UNIVERSITATEA BABEŞ-BOLYAI CLUJ-NAPOCA

FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ

SPECIALIZAREA INFORMATICĂ

LUCRARE DE LICENȚĂ

ANALIZA AUTOMATĂ A COMPORTAMENTULUI DINAMIC AL VIRUŞILOR

Conducători ştiinţifici
C. d. asociat TOPAN VLAD-IOAN
CONF. DR. BOIAN RAREŞ

Absolvent
OANCEA ALIN ANDREI

2019

Cuprins

1. Introducere

Principalele obiective ale acestei lucrări sunt prezentarea conceptelor de bază despre securitatea cibernetică, riscurile infecţiei calculatoarelor cu malware, tehnicile de prevenţie care există în domeniul securităţii, dar şi metodele prin specialiştii au reuşit să analizele fişierele maliţioase.

Aplicaţia dezvoltată foloseşte tehnicile de analiză dinamică pentru a monitoriza activitatea maliţioasă din interiorul unui mediu controlat, configurat special pentru a fi folosit în acest scop. Beneficiile pe care aplicaţia le oferă sunt monitorizarea schimbărilor pe care fişierul maliţios le efectuează asupra maşinii ţintă atât la nivel de regiştrii, dar şi la nivelul fişierelor de pe disc şi posibilitatea de execuţie a unui număr mare de malware, într-un timp scurt şi afişarea rezultatelor extrase într-o interfaţă web.

În primul capitol se prezintă pe scurt istoria securităţii informaţiilor şi modelul care stă la baza securităţii informaţiilor, triunghiul C.I.A, dar şi modelul extins al acestuia. Un al aspect important discutat este securitatea cibernetică şi ameninţările prezente: cyberterrorism, cyberwarfare şi cyberespionage. În finalul capitolului se discută despre importanţa securităţii cibernetice atât în viaţa persoanelor, cât şi în interiorul guvernelor şi companiilor.

În capitolul 2, **Malware**, se descrie istoria malware-ului, cum s-a ajuns la producerea acestora, precum şi primul virus din istorie. Mai apoi se detaliază tipurile de malware existente, diferenţele şi asemănările pe care acestea le au, dar şi caracteristicile de bază de pe care le posedă, modurile de răspândire şi intenţiile pe care le au când aceştia ajung pe calculatoarele victimelor. Apoi se descriu câteva soluţii ce protejare a calculatoarele de o infecţie cu malware care poate produce daune semnificative.

Capitolul 3, **Analiza de malware**, discută despre modalităţile principale prin care un malware poate fi analizat: analiza statică, dinamică şi automată. În sub-capitolele de analiză statică şi dinamică sunt explicate avantajele şi dezavantajele fiecărei metode şi cum s-a ajuns la nevoia utilizării inteligenţei artificiale în analiza automate.

Capitorul 4, **State of the Art**, prezintă diverse abordări de analiză pe care aplicaţiile specializate în analiza de malware le-au implementat. Se prezintă atât aplicaţii de analiză

open-source, aplicaţii care folosesc medii de testare proprii sau de la alte companii pentru analiza malware-ului şi aplicaţii care implementează soluţii antivirus cu care scanează fişierele maliţioase.

În capitolul 5, **Analiza automată a comportamentului dinamic al viruşilor**, este descrisă în amănunt abordarea aplicaţiei pentru analiza dinamică a malware-ului. Această aplicaţie face posibilă analiza unui fişier maliţios, monitorizarea comportamentului malware-ului în interiorul unei maşini virtuale create cu ajutorul aplicaţiei de virtualizare VirtualBox şi extragerea rezultatelor relevante unei analize dinamice de malware. Tot în acest capitol prezentăm modul în care este recomandată crearea unei maşini de analiză precum şi aplicaţiile dezvoltate pentru monitorizarea unui mediu de testare.

1.1. Securitatea informaţiilor

Securitatea informaţiilor este procesul care implică protecţia informaţiilor prin diminuarea factorilor de risc. Aceasta implică prevenţia sau reducerea probabilităţii accesării, utilizării, dezvăluirii, distrugerii, coruperii, modificării, examinării, înregistrării de către persoane sau sisteme neautorizate. Obiectivele generale ale securităţii informaţiilor sunt păstrarea confidenţialităţii, integrităţii şi disponibilităţii informaţiilor şi a resurselor informaţionale. Protecţia acestor obiective a devenit o necesitate pentru fluxului financiar, profitabilităţii şi cerinţele legale, derivată din dreptul de proprietate a informaţiilor[18].

Valoarea informaţiilor provine din caracteristicile pe care acestea le posedă. Când o caracteristică a acestei informaţii se schimbă, valoarea acelei informaţii poate să crească sau mai frecvent, să scadă. Unele caracteristici afectează valoarea informaţiei pentru un utilizator mai mult decât altele, depinzând de circumstanţe. De exemplu, actualitatea unei informaţii poate să fie un factor critic, deoarece valoarea informaţiei poate să scadă considerabil atunci când aceasta este furnizată târziu. Totuşi specialiştii din domeniul securităţii informaţiilor şi utilizatorii finali împart un acord privind caracteristicile informaţiei - pot apărea tensiuni atunci când necesitatea securizării informaţiilor de ameninţări intervine în nevoia utilizatorului de accesarea acestor informaţii. De exemplu, utilizatorii pot percepe o întârziere de 0.1 secunde în procesarea datelor ca fiind un disconfort. Pe de altă parte, specialiştii din domeniul securităţii pot percepe întârzierea de 0.1 secunde ca fiind una necesară care poate permite execuţia unei sarcini importante, precum criptarea datelor[24].

Triada securităţii, triada/triunghiul/modelul C.I.A, este un model remarcabil pentru dezvoltarea mecanismelor de securitate, care distinge trei caracteristici ale informaţiei din punct de vedere al securităţii, şi anume: confidenţialitatea (eng. confidentiality - C) datelor, integritatea (eng. integrity - I) datelor şi disponibilitatea (eng. availability - A) datelor, aşa cum este prezentat şi în figura 1.1.1. O breşă de securitate în oricare din aceste zone poate cauza probleme daune mari asupra sistemelor[11]. În următoarele linii o să prezentăm pe scurt fiecare din caracteristicile C.I.A:

- confidenţialitatea: se referă la limitarea accesului la informaţie doar utilizatorilor autorizaţi şi prevenirea accesului utilizatorilor sau sistemelor neautorizate[6];
- integritatea: se referă la corectitudinea informaţiilor stocate. Informaţia este originală (de la creator), corectă (nu falsificată sau fabricată) şi nemodificată de către persoane neautorizate (valoarea este neschimbată)[6];
- disponibilitatea: se referă la proprietatea informaţiilor de a fi accesibile, furnizarea informaţiilor oricând aceasta este cerută. Disponibilitatea informaţiei înseamnă că protocoalele şi serviciile trebuie să fie operaţionale chiar şi în prezenţa defecţiunilor de origine maliţioasă[6].

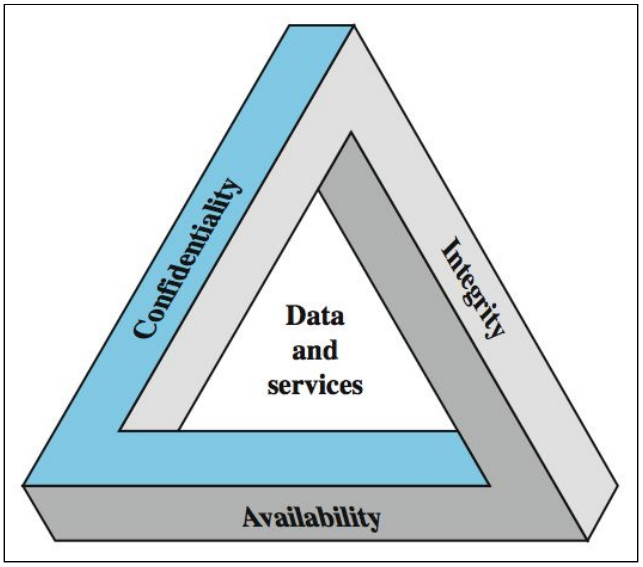


Figura 1.1.1. Triada securităţii sau triada C.I.A[1]

Mai jos o să definim fiecare caracteristică critică a informaţiei, acestea fiind considerate şi triunghiul extins C.I.A[24]:

