη νομοθεσία.
- Οι σύγχρονοι αλγόριθμοι κρυπτογράφησης είναι ήδη ισχυροί με αποτέλεσμα η διαδικασία της επίθεσης εξαντλητικής αναζήτησης να καθίσταται πρακτικά αδύνατη.

Σε άλλους τομείς της ψηφιακής σήμανσης ηλεκτρονικών υπολογιστών είναι συχνά προφανές ότι ένα έγκλημα έχει διεπραχθεί: πορνογραφικό υλικό που ανακαλύπτεται σε έναν σκληρό δίσκο ή ένα rootkit το οποίο έχει εγκατασταθεί σε ένα σύστημα αποτελούν τέτοιου είδους παραδείγματα. Στην περίπτωση μιας πιθανής παραβίασης σε μια ΒΔ ενδέχεται με την πρώτη ματιά να μην εντοπιστεί τίποτα το επιλήψιμο αλλά μετά από προσεκτική έρευνα να προκύψει το συμπέρασμα πως τα δεδομένα ενός πίνακα έχουν αλλοιωθεί, οπότε και έχει υπάρξει παραβίαση του συστήματός μας.

## 5. Πρόληψη – ανίχνευση παραβίασης (αναφορά τεχνικής επικύρωσης των log files)

Η δημιουργία και αποθήκευση log files προϋποθέτει την ικανοποίηση μιας απαραίτητης συνθήκης, αυτήν της ακεραιότητας των log files. Δυστυχώς η Oracle δεν έχει κάποιο ενδογενή μηχανισμό ούτως ώστε να εξασφαλίζει αυτή την ακεραιότητα. Επιπλέον πρόσφατες μόλις έρευνες αποδεικνύουν πως σε μια ΒΔ δεν μπορούμε να εμπιστευτούμε για την διαχείριση των log files ούτε τον log auditor [4]. Παρόλα αυτά υπάρχουν αρκετά ακαδημαϊκά project τα οποία προτείνουν λύσεις σε αυτό το πρόβλημα.

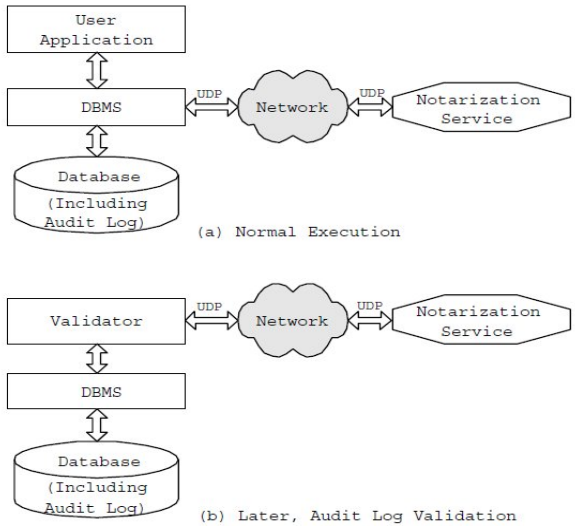
Ένα από αυτά τα project προτείνει μηχανισμούς εντός του DBMS βασισμένο σε ισχυρές κρυπτογραφικές μονόδρομες hash συναρτήσεις, που αποτρέπουν έναν εισβολέα, συμπεριλαμβανομένου ενός ελεγκτή ή ενός υπαλλήλου ή ακόμα και ενός άγνωστου bug μέσα στην DBMS από το να αλλοιώνουν σιωπηλά τα log files [4].

Η ιδέα προϋποθέτει την ύπαρξη μιας έμπιστης υπηρεσίας πιστοποίησης (Trusted Notarization Service), η οποία δίνοντας της ένα ψηφιακό έγγραφο επιστρέφει μια μοναδική τιμή και την ύπαρξη μιας έμπιστης και ανεξάρτητης υπηρεσίας επικύρωσης log files, η οποία, δίνοντας της πρόσβαση σε ένα αντίγραφο της ΒΔ, θα πιστοποιεί την εγκυρότητα των log files. Τελευταία προϋπόθεση αποτελεί η ακεραιότητα των δύο αυτών υπηρεσιών να παραμένει άθικτη ακόμη και σε περίπτωση πλήρους κατάληψης της ΒΔ.

Η βασική ιδέα είναι να αποθηκευτεί ένα πεδίο ελέγχου σε κάθε μια εγγραφή. Αυτό το πεδίο ελέγχου δεν μπορεί να υπολογιστεί άμεσα από τα στοιχεία (και τα timestamps) της εγγραφής, επειδή σε αυτή την περίπτωση ο επιτιθέμενος θα μπορούσε απλά να επαναϋπολογίσει το

πεδίο ελέγχου αφού πρώτα έχει αλλάξει την εγγραφή. Πράγματι, εάν είναι απαραίτητο θα μπορούσε να επαναλάβει όλες τις συναλλαγές, κάνοντας οποιεσδήποτε αλλαγές ήθελε στα δεδομένα ή τα timestamps. Για αυτό τον λόγο χρησιμοποιείται μια υπηρεσία πιστοποίησης που όταν της δίνεται ένα ψηφιακό έγγραφο, παρέχει ένα μοναδικό ID.

Αργότερα, κατά τη διάρκεια της επικύρωσης των log files, η υπηρεσία πιστοποίησης μπορεί να εξακριβώσει, όταν παρουσιάζεται το υποθετικά αμετάβλητο έγγραφο και το μοναδικό ID, εάν το έγγραφο έχει υποστεί αλλοίωση και αν ναι, πότε. Αυτό το επιτυγχάνει ως εξής: Σε κάθε αλλαγή εγγραφής, το DBMS λαμβάνοντας υπόψη του το timestamp, υπολογίζει μια κρυπτογραφικά ισχυρή μονόδρομη hash συνάρτηση των (νέων) δεδομένων της εγγραφής και του timestamp, και στέλνει εκείνη την hash τιμή, ως ψηφιακό έγγραφο, στη υπηρεσία πιστοποίησης, η οποία επιστρέφει ένα μοναδικό ID έχοντας προηγουμένως αποθηκεύσει το ID που επιστρέφει και την ώρα που αυτό δημιουργήθηκε. Το DBMS κρατάει αυτό το ID στην εγγραφή. Σε αυτό το σημείο κάνουμε την εξής υπόθεση: όσο ο νόμιμος χρήστης κάνει αλλαγές στην βάση, η υπηρεσία πιστοποίησης λειτουργεί πλήρως επιστρέφοντας ένα μοναδικό ID για κάθε αλλαγή που κάνει ο χρήστης, κάτι το οποίο δεν ισχύει στην περίπτωση όπου η βάση έχει καταλειφθεί από ένα μη-νόμιμο χρήστη, με την υπηρεσία πιστοποίησης να προσφέρει μόνο την λειτουργία της ανάγνωσης από αυτή (χωρίς δηλαδή την επιστροφή νέων μοναδικών ID's για τις αλλαγές που κάνει ο επιτιθέμενος – Validator Mode – Figure 1). Αυτή η υπόθεση είναι λογική αφού τονίσαμε προηγουμένως πως η υπηρεσία πιστοποίησης θεωρείται ασφαλής και ελεγχόμενη μόνο από τον νόμιμο χρήστη ακόμα και αν η βάση έχει καταληφθεί από ένα μη-νόμιμο χρήστη.




Σχήμα 1. Βασική και Validator Λειτουργία Βάσης. Πηγή [4]

Αργότερα, ο εισβολέας (Εύα) αποκτά πρόσβαση στη ΒΔ. Εάν η Εύα αλλάξει τα δεδομένα ή το timestamp, το μοναδικό ID θα είναι τώρα ασυμβίβαστο με την υπόλοιπη εγγραφή. Η Εύα δεν μπορεί να χειριστεί τα δεδομένα ή το timestamp έτσι ώστε το μοναδικό ID να παραμείνει έγκυρο, επειδή η hash συνάρτηση είναι μονόδρομη. Σημειώστε ότι αυτό ισχύει ακόμα και όταν η Εύα έχει πρόσβαση στην ίδια τη hash συνάρτηση. Η Εύα μπορεί αντί αυτού να υπολογίσει μια νέα hash τιμή για την αλλαγμένη εγγραφή, αλλά εκείνη η hash τιμή θα διαφέρει από αυτήν που προϋπήρχε. Αυτή την διαφορά θα παρατηρήσει η ανεξάρτητη υπηρεσία επικύρωσης log files, στην οποία εάν δοθεί η ΒΔ, μπορεί, για κάθε εγγραφή, αφού πρώτα εφαρμόσει την hash συνάρτηση στα δεδομένα και στο timestamp να στείλει το hash που υπολόγισε στην υπηρεσία πιστοποίησης. Ως απάντηση θα πάρει ένα ID, εάν αυτό είναι ίδιο με αυτό που υπάρχει στην εγγραφή τότε δεν είχαμε αλλοίωση. Επιπλέον συγκρίνει την χρονική στιγμή δημιουργίας του ID (η πληροφορία αυτή υπάρχει στον server της υπηρεσίας πιστοποίησης) με το timestamp της εγγραφής που είναι αποθηκευμένο στην βάση σε περίπτωση που ο επιτιθέμενος προσπαθήσει υπολογίζοντας τα δεδομένα (hash, ID) σε δική του βάση να τα αντικαταστήσει σαν πραγματικά στην βάση ¨θύμα¨. Με αυτό τον τρόπο, σε κάθε περίπτωση η υπηρεσία επικύρωσης θα έβγαζε το συμπέρασμα εάν τα δεδομένα της ΒΔ είναι έγκυρα.