[1] http://mattjjohansen.blogspot.com/2009/01/computer-security-week-1.html

- disponibilitatea - permite utilizatorilor autorizaţi (oameni sau calculatoare) să acceseze informaţiile fără interferenţe sau blocaje şi să o primească într-un format solicitat. Să considerăm de exemplu o bibliotecă care necesită identificare înaintea intrării. Bibliotecarii protejează conţinutul bibliotecii astfel că informaţiile sunt disponibile doar persoanelor autorizate. Bibliotecarul va trebui să identifice clienţii înainte ca aceştia să aibă acces la cărţile bibliotecii. Odată identificaţi, clienţii se aşteaptă să găsească informaţiile de care aceştia au nevoie[24];
- acurateţea - informaţia are acurateţe atunci când aceasta este lipsită de greşeli sau erori. Să considerăm verificarea soldului dintr-un cont bancar, de exemplu. O persoană care îşi verifică soldul din cont se aşteaptă ca informaţia prezentată este o reprezentare reală a situaţiei a situaţiei financiare. În schimb, dacă sistemul afişează o valoare de două ori mai mare din cauza unui defect software, această informaţie ar fi greşită. O valoare inexactă a soldului contului bancar poate cauza greşeli precum respingerea unei plăţi[24];
- autenticitatea - este calitatea sau condiţia ca informaţia să fie veritabilă sau originală. Informaţia este autentică când este în aceeaşi stare ca cea în care a fost creată, situată, stocată sau transferată. Să considerăm câteva presupuneri comune despre email. Când un email este primit, poţi să presupui că o persoană sau un grup specific a creat şi transmis email-ul în cauză - asumi că ştii originea sa. Această presupunere nu este mereu adevărată. Actul de a trimite mesaje email cu unele câmpuri modificate, denumit eng. email spoofing, este o mare problemă actuală pentru mulţi oameni, deoarece câmpul cel mai des modificat într-un email este adresa persoanei care a trimis email-ul. Modificând adresa expeditorului poate păcăli destinatarii email-ului să creadă că mesajele transmise sunt autentice, astfel determinând deschiderea email-ului care în caz contrar s-ar putea să nu se fi întâmplat[24];
- confidenţialitatea - informaţia este confidenţială când aceasta este protejată de dezvăluirea sau expunerea către sisteme sau persoane neautorizate. Confidenţialitatea presupune că numai utilizatorii cu privilegii pot accesa informaţia. Atunci când sisteme sau persoane neautorizate pot vizualiza informaţia, confidenţialitatea este încălcată. Pentru protejarea confidenţialităţii informaţiei, se pot utiliza mai multe măsuri, precum: clasificarea informaţiei, securizarea documentelor stocate, aplicarea politicilor de securitate generală, educarea administratorilor de informaţii şi a

utilizatorilor. Valoarea confidenţialităţii informaţiei este crescută când este vorba de informaţii despre angajaţi, clienţi sau pacienţi. Persoanele care tranzacţionează cu o organizaţie se aşteaptă ca informaţiile lor personale să rămână confidenţiale, chiar dacă organizaţia este o agenţie guvernamentală sau o companie. Problemele apar atunci când organizaţiile dezvăluie informaţiile confidenţiale. Dezvăluirea poate să fie intenţionată, dar poate să fie şi din greşeală - de exemplu, atunci când informaţii confidenţiale sunt trimise prin email către o persoană dinafara organizaţiei şi nu cuiva din interiorul acesteia. Alte exemple de încălcări ale confidenţialităţii sunt atunci când un angajat aruncă un document cu informaţii critice fără ca acesta să fie distrus, sau când un atacator care pătrunde cu succes într-o bază de date a unei organizaţii şi sustrage informaţii confidenţiale despre clienţi, precum nume, adrese sau numerele cardurilor de credit[24];

- integritatea - informaţia are integritate atunci când aceasta este întreagă, completă şi nu prezintă stricăciuni. Integritatea informaţiei este ameninţată când este expusă coruperii, deteriorării, distrugerii sau a altor stricăciuni ale stării autentice. Coruperea informaţiei poate avea loc când informaţia este stocată sau transferată. Multe programe malware sunt proiectate cu scopul de a corupe date. O metodă prin care se poate verifica integritatea informaţiei este eng. „file hashing” (trad. „hashing” pe un fişier), unde un fişier este citit de un algoritm special care calculează un număr denumit valoare de hash. Dacă un calculator execută acelaşi algoritm pe acelaşi fişier şi obţine o valoare de hash diferită faţă de valoarea înregistrată pentru fişierul respectiv, atunci se poate spune că fişierul a fost compromis şi că integritatea informaţiei este pierdută[24];
- utilitatea - este calitatea informaţiei de a fi folosită pentru un scop final. În alte cuvinte, informaţia are valoare atunci când poate servi unui scop. Dacă informaţia este disponibilă, dar formatul acesteia nu îi este de folos utilizatorului, aceasta nu este utilă. De exemplu, datele unui recensământ pot deveni greu de interpretat pentru un cetăţean; totuşi, pentru un politician, aceleaşi date pot dezvălui informaţii despre rezidenţii unei zone, pentru rasa, sexul şi vârsta. Aceste informaţii pot ajuta, de exemplu, la strategia următoarei campanii a politicianului[24];
- deţinerea - este calitatea informaţiei de a fi deţinute sau controlate. Informaţia este în posesia cuiva dacă au fost obţinute independent de format sau alte caracteristici. Pe

când o încălcare a confidenţialităţii mereu rezultă într-o încălcare a posesiei, o încălcare a posesiei nu conduce mereu la o încălcare a confidenţialităţii. Spre exemplu, să presupunem că o companie stochează datele importante ale clienţilor utilizând un sistem criptat. Un angajat care a demisionat decide să copieze aceste date şi să le vândă competiţiei. Îndepărtarea datelor din mediul securizat este o încălcare a posesiei. Cu toate acestea, pentru că datele sunt criptate, nici fostul angajat, nici altcineva nu poate citi aceste date fără metoda adecvată de decriptare; astfel neexistând o încălcare a confidenţialităţii[24].

Conceptul de securitatea informaţiilor, practicile şi procedurile sunt într-o continuă dezvoltare pentru a satisface desfăşurarea fluid al unei afaceri. Cu toate acestea, implementările simple ale soluţiilor de securitatea informaţiilor de către organizaţii sunt insuficiente. Lumea din afara organizaţiilor a devenit progresiv orientată spre informaţii, ca rezultat, principiile securităţii informaţiilor au devenit aplicabile şi informaţiile utilizate în context personal. În prezent, utilizatorii Internetului sunt nevoiți să aibă cel puțin un nivel de bază despre informările din domeniul securității cibernetice pentru a-și putea efectua activitățile zilnice într-un mediu securizat[18].

1.2. Securitatea cibernetică

Securitatea cibernetică este o colecţie de unelte, metode, concepte de securitate, măsuri de securitate, instrucţiuni, abordări de gestionare a riscurilor, acţiuni, pregătiri, bune practici şi tehnologii care pot fi folosite pentru protejarea spaţiului cibernetic şi organizaţiilor. Organizaţiile şi bunurile utilizatorilor includ dispozitive conectate, personal, infrastructură, aplicaţii, servicii, sisteme de telecomunicaţie şi totalitatea informaţiilor transmise sau stocate în spaţiul cibernetic. Securitatea cibernetică încearcă să protejeze informaţiile dintr-o organizaţie şi a bunurilor utilizatorilor împotriva riscurilor de securitate din spaţiul cibernetic. Obiectivele generale ale securităţii sunt[19]:

- confidentiality (trad. confidenţialitatea)
- integrity (trad. integritatea)
- availability (trad. disponibilitatea)

Se poate observa astfel că cele trei obiective enumerate mai sus sunt foarte similare cu cele ale securităţii informaţiilor. Standardul internaţional, ISO/IEC 27002 (2005), spune că securitatea informaţiilor este protejarea confidenţialităţii, integrităţii şi disponibilităţii

informaţiilor. În contextul ISO/IEC 27002 (2005), informaţia poate fi imprimată sau scrisă pe hârtie, stocată digital, transmisă prin poştă sau prin mijloace electronice, transmisă prin conversaţie şi aşa mai departe[19].

În lucrarea „Principles of information security”[24] scrisă de Whitman şi Mattord, cei doi definesc securitatea informaţiei ca „protecţia informaţiei şi a elementelor sale critice, incluzând sistemele care sunt implicate în utilizarea, stocarea şi transmiterea acelei informaţii”. Autorii identifică câteva caracteristici critice ale informaţiei care oferă valoare în organizaţii. Aceste caracteristici includ confidenţialitatea, integritatea şi disponibilitatea informaţiei, menţionate de asemenea în ISO/IEC 27002 (2005), dar nu sunt limitate doar la aceste caracteristici. Conform autorilor Whitman şi Mattord, asigurarea confidenţialităţii, integrităţii şi disponibilităţii informaţiei, cunoscută şi ca triunghiul CIA (eng. Confidentiality, Integrity, Availability), a fost un standard în industrie[19].

Atacurile cibernetice pot fi de forma viruşilor, *ransomware-ului*, *phishingu-lui*, etc. Atacatorii cibernetici au dezvoltat şi automatizat tehnici sofisticate de atac, motivaţi în principal de rentabilitatea tot mai mare a atacurilor. Ca rezultat, operaţiunile şi strategiile de securitate împotriva acestor atacatori fac faţă cu greu cerinţelor actuale, mai ales în sistemele guvernamentale şi companii unde adesea atacurile cibernetice au ca ţintă secrete de stat sau bunuri naţionale. Cele mai întâlnite ameninţări sunt următoarele[16]:

- *cyberterrorism* (trad. terorism cibernetic): constă în utilizarea reţelei Internet pentru dirijarea de acte de violenţă care pot rezulta în ameninţarea, pierderea vieţii sau vătămări corporale semnificative, cu scopul de a atinge obiective sau ideologii politice obţinute prin ameninţări sau intimidări. Un act de terorism pe Internet sunt considerate şi activităţile precum întreruperi masive a reţelelor de calculatoare, în special al calculatoarelor personale conectate la Internet cu unelte precum viruşi, *worms*, *phishing* şi alte software-uri maliţioase. Terorismul cibernetic este un termen controversat. Unii autori folosesc o definiţie foarte restrânsă, care se referă la declanşarea de către cunoscute grupuri teroriste a atacurilor de distrugere a informaţiilor, cu scopul de a alarmă, panică sau distrugeri fizice[30].
- *cyberwarfare* (trad. război cibernetic), este un termen general care descrie utilizarea forţei tehnologice în interiorul spaţiului cibernetic. *Cyberwarfare* nu implică mărimea, prelungirea sau violenţa, termeni care sunt de obicei asociaţi cu războiul. Există o dezbatere între experţi în legătură cu definiţia războiului cibernetic şi dacă

acesta ar exista. Câteva definiţii au fost propuse pentru războiul cibernetic, dar nici una dintre acestea nu a fost adoptată internaţional. Richard A. Clarke defineşte *cyberwarfare* ca „acţiunile unei naţiuni de pătrundere în calculatoarele sau reţelele altei naţiuni cu scopul de a cauza daune sau distrugeri". Martin Libicki defineşte două tipuri de *cyberwarfare*: strategic şi operaţional. Războiul cibernetic strategic fiind „o campanie de atac cibernetic pe care o entitate o desfăşoară asupra alteia", iar cel operaţional „implică utilizarea atacurilor cibernetice împotriva armatei celeilalte naţiuni cu scopul unui război fizic"[31].

- *cyberespionage* (trad. spionajul cibernetic), este un act sau o practică de obţinere a informaţiilor secrete fără permisiunea şi cunoştinţa deţinătorului informaţiilor de la indivizi, competitori, rivali, grupuri, guverne şi inamici pentru avantaj personal, economic, politic sau militar, utilizând metode Internet, reţele sau calculatoare personale prin folosirea de servere *proxy*, tehnici de spargere şi software maliţios ca *spyware* şi cal troian. Spionajul cibernetic de obicei implică utilizarea unor astfel de secrete şi informaţii clasificate sau controlul de computere personale sau al unei întregi reţele pentru un avantaj strategic şi pentru activităţi psihologice, politice şi fizice, precum şi sabotaj[29]. De exemplu, un APT - eng. *advanced persistent threat* (trad. ameninţare persistentă avansată), este un atac ascuns asupra calculatoarelor în care o persoană sau un grup obţine acces neautorizat asupra unei reţele şi rămâne nedetectat pentru o perioadă lungă de timp. APT se referă de obicei la un grup, precum un guvern, cu capabilitatea şi scopul de a ţinti, persistent şi în mod eficient o entitate specifică[25].

1.3. Importanţa securităţii

Securitatea cibernetică este considerată o parte importantă al indivizilor şi familiilor, precum şi a organizaţiilor, guvernelor, instituţiilor educaţionale şi a business-urilor. Este esenţial pentru familie ca părinţii să protejeze copiii şi membrii familiei de frauda online. În termeni de securitate financiară, securizarea informaţiilor financiare este crucială deoarece statutul financiar poate sa fie afectat. Internetul este important şi benefic pentru facultăţi, studenţi, angajaţi şi instituţii educaţionale, oferind multe oportunităţi de învăţare cu tot atâtea riscuri online. Este o necesitate vitală pentru utilizatorii de internet să înţeleagă cum să se protejeze de frauda online şi furtul de identitate. Învăţarea adecvată despre comportamentul

online şi sistemele de protecţie pot rezulta în reducerea de vulnerabilităţi şi într-un mediu online mai sigur. Companiile mici şi mijlocii experimentează numeroase provocări de securitate din cauza resurselor limitate şi a cunoştinţelor despre securitatea cibernetică la fel de limitate. Expansiunea rapidă a tehnologiilor creează şi face de asemenea securitatea cibernetică mai complicată pentru că nu deţinem soluţii pentru problemele în cauză. Cu toate că luptăm şi prezentăm diverse arhitecturi sau tehnologii pentru a ne proteja reţelele şi informaţiile, din păcate aceste măsuri oferă protecţie pentru o scurtă perioadă de timp. Totuşi, o mai bună înţelegere a securităţii şi a strategiilor adecvate ne pot ajuta să protejăm proprietatea intelectuală, secrete şi să reducem pierderile financiare[17].

Securitatea cibernetică este importantă pentru că organizaţiile precum guverne, armata, companii, financiare şi medicale colectează, procesează şi stochează imense cantităţi de date valoroase pe calculatoare şi alte dispozitive. O parte semnificativă de date pot fi informaţii confidenţiale, chiar dacă aceste date pot fi de natură intelectuală, financiară sau personală, consecinţele negative pe care le poate avea accesul neautorizat pot fi devastatoare. Organizaţiile transfera date importante prin internet; securitatea cibernetică descrie metodele de protecţie a acestor informaţii şi a dispozitivelor folosite pentru procesarea şi stocarea acestor date. Din cauza creşterii atât în volum cât şi în complexitate al atacurilor cibernetice, companiile şi organizaţiile, în special cele care protejează informaţii de securitate naţională, sănătate sau date financiare, au fost introduse directive care să protejeze aceste date. Specialişti din domeniul securităţii informaţiilor au avertizat că atacurile cibernetice şi spionajul digital sunt ameninţările de vârf asupra securităţii naţionale a ţărilor dezvoltate[13].

Provocarea cea mai dificilă în securitatea cibernetică este evoluţia constantă a riscurilor cibernetice. Organizaţiile şi-au concentrat majoritatea resurselor pentru securitate cu scopul de a proteja doar sistemele cruciale şi combaterea ameninţărilor cunoscute. Această abordare nu mai constituie o soluţie, deoarece ameninţările evoluează cu o viteză cu care organizaţiile nu pot ţine pasul[13].

2. Malware

Malware sau software maliţios este un software utilizat sau creat de către atacatori cu scopul de a împiedica funcţionarea normală a calculatorului sau pentru a obţine acces la sisteme de calculatoare private. Malware este termenul folosit în general pentru a referi forme ostile sau intruzive de bucăţi software. Malware-ul include *viruses* (trad. viruşi), *ransomware*, *worms* (trad. viermi), *trojan horses* (trad, cal troian), *rootkits*, *keyloggers*, *dialers*, *spyware*, *adware* şi alte tipuri de programe maliţioase; majoritatea ameninţărilor cu malware sunt *worms* sau *trojans*[15]. Malware-ul are o intenţie maliţioasă, acţionând împotriva intereselor utilizatorului, de aceea nu include software care cauzează daune neintenţionat datorită unor defecte de proiectare. Programele oficiale oferite de companii pot fi considerate malware dacă acestea acţionează împotriva intereselor utilizatorului. De exemplu, compania Sony vindea CD-uri cu muzică care instalau un software maliţios pe calculatorul cumpărătorilor cu intenţia de a preveni copierea ilicită, dar care de asemenea raporta obiceiurile ascultătorilor şi de asemenea crea vulnerabilităţi de securitate[32].

2.1. Istorie

O scurtă privire asupra istoriei ne arată că intenţiile maliţioase datează încă de pe vremea primelor calculatoare. Ideea de virus apare din 1949, când John von Neumann a scris „Theory and Organization of Complicated Automata”[9] - lucrare care sugerează modul în care un program software se poate auto-reproduce. În anii 1950, angajaţii companiei „Bell Labs”, inspiraţi de ideea lui Neumann, creează un joc numit „Core Wars”. În acest joc, programatorii eliberează „organisme” software care concurează pentru control asupra calculatorului[1].

Termenul de „virus” este introdus abia în anul 1986, de către Fred Cohen în lucrarea lui de doctorat - „Computer Viruses - Theory and Experiments”[5]. Cohen a definit „virusul” ca fiind un program care poate infecta alte programe prin modificarea lor pentru a include, o nouă versiune a sa. Estimările arată că infecţiile cu malware au ajuns la o treime din computerele globale. „Creeper Worm” un program care se putea auto-replica, scris de Bob Thomas la BBN Technologies, fiind considerat primul malware. „Creeper Worm” se putea copia pe un alt sistem din aceeaşi reţea unde afişa mesajul: „I’m the creeper, catch me if you can!” (trad. Eu sunt creeper, prinde-mă dacă poţi!)[1].

2.2. Tipuri de malware

Următoarele categorii descriu câteva funcţionalităţi pe care un malware le poate utiliza în acţiunile sale; malware-ul poate face parte din mai multe categorii.