Το βασικό σχήμα, αν και επιτυγχάνει τον στόχο της εξασφάλισης της ακεραιότητας παρουσιάζει μη-αποδεκτή απόδοση. Οι αλληλεπιδράσεις με τη υπηρεσία πιστοποίησης πρέπει να είναι σπάνιες, επειδή τέτοιες αλληλεπιδράσεις κοστίζουν, απαιτώντας μη-τοπικές μεταδόσεις δεδομένων μέσω δικτύων. Επιπλέον, το overhead αυξάνεται υπερβολικά, δεδομένου ότι ένα μοναδικό ID πρέπει να αποθηκευτεί σε κάθε εγγραφή. Οφείλουμε να λάβουμε υπόψη ότι κάθε συναλλαγή μπορεί να μεταβάλει εκατομμύρια εγγραφών καθιστώντας αυτή την τεχνική μη εφαρμόσιμη. Για αυτό το λόγο οι κύριες τεχνικές που υιοθετούνται για την βελτίωση της απόδοσης είναι το ευκαιριακό hashing και το linked hashing [4].

### *a. Ευκαιριακό hashing (Opportunistic Hashing)*

Το πρώτο βήμα είναι να μειωθούν οι αλληλεπιδράσεις με την υπηρεσία πιστοποίησης σε μία ανά συναλλαγή, παρά σε μία ανά αλλαγή εγγραφής. Για αυτό τον λόγο κάνουμε hash σε όλες τις εγγραφές που μεταβλήθηκαν υπολογίζοντας μια τιμή hash των 20 byte η οποία στέλνεται στη υπηρεσία πιστοποίησης. Το μοναδικό ID που επιστρέφεται αποθηκεύεται σε έναν ξεχωριστό πίνακα πιστοποίησης, ο οποίος περιέχει μια εγγραφή για κάθε συναλλαγή. Ο ελεγκτής πιστοποίησης των log files σαρώνει όλες τις σελίδες των ελεγχόμενων πινάκων, διατηρώντας μια τρέχουσα hash τιμή για κάθε συναλλαγή,

με κάθε συναλλαγή να ταυτοποιείται από ένα χρόνο εκκίνησης και τερματισμού. Αφότου έχει σαρωθεί η ΒΔ, ο ελεγκτής έχει τη hash τιμή για κάθε συναλλαγή. Μπορεί έπειτα να ελέγξει το πίνακα πιστοποίησης για να δει ποια συναλλαγή αλλοιώθηκε.

*b. Linked hashing*

Η αλληλεπίδραση με την υπηρεσία πιστοποίησης σε κάθε συναλλαγή κοστίζει όμως και αυτή αρκετά, ειδικά στα σύγχρονα υψηλής απόδοσης συστήματα, τα οποία μπορούν να ολοκληρώσουν χιλιάδες συναλλαγές το δευτερόλεπτο.

Μια ελκυστική λύση είναι να χρησιμοποιηθούν τιμές μερικής επικύρωσης αποτελέσματος για να συνδέσουν τις συναλλαγές. Με την δημιουργία της ΒΔ παίρνουμε την hash τιμή του σχήματος της βάσης και του timestamp, όπου την τιμή αυτή την αποθηκεύουμε στο πίνακα πιστοποίησης. Κατόπιν, για κάθε συναλλαγή, παίρνουμε την hash τιμή της συναλλαγής όπως και στην περίπτωση του ευκαιριακού hashing. Στην επικύρωση συναλλαγής (commit) ξαναπαίρνουμε την τιμή hash της τρέχουσας συναλλαγής και της προηγούμενης τιμής για να πάρουμε μια καινούργια τιμή.

Περιοδικά, έστω τα μεσάνυχτα, η hash τιμή της πιο πρόσφατης συναλλαγής, στέλνεται στη υπηρεσία πιστοποίησης επιστρέφοντας ένα μοναδικό ID. Παίρνουμε την πιο πρόσφατη hash τιμή μαζί με το μοναδικό ID της, για να υπολογίσουμε μια νέα hash τιμή που χρησιμοποιείται για να συνδέσει τις επόμενες συναλλαγές. Για να ελέγξουμε την ισχύ, επαναλαμβάνουμε το hashing, με χρονική σειρά ανά συναλλαγή, ελέγχοντας τις τιμές που πήραμε για τη συναλλαγή τα μεσάνυχτα με την υπηρεσία πιστοποίησης.