1. Virus

   Un virus este un software maliţios care, când este executat, se răspândeşte prin modificarea altor programe de pe calculator prin introducerea codului propriu. Când replicarea reuşeşte, zonele afectate se numesc zone „infectate" cu un virus. Autorii de viruşi folosesc decepţia *social engineering* (trad. inginerie socială) şi cunoştinţe detaliate în exploatarea vulnerabilităţilor de securitate pentru infecţia iniţială a sistemelor şi pentru răspândirea virusului. Majoritatea viruşilor au ca obiectiv infectarea sistemelor care rulează Microsoft Windows, folosind o varietate de mecanisme pentru infectarea noilor gazde, şi de asemenea folosesc numeroase strategii anti-detecţie pentru evitarea produselor antivirus. Viruşii cauzează în prezent pierderi de milioane de dolari în fiecare an, datorat provocării defecţiunilor de sistem, irosirea resurselor sistemelor, coruperii datelor, creşterea costurilor de mentenanţă, etc. Termenul de „virus" este folosit în mod greşit prin referirea la alte tipuri de malware. Majoritatea ameninţărilor active cu malware fiind defapt programele de tip *trojan horse* sau *worm* decât viruşii[27].

2. *Worm* (trad. vierme)

   Un *worm* este un software autonom care se răspândeşte automat pentru a infecta alte calculatoare. De obicei malware-ul de tip *worm* foloseşte o reţea pentru a se răspândii, bazându-se pe vulnerabilităţi de securitate de pe calculatoarele ţintă pentru a le accesa. Infecţiile de tip *worm* cauzează mereu stricăciuni reţelei, chiar dacă doar prin consumarea lăţimii de bandă, în timp ce viruşii întotdeauna corup sau modifică fişiere pe maşina atacată. Multe tipuri de *worm* sunt proiectaţi doar pentru a se răspândi şi nu încearcă să modifice sistemele pe care le infectează. Termenul de „*worm*" a fost folosit pentru prima dată de către John Brunner în 1975, în romanul său „The Shockwave Rider"[8]. În acest roman, Nichlas Haflinger proiectează şi lansează un *worm* pentru adunarea datelor într-un act de răzbunare împotriva unui om influent care conduce o reţea naţională de informaţii care induce supunerea tuturor. Pe 2 noiembrie 1988, Robert Tappan Morris, un absolvent în informatică de la universitatea

Cornell, lansează malware-ul care va fi cunoscut drept „Morris worm", distrugând o mare parte din calculatoare - o zecime din calculatoarele conectate din acea vreme. Morris fiind prima persoană judecată şi condamnată în temeiul CFAA - „Computer Fraud and Abuse Act"[28].

3. *Trojan* (trad. troian)

Trojan este orice formă de malware care păcălește utilizatorul cu privire la intenţia sa. Termenul de *„trojan*" provine din Grecia Antică, o poveste despre un cal de lemn care a dus la cucerirea oraşului Troia. Troienii sunt răspândiți printr-o formă de *social engineering*, ca de exemplu atunci când un utilizator este păcălit în a executa un ataşament dintr-un e-mail care pare inofensiv sau prin accesarea unor reclame false. Cu toate că *payload-ul* (trad. încărcătura) acestora poate fi orice, multe dintre acestea acţionează sub forma de *„backdoor*", contactând atacatorul care poate avea acces neautorizat asupra calculatorului infectat. Troienii pot permite atacatorilor să acceseze informațiile personale ale utilizatorilor precum informații bancare, parole sau identitatea personală. Poate de asemenea să șteargă fișierele utilizatorilor sau să infecteze alte calculatoarea conectate la reţea. În atacurile de tip *ransomware* este folosit adesea *malware-ul* de tip *trojan*. Spre deosebire de virus sau *worm*, troienii în general nu încearcă să se injecteze în alte fișiere sau să se propage[36].

4. *Rootkit*

Un *rootkit* este o colecţie de software, de obicei maliţios, creat pentru a permite accesul la un calculator sau o zonă din software care în mod normal nu ar fi posibilă și adesea maschează existenţa acestuia sau existenţa unui alt software. Termenul de *rootkit* este concatenarea cuvântului „*root*" (utilizatorul cu drepturi privilegiate din sistemele de operare Unix) şi cuvântul „*kit*" (care se referă la componenta software care implementează instrumentul. Instalarea *rootkit-ului* poate fi automată, sau poate fi instalat de un atacator după obţinerea drepturilor de administrator. Obţinerea acestor drepturi de acces este rezultatul unui atac direct asupra sistemului; de exemplu, exploatarea unei vulnerabilităţi cunoscute (pentru ridicarea privilegiilor) sau a unei parole (obţinute prin forţa brută sau prin tactici de *social engineering* precum *phishing*. Odată instalat, este posibilă ascunderea acestuia precum și menținerea drepturilor de acces. Detecţia *rootkit-ului* este dificilă pentru că acesta poate împiedica software-ul menit să îl găsească. Ștergerea *rootkit-ului* poate fi complicată sau practic

imposibilă, mai ales în cazurile în care *rootkit-ul* se află în nucleul sistemului; reinstalarea sistemului de operare fiind poate singura soluție[35].

5. *Ransomware*

Ransomware-ul este un tip de malware din categoria „*cryptovirology*" (trad. cripto virologiei) care amenință să publice datele victimei sau blochează permanent datele dacă o răscumpărare nu este plătită. În timp ce unele fișiere malware de timp *ransomware* mai simpli pot bloca sistemul într-un fel în care o persoană specializată ar putea debloca, fișierele malițioase mai avansate utilizează tehnici „*cryptoviral extorion*", în care criptează fișierele victimei, făcându-le inaccesibile, cerând plata unei răscumpărări pentru decriptarea acestora. Atacurile de tip *ransomware* sunt de obicei efectuate cu ajutorul *malware-ului* de tip *trojan*, utilizatorul fiind păcălit în descărcarea sau deschiderea acestuia atunci când *malware-ul* ajunge sub forma unui atașament email. Totuși, *malware-ul* „WannaCry worm" se răspândea automat între calculatoare fără interacțiunea unui utilizator. Începând în jurul anului 2012, utilizarea atacurilor cu *ransomware* a crescut internațional. În primele șase luni ale anului 2018, s-au înregistrat 181.5 milioane de atacuri, o creștere cu 229% față de atacurile din aceeași perioadă a anului 2017. Atacul cu „CryptoLocker" a fost foarte distrugător, obținând câștiguri de aproximativ 3 milioane de dolari. Un al exemplu este „CryptoWall", care conform Biroului Federal de Investigații (eng. *Federal Bureau of Investigation* - FBI) a acumulat 18 milioane de dolari până în iunie 2015[34].

6. Botnet

Un *botnet* este un număr de mai multe calculatoare conectate prin Internet, fiecare calculator rulând unul sau mai mulți *bots* (trad. roboți). Botnet-urile pot fi folosite pentru efectuarea atacurilor de tip DDoS (eng. *distributed denial-of-service*), furtului de date, trimiterea de mesaje spam, permițând atacatorului să acceseze calculatorul infectat și conexiunile acestuia. Posesorul poate controla botnetul folosind programe de tip *command and control* - CC (trad. control și comenzi). Un botnet poate să fie o colecție de dispozitive conectate prin Internet ca și calculatoare, telefoane, dispozitive IoT (eng. *Internet of Things*) a căror securitate a fost compromisă. Un astfel de dispozitiv compromis se numește *bot* (trad. robot), creat în momentul în care dispozitivul este compromis de un malware. Deținătorul *botnetului* are posibilitatea să controleze activitatea calculatoarelor compromise prin diverse canale de comunicare

formate din protocoale de reţea de bază precum IRC (eng. *Internet Relay Chat)* şi HTTP (eng. *Hypertext Transfer Protocol*)[26].

2.3. Soluţii

Un program antivirus de încredere este esenţial pentru securitatea unui sistem. Antivirusul este un program special creat pentru detecţia, prevenirea şi înlăturarea programelor maliţioase. Programele de tip antivirus folosesc semnături unice pentru detecţia viruşilor. O semnătură de virus se bazează pe o caracteristica invarianta a virusului; semnăturile sunt distribuite sub forma de actualizări pentru antivirus[12].

Pe lângă programele software care protejează automat calculatorul de viruşi, un alt lucru important pentru protejarea calculatorului este schimbarea obiceiurilor defectuoase. În primul rând, evitarea deschiderii de email-uri sau ataşamente ale acestora primite din surse necunoscute. Chiar dacă persoana care a trimis email-ul este una cunoscută trebuie verificat cu atenţie că ataşamentul email-ului este legitim. O cale prin care malware-ul se răspândeşte este aceea de a se copia şi trimite persoanelor din lista de contacte prin intermediul poştei electronice[21].

3. Analiza de malware

Analiza de malware este studiul sau procesul pentru determinarea funcţionalităţii, originii şi potenţialului impact al unui anumit tip de malware ca virus, *worm*, *trojan*, rootkit sau *backdoor*[33]. Tehnicile pentru analiza de malware permit unui analist să înţeleagă repede şi în detaliu riscul şi intenţia unui malware. Această perspectivă permite analistului să reacţioneze la noi tipuri de malware dezvoltat sau să îmbunătăţească tehnicile de detecţie existente pentru diminuarea ameninţărilor provenite de la malware-ul analizat. Dorinţa analiştilor este aceea de a înţelege comportamentul unui malware, iar intenţia autorului de malware este aceea de a ascunde comportamentul maliţios a malware-ului dezvoltat. Pentru că tehnicile şi uneltele de analiză devin din ce în ce mai elaborate, atacatorii dezvoltă tehnici de evaziune cu scopul de a împiedica analiza malware-ului creat. Tehnici precum auto-modificarea sau generarea de cod dinamic, precum şi abordări care detectează prezenţa unui mediu de analiză, astfel permiţând malware-ului să ascundă comportamentul maliţios[3]. În principal, există trei cazuri care determină necesitatea analizei de malware[33]:

- incidente în securitatea calculatoarelor: dacă o organizaţie descoperă sau suspectează că sistemele lor ar putea fi infectate cu malware, o echipă responsabilă cu securitatea poate să efectueze analiză de malware asupra potenţialelor exemplare descoperite în timpul procesului de investigaţie, pentru a determina dacă probele sunt cu adevărat infecţii, precum şi impactul pe care malware-ul l-a avut asupra sistemelor atacate din interiorul organizaţiei[33];
- cercetare de malware: cercetători din mediul academic sau din industria de securitate pot efectua analiză de malware pentru a înţelege cum se comportă malware-ul şi cele mai recente tehnici folosite pentru dezvoltarea acestuia[33];
- extracţia indicatorilor de compromis (eng. *indicator of compromise*): dezvoltatori de produse şi soluţii software pot efectua analiza de malware pentru a determina noi potenţiali indicatori de compromis; aceste informaţii pot ajuta produsele şi soluţiile de securitate în apărarea organizaţiilor împotriva atacului cu malware[33].

3.1. Analiza statică

Analiza statică sau analiza de cod este efectuată prin „disecţia” codului executabil al fişierului binar fără a-l executa. Fişierul binar poate fi de asemenea dezasamblat sau *reverse*

engineered (trad. inversat) folosind dezasambloare. Codul maşină poate fi de obicei translatat în cod de asamblare care poate fi citit şi înţeles de oameni: analistul de malware poate mai apoi înţelege instrucţiunile de asamblare şi îşi poate forma o imagine despre intenţiile pe care le are programul analizat. Unele tipuri de malware sunt dezvoltate utilizând tehnici de protecţie împotriva analizei statice, de exemplu prin introducerea construcţiilor sintactice funcţionale, dar care nu corespund cu aşteptările sau presupunerile dezasamblorului[33].

Pattern-urile utilizate în analiza statică includ semnături de stringuri, apeluri ale funcţiilor din librării, grafuri de control şi distribuţia octeţilor de cod. Executabilul trebuie să fie despachetat (eng. *unpacked*) şi decriptat (eng. *decrypted*) înainte ca analiza statică să poată avea loc. Instrumente de dezasamblare/depanare şi dumpere de memorie pot fi folosite pentru a inversa executabile de Windows compilate. Dezasambloare/depanatoare precum IDA Pro şi OllyDbg afişează codul executabilului ca instrucţiuni de asamblare, care furnizează multe informaţii referitor la acţiunile pe care software-ul analizat le întreprinde şi *pattern-uri* pentru identificarea atacurilor[4].

De obicei, codul sursa al malware-ului nu este disponibil. Acest fapt reduce aplicabilitatea tehnicilor de analiză statică la executabilele care fac posibilă obţinerea de informaţii din fişierul binar al malware-ului. De exemplu, un număr mare de programe maliţioase execută instrucţiuni din setul de instrucţiuni IA32 (eng. Intel Architecture, 32-bit). Astfel, dezasamblarea programelor de acest fel pot produce rezultate ambigue dacă fişierul executabil foloseşte tehnici de auto-modificare a codului. Autorii de malware cunosc limitările metodelor analizei statice şi de aceea vor încerca să creeze instanţe malware care folosesc tehnici de prevenire a analizei statice. Prin urmare, este necesar să fie dezvoltate tehnici de analiză rezistente la aceste modificări, fiabile analizei de software maliţios[20].

3.2. Analiza dinamică

Analiza dinamică sau comportamentală se realizează prin observarea comportamentului malware-ului în timp ce acesta rulează pe un sistem gazdă. Această formă de analiză se efectuează într-un mediu *sandbox* pentru a preveni infecţia sistemelor din producţie; multe astfel de *sandbox-uri* sunt sisteme virtuale care pot fi uşor aduse în starea iniţială - după ce analiza a fost terminată. Programe malware mai recente pot expune o varietate de tehnici proiectate pentru protejarea împotriva analizei dinamice. Astfel de tehnici pot să includă teste pentru determinarea dacă malware-ul este rulat într-un mediu virtual sau

sunt ataşate depanatoare, amânarea execuţiei codului maliţios, sau necesitatea interacţiunii unui utilizator[33].

În principal, pot fi identificate două abordări ale analizei dinamice:

- analiza diferenţelor dintre două momente de timp definite: un malware specific este executat pentru o anumită perioadă de timp, după care modificările făcute sistemului sunt analizate prin compararea stării curente cu cea iniţială[20];
- observarea comportamentului în timpul rulării: activitatea maliţioasă pe care malware-ul o efectuează este monitorizată în timpul rulării utilizând instrumente specializate[20].

Un exemplu al primei abordări este Truman (eng. „The Reusable Unknown Malware Analysis Net"). Un malware este executat pe un sistem real cu Windows nu într-un mediu virtual. În timpul rulării, Truman oferă o conexiune virtuală către Internet pe care malware-ul o poate folosi pentru comunicarea în reţea. După execuţie, maşina gazdă este repornită, încărcându-se o imagine de Linux, care mai apoi extrage datele relevante din imaginea de Windows anterioară, precum chei de regiştri şi o listă a fişierelor. În final, imaginea Windows este adusă la starea iniţială. Prin folosirea unui mediu nativ, Truman împiedică posibilele măsuri de protecţie pentru depanare ale malware-ului. Cu toate acestea, deoarece rezultatul este numai o captură a sistemului infectat, informaţiile referitoare la activitatea dinamică precum procesele create şi fişierele temporar create sunt pierdute[20].

Deoarece un număr mare de noi tipuri de malware ajung la producătorii de soluţii antivirus în fiecare zi, o abordare automată este necesară pentru a limita numărul celor care necesită analiza mai amănunţită a unui specialist[4].

3.3. Analiza automată

Deoarece analiza de malware poate să fie costisitoare atât din punct de vedere al timpului, cât şi din punct de vedere financiar, au fost dezvoltate metode de analiză automată a tipurilor de malware. Analiza automată este constituită din utilizarea tehnicilor de analiză statică sau dinamică într-o manieră automată cu scopul de a analiza un număr mare de fişiere maliţioase într-un timp relativ scurt.

Pe lângă analiza unor cantităţi mari de fişiere maliţioase, analiza automată poate să se folosească de tehnici precum *machine learning* şi *threat intelligence* pentru evaluarea rezultatelor obţinute în urma analizei statice sau dinamice. *Threat Intelligence* se bazează pe

colecţii de informaţi provenite din date publice disponibile online (eng. *Open-source intelligence - OSINT*), date din mediul social provenite din utilitare sau soluţii care permit monitorizarea conversaţiilor şi canalelor sociale (eng. *Social media intelligence - SOCMINT*), date provenite din contactele personale (eng. *Human intelligence - HUMINT*), date tehnice (eng. Technical intelligence - TECHINT) sau informaţii din *deep* şi *dark web*.

4. State of the Art

Mai sus am prezentat tehnicile prin care un „sample” de malware poate fi analizat, fie prin tehnici care nu necesită execuţia propriu zisă a acestuia, precum şi prin tehnici care analizează comportamentul maliţios din timpul activităţilor malware-ului în timpul rulării. Deoarece foarte multe tipuri de malware apar în fiecare zi, dezvoltarea unor platforme automate care să efectueze analiză de malware a devenit o necesitate. O aplicaţie automată care efectuează o analiză de malware ar putea efectua aceasta cu ajutorul tehnicilor de analiză statică, analiză dinamică sau analiză de trafic de reţea.

În continuare, vom prezenta câteva aplicaţii care efectuează analiză de malware într-un mod automat prin folosirea tehnicilor descrise anterior.

4.1. Platforma SNDBOX

SNDBOX este o platformă puternică, multi-vector, bazată pe inteligenţă artificială în cloud care pe lângă analiza fişierelor de atribute şi vectori prin monitorizarea comportamentului, are de asemenea capacitatea de a converti intrările de comportament dinamic în vectori de căutare, astfel permiţând utilizatorilor să caute în baza de date a SNDBOX un număr mare de rezultate a experimentelor în urma analizei de malware[14].

Pentru eficientizarea monitorizării, SNDBOX execută binarele încărcate într-un mediu controlat utilizând un agent „kernel-mode” invizibil, care păcăleşte malware-ul în a crede că acesta este executat într-un sistem real pe care acesta vrea să îl atace. Localizat între nivelul utilizator şi kernel, agentul invizibil „kernel-mode” al SNDBOX monitorizează indetectabil acţiunile pe care fişierul maliţios le întreprinde[14].

SNDBOX monitorizează comportamentul executabilelor, începând de la simple modificări ale sistemului de resurse până la activităţi avansate ale reţelei, după care utilizează algoritmi de inteligenţa artificială pentru procesarea numărului mare de date colectate, care pot fi peste 200MB pentru un executabil binar de doar 10KB, pentru a produce rezultate mult mai uşor de înţeles. Platforma este proiectată astfel încât să dezvolte într-un mod automat cunoştinţe şi înţelegerea în amănunt asupra anumitor aspecte, amprente de comportament, vectori, atribute, clasificări şi semnături, în timp, din investigaţiile sample-urilor încărcate şi analizate[14].

Odată încărcat un fişier în platformă, SNDBOX începe să efectueze analiză statică şi dinamică pe fişier. La terminarea analizei fişierului, platforma furnizează un raport cu trei secţiuni care poate fi utilizat pentru a învăţa mai multe despre malware şi ce acţiuni a întreprins, şi de asemenea un raport în format JSON poate fi descărcat. Cele trei secţiuni sunt[10]:

- analiza statică - figura 4.1.1 - permite vizualizarea informaţiilor despre fişierul încărcat precum „metadata", tabela de secţiuni, tabela de importuri şi exporturi. Desigur, aceste informaţii pot fi obţinute cu multe alte utilitare şi site-uri disponibile online[10].

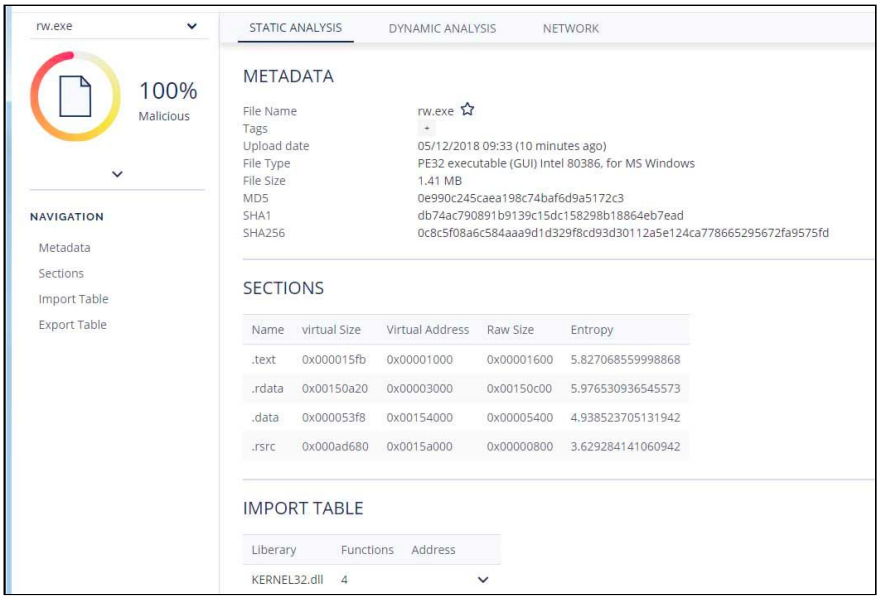


Figura 4.1.1. Secţiunea de analiză statică din SNDBOX

- analiza dinamică - figura 4.1.2 - este secţiunea în care adevărata putere a SNDBOX intră în joc. În timpul analizei, SNDBOX va urmării toate fişierele şi procesele care au fost create, precum şi orice apel către funcţiile de sistem, interogări şi modificări de regiştri şi cereri WMI (eng. Windows Management Instrumentation). Inteligenţa artificială este folosită când se analizează amprentele execuţiei malware-ului şi a codului pentru a clasifica sample-ul ca un anumit tip de malware sau comportament. De exemplu, bazat pe faptul că malware-ul încearcă să şteargă „Shadow Volume Copies" - este o tehnologie inclusă în Microsoft Windows care poate crea copii „backup" a fişierelor sau volumelor, chiar dacă acestea sunt folosite - acesta va fi adăugat în categoria de „Ransomware" şi dacă creează un alt executabil, va fi adăugat

în categoria de „Dropper”. Tot în această secţiune vor fi afişate fişierele create, va căuta şiruri de caractere deosebite şi dacă este posibil le va decodifica[10].

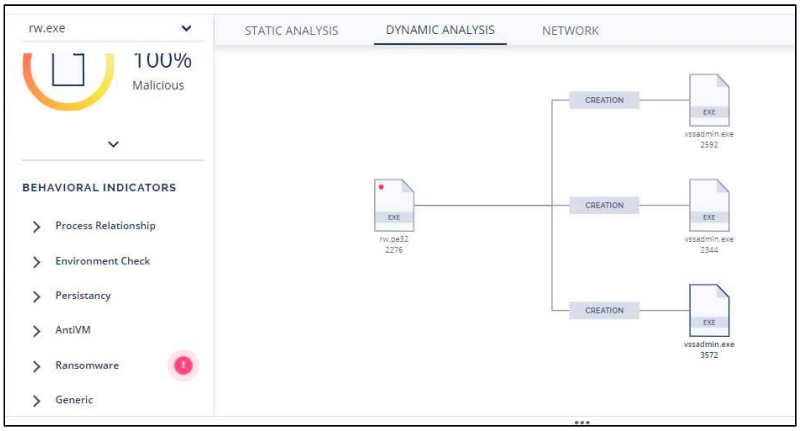
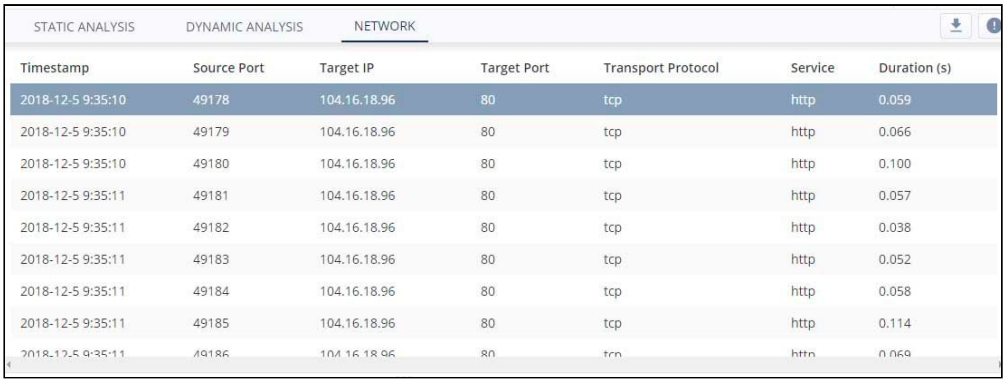


Figura 4.1.2. Secţiunea de analiză dinamică din SNDBOX

- analiza traficului de reţea - figura 4.1.3 - permite vizualizarea întregului trafic de reţea produs în timpul rulării malware-ului. Utilizând aceste informaţii, inteligenţa artificială va căuta informaţii neobişnuite. Acestea vor putea ajuta utilizatorul să descopere repede traficul de reţea rar sau neobişnuit. De asemenea, traficul de reţea va fi împărţit în diferite servicii internet, precum DNS (eng. *Domain Name System*) sau HTTP pentru ca utilizatorul să se poată preocupa doar de anumite servicii[10].



Timestamp	Source Port	Target IP	Target Port	Transport Protocol	Service	Duration (s)
2018-12-5 9:35:10	49178	104.16.18.96	80	tcp	http	0.059
2018-12-5 9:35:10	49179	104.16.18.96	80	tcp	http	0.066
2018-12-5 9:35:10	49180	104.16.18.96	80	tcp	http	0.100
2018-12-5 9:35:11	49181	104.16.18.96	80	tcp	http	0.057
2018-12-5 9:35:11	49182	104.16.18.96	80	tcp	http	0.038
2018-12-5 9:35:11	49183	104.16.18.96	80	tcp	http	0.052
2018-12-5 9:35:11	49184	104.16.18.96	80	tcp	http	0.058
2018-12-5 9:35:11	49185	104.16.18.96	80	tcp	http	0.114
2018-12-5 9:35:11	49186	104.16.18.96	80	tcp	http	0.069

Figura 4.1.3. Secţiunea de analiză de reţea din SNDBOX

Pe lângă partea vizuală prin care un utilizator poate interacţiona cu platforma SNDBOX, aceasta dispune şi de un API (eng. *Application Programming Interface*) prin care utilizatorii pot accesa rapoarte, descărca sau încărca „sample-uri” de malware. Astfel, acest API pus la dispoziţie de către dezvoltatori face posibilă realizarea de sisteme automate care

pot analiza malware şi astfel se poate reduce semnificativ necesitatea ca o persoană specializată să efectueze cercetări mai amănunţite.

4.2. Cuckoo Sandbox

Cuckoo Sandbox, figura 4.2.1, este un sistem automat de analiză de malware open-source care poate fi folosit împotriva diferitelor tipuri de malware, de la documente Office până la executabile. Maşina pe care se pot face verificările poate sa fie de tip Windows, Linux, macOS sau Android[22].

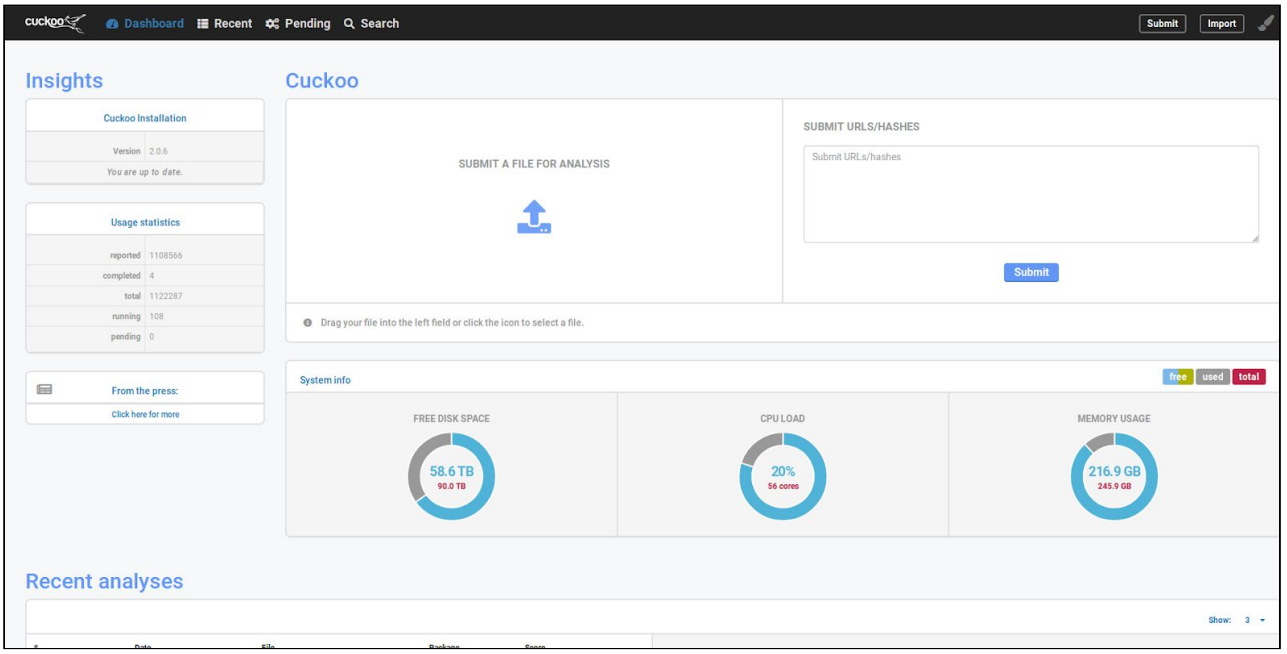


Figura 4.2.1. Tablou de control din Cuckoo

Un fişier poate fi încărcat în Cuckoo prin intermediul unei interfeţe web, printr-o interfaţă programabilă - API, sau cu ajutorul consolei. Cuckoo va încerca să determine cea mai bună metodă de analiză şi imaginea virtuală pe care o va folosi. Cuckoo programează un „task”, mai apoi încarcă imaginile virtuale pentru a executa sarcina[22].

Platforma Cuckoo este scrisă în Python şi are capacitatea de a lucra cu multiple hipervizoare. Cel mai folosit hipervizor este VirtualBox, deoarece este gratuit şi uşor de instalat. Pe lângă acesta, următoarea lista descriere diferitele hipervizoare pe care Cuckoo le suportă: ESX/ESXI, KVM, QEMU, VMware, vSphere, XenServer, Proxmox[22].

Odată ce este configurat un hipervizor pentru a funcţiona cu Cuckoo, fişierele încărcate sunt trimise către imaginea virtuală selectată - această imagine rulează un script „agent” prin care Cuckoo comunică cu aceasta. Acest „agent” este responsabil pentru pornirea şi colectarea datelor din timpul execuţiei malware-ului. La final, agentul trimite datele colectate sistemului gazdă pentru a fi conceput un raport - figura 4.2.2 - cu privire la activitatea malware-ului[22].

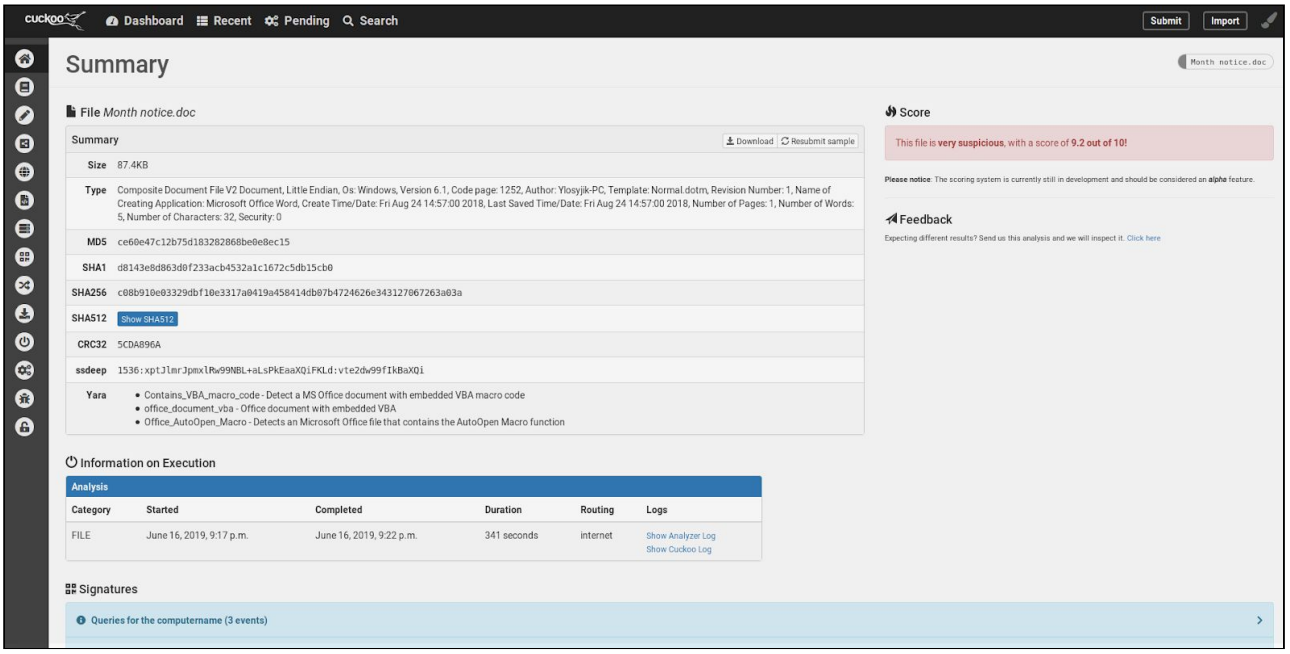


Figura 4.2.2. Raport de analiză din Cuckoo

4.3. Hybrid Analysis

Hybrid Analysis, figura 4.3.1, este un website care oferă servicii de analiză de malware gratuite. Utilizând website-ul ai posibilitatea de a încărca fişiere pentru a fi analizate în detaliu, atât folosind tehnici de analiză statice, cât şi dinamice. Hybrid Analysis foloseşte pentru analiza unui malware tehnologia „Falcon Sandbox” oferită de CrowdStrike[7].

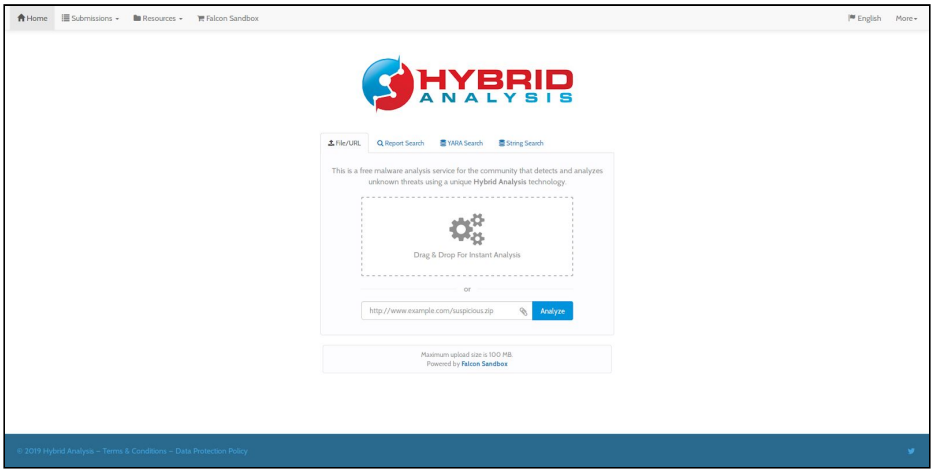


Figura 4.3.1. Pagina principală a Hybrid Analysis

„CrowdStrike Falcon Sandbox" este o soluție automată de analiză de malware care oferă putere echipelor de securitate prin combinarea threat intelligence cu cea mai puternică soluție sandbox. Această combinație unică oferă informații detaliate, permițând analiștilor să înțeleagă mai bine atacurile sofisticate cu malware și să își îmbunătățească apărarea. „Falcon Sandbox" efectuează analiza profundă a amenințărilor evazive și necunoscute, combină rezultatele cu threat intelligence și oferă indicatori de compromis. „Falcon" permite echipelor de securitate cibernetică de orice nivel să își îmbunătățească înțelegerea despre amenințările cu care se confruntă, folosind cunoștințele în apărarea atacurilor viitoare[2].

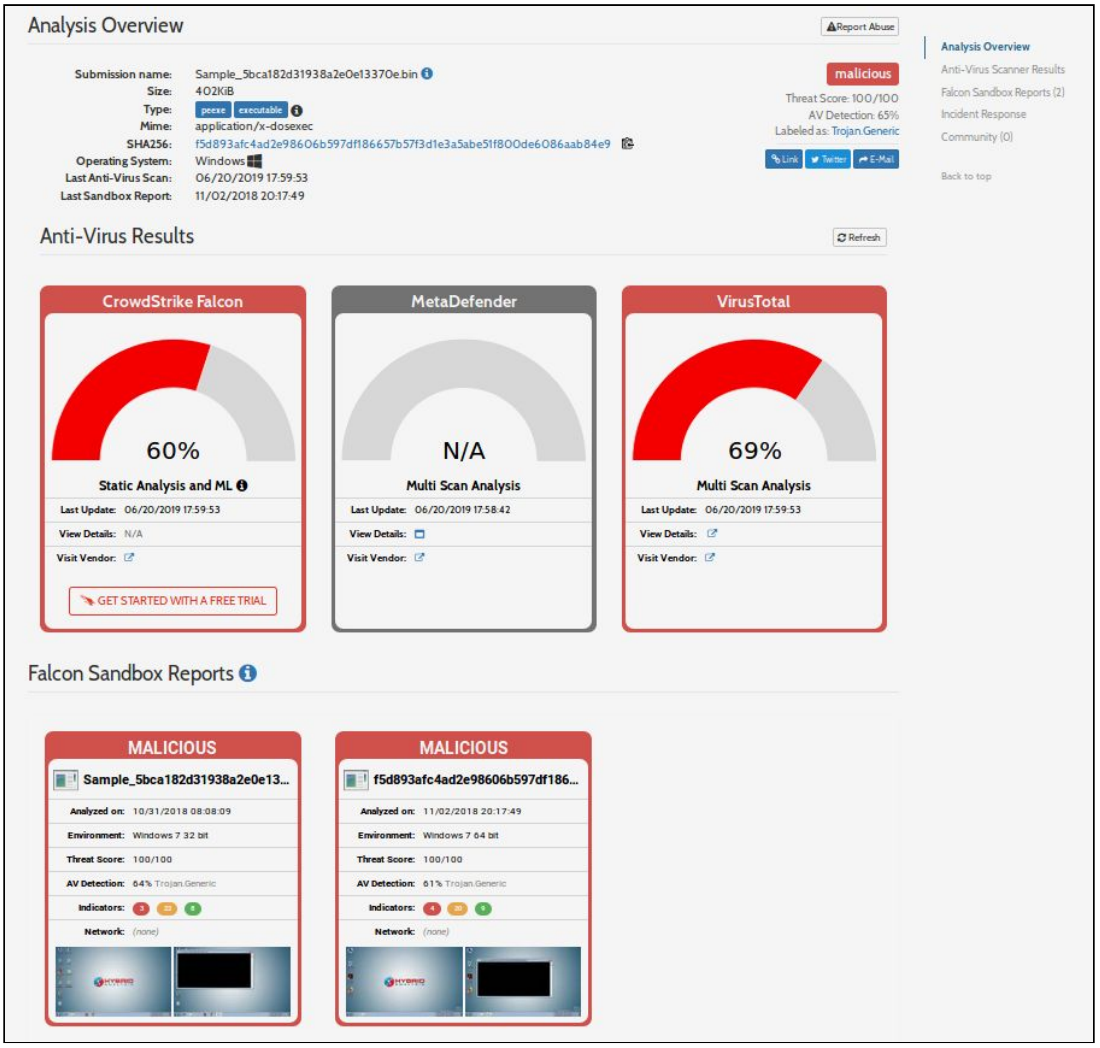


Figura 4.3.2. Prezentarea generală a unui raport de analiză

Cum se poate observa în figura 4.3.2, Hybrid Analysis afişează câteva informaţii de bază despre fişierul încărcat în sistem, scanează cu ajutorul câtorva soluţii antivirus fişierul încărcat afişând probabilitatea ca sample-ul să fie maliţios, dar şi rapoartele de analiză pentru diferite versiuni de Windows. Pentru fiecare din aceste rapoarte există posibilitatea afişării indicatorilor de compromis, precum şi a rapoartelor de analiză statică şi dinamică. În figura 4.3.3 se poate observa structura unui raport de incident care conţine secţiuni precum: detalii despre fişier, capturi de ecran din timpul execuţiei, analiza de reţea, etc.

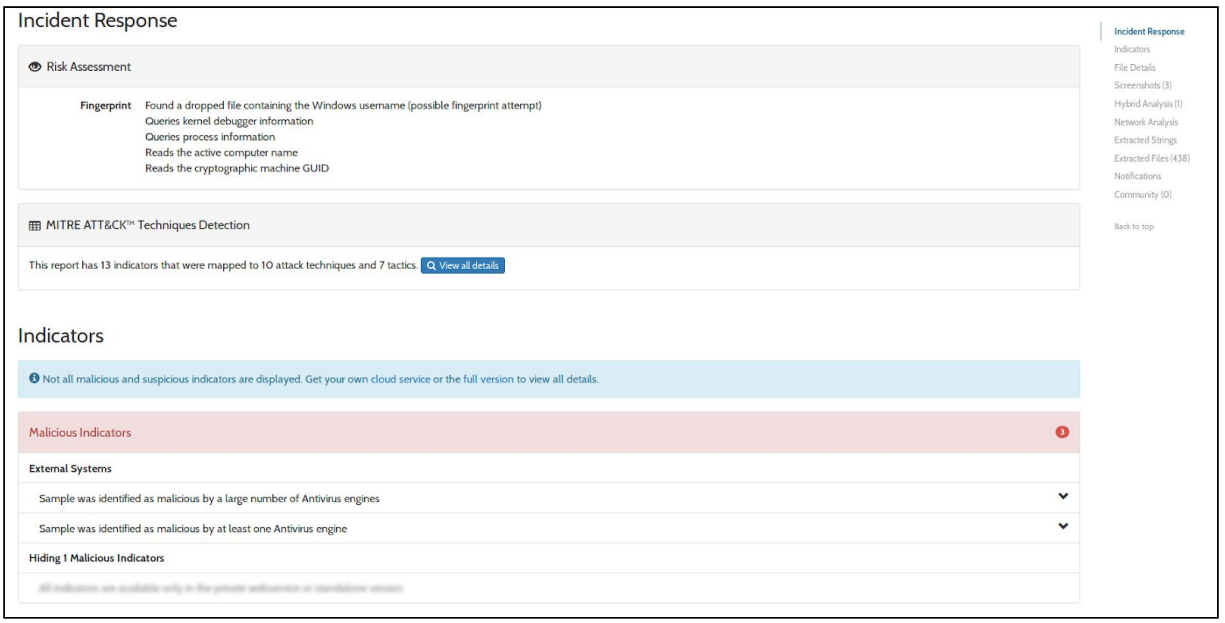


Figura 4.3.3. Raport de incident din Hybrid Analysis

4.4. VirusTotal

VirusTotal este un website creat de o companie de securitate spaniolă, Hispasec Sistemas. Lansat în iunie 2004, a fost cumpărat de Google Inc. în septembrie 2012. În ianuarie 2018, compania Chronicle, subsidiar al Alphabet Inc., preia conducerea VirusTotal[37].

VirusTotal are incorporate produse antivirus şi motoare de scanare online pentru verificarea fişierelor utilizatorilor pentru care propriul antivirus ar fi putut face detecţii incorecte sau pentru verificarea falsurilor pozitive. Producătorii de soluţii antivirus pot primi copii ale fişierelor pe care alte produse le-au detectat ca fiind malware, dar au trecut verificările motorului lor - pentru a-i ajuta să-şi îmbunătăţească produsul. Pentru analiza dinamică de malware, VirusTotal foloseşte Cuckoo „sandbox”[37].

Platforma inspectează elementul încărcat cu peste 70 de scanere antivirus - figura 4.4.1 - şi servicii de *blacklisting* pentru domenii sau URL-uri (eng. *Uniform Resource Locator*), pe lângă alte utilitare pentru extragerea de semnale din elementul studiat. Orice utilizator are posibilitatea să selecteze un fişier de pe calculatorul personal utilizând browser-ul de internet şi să îl trimită către VirusTotal. Pe lângă metoda interfeţei web, VirusTotal oferă şi alte metode de încărcarea a elementelor în vederea analizei: aplicaţii de upload, extensii pentru