## 6. Database forensics και προσωπικό απόρρητο

Οι ΒΔ που κρατούν ένα ιστορικό αρχείο δραστηριοτήτων προσφέρουν ένα σημαντικό όφελος: τα αποθηκευμένα log files μπορούν να αναλυθούν ώστε να ανιχνευτούν πιθανές παραβιάσεις. Η διατήρηση του ιστορικού όμως μπορεί επίσης να αποτελέσει απειλή για την ιδιωτικότητα. Οι σχεδιαστές συστημάτων πρέπει να εξισορροπούν προσεκτικά την ανάγκη για ιδιωτικότητα και έλεγχο του συστήματος με το να εξετάζουν το πώς και πότε ένα log file θα δημιουργείται και θα διατηρείται από το σύστημα και ποιος θα είναι σε θέση να το ανακτήσει και να αναλύσει [5].

Συνήθως, ο αναλυτής forensics είναι ένας επαγγελματίας ο οποίος έχει νομίμως πάρει την άδεια να εξετάσει ένα υπολογιστικό σύστημα αποκτώντας απεριόριστη πρόσβαση σε αυτό. Δυστυχώς όμως η forensics ανάλυση ενός υπολογιστή μπορεί να εκτελεσθεί από μια σειρά πιθανών αντιπάλων: χάκερ, νόμιμους χρήστες με κατάλληλα δικαιώματα, ή από οποιονδήποτε έχει φυσική πρόσβαση στο υλικό μέσω της κλοπής ή της απώλειας. Ενώ η κρυπτογράφηση στοιχείων μπορεί να βοηθήσει στο να προστατεύσει κάποια ενδεχόμενη αναρμόδια πρόσβαση, αυτό μπορεί να μην αποτρέψει κάποιον νόμιμο χρήστη από το να αποκτήσει πρόσβαση.

## 7. Προετοιμασία συστήματος

Μια σημαντική παράμετρος η οποία αφορά την αποτελεσματικότητα της διαδικασίας της forensics ανάλυσης είναι η προετοιμασία του συστήματος. Για μια επιτυχημένη forensics ανάλυση απαιτείται η κατάλληλη προεργασία με κύριο τμήμα αυτής το σωστό στήσιμο από πλευράς ασφάλειας τόσο του συστήματος που θα υποδεχτεί την βάση όσο και της ίδιας της βάσης. Έτσι μπορεί κανείς στο διαδίκτυο να εντοπίσει checklists ρυθμίσεων, από μεγάλους οργανισμούς που ασχολούνται με την ασφάλεια υπολογιστικών συστημάτων, για την διασφάλιση ενός στάνταρ επιπέδου ασφάλειας. Μια πλήρης λίστα αποτελεί αυτή της βιβλιογραφικής αναφοράς [6] η οποία έχει εκδοθεί από τον διεθνή οργανισμό SANS και τα κύρια σημεία στα οποία αναφέρεται είναι τα εξής:

1. Σχεδιασμός και Εκτίμηση Ρίσκου
2. Θέματα Ασφάλειας που αφορούν τον ¨Οικοδεσπότη¨ Λειτουργικό Σύστημα
3. Σύστημα Πιστοποίησης της Oracle
4. Σύστημα Ελέγχου Πρόσβασης της Oracle
5. Auditing
6. Δίκτυο
7. Διαθεσιμότητα/ backup / Ανάκτηση
8. Ανάπτυξη Εφαρμογής
9. Servers Παροχής Υπηρεσιών

## 8. Συμπεράσματα

Τα συμπεράσματα που προέκυψαν στα πλαίσια της παρούσας εργασίας είναι:

1) Η επέκταση της διαδικασίας του logging γεγονότων σε μεγάλο βαθμό μπορεί να καταλήξει σε ρήγμα ασφάλειας αφού κάποιος που έχει πρόσβαση στο σύστημα αρχείων ενός συστήματος μπορεί από το να δει απλά τα διάφορα passwords της ΒΔ μέχρι και να καταλάβει αν η ΒΔ είναι στην παραγωγή, αν δηλαδή χρησιμοποιείται κανονικά και περιέχει πραγματικά δεδομένα ή απλά χρησιμοποιείται για δοκιμαστικούς σκοπούς.

2) Δυστυχώς μέχρι και σήμερα δεν έχει καθοριστεί κάποιο πρότυπο όσο αφορά τη διαδικασία ανάλυσης, το οποίο να είναι αποδεκτό από όλους τους ερευνητές.

3) Οι παραβιάσεις συνήθως δεν ανακοινώνονται – δημοσιεύονται για λόγους δυσφήμισης. Αυτό ισχύει ιδιαίτερα στις ΒΔ λόγω της σοβαρότητας και της κρισιμότητας των πληροφοριών που περιέχουν.

4) Το αντικείμενο της forensics ανάλυσης περιλαμβάνει συνδυασμό θεωρητικής και τεχνικής γνώσης δίνοντας βέβαια τεράστια έμφαση στο δεύτερο αφού δεν αρκεί να έχει κανείς γνώση για το πως λειτουργεί ένα σύστημα αλλά και να μπορεί να αντλήσει στοιχεία από αυτό εφαρμόζοντας κατάλληλες μεθόδους. Επιπλέον υπάρχουν περιπτώσεις όπου το μεγαλύτερο τμήμα της ανάλυσης γίνεται σε χαμηλό επίπεδο κάτι που συνεπάγεται και βαθειά γνώση τρόπου λειτουργίας αλλά και μηχανισμών χαμηλού επιπέδου.

5) Σημαντικότερος εχθρός ο χρόνος και οι αποφάσεις. Ο χρόνος είναι ο πρώτος και πιο σημαντικός παράγοντας με τον οποίο έρχεται αντιμέτωπος ο αναλυτής forensics. Όσο πιο γρήγορα κληθεί να αντιμετωπίσει ένα πρόβλημα, τόσο πιο γρήγορη θα είναι η εξαγωγή συμπεράσματος για το τι συνέβη σε ένα σύστημα, αφού με το πέρασμα του χρόνου επέρχεται αλλοίωση δεδομένων όχι απαραιτήτως από φυσική πρόσβαση ή φυσικά αίτια αλλά πχ. από ρουτίνες του ίδιου του λειτουργικού συστήματος οι οποίες δρουν πάνω σε αρχεία προκαλώντας αλλοιώσεις τόσο στα μεταδεδομένα όσο και στα περιεχόμενα τους. Επίσης καθόλη την διάρκεια της ανάλυσης ο αναλυτής καλείται να πάρει κρίσιμες αποφάσεις οι οποίες στηρίζονται συνήθως σε πιθανά διλήμματα (trade-offs). Ένα κλασσικό δίλημμα είναι αυτό της αποσύνδεσης της βάσης ή όχι από το δίκτυο μετά την διαπίστωση ότι έχει παραβιαστεί, θέμα το οποίο αναλύθηκε στην ενότητα 3.

6) Τα εργαλεία δεν κατασκευάζονται με γνώμονα τις ιδιαιτερότητες της forensics ανάλυσης και αν κατασκευαστούν μπορεί να χρησιμοποιηθούν από χάκερς αφού βασική απαίτηση αυτών των εργαλείων θα ήταν να μην αφήνουν κανενός είδους αποτύπωμα στο σύστημα. Βέβαια αυτός είναι ένας τομέας όπου χωρά αρκετή έρευνα η οποία μπορεί να οδηγήσει και στις κατάλληλες λύσεις.

Τέλος όπως αναφέρθηκε και στην ενότητα 7 η σωστή προετοιμασία τόσο του συστήματος όσο και της ΒΔ αποτελεί καθοριστικό παράγοντα για την αποτελεσματικότητα της forensics ανάλυσης.

## 9. Αναφορές


[1] D. Litchfield, *Oracle Forensics Part 4: Live Response*, http://www.databasesecurity.com/oracle-forensics.htm, 2007.

[2] D. Litchfield, *Oracle Forensics In A Nutshell*, http://www.databasesecurity.com/oracle-forensics.htm, 2007.

[3] E. Huebner – D. Bem – O. Bem, "Computer Forensics – Past, Present And Future", *JIST* 5(3), 2008, pp 43-59.

[4] R T. Snodgrass, S. S. Yao and C. Collberg, "Tamper Detection in Audit Logs", *Proceedings 30th VLDB Conf.*, 2004, pp 504-515.

[5] P. Stahlberg - G. Miklau – B. N. Levine, "Threats to Privacy in the Forensic Analysis of Database Systems", *SIGMOD*, 2007, pp 91-102.

[6] http://www.sans.org/score/checklists/Oracle_Database_Checklist.pdf - Oracle Checklist (last date accessed – 07/03/2009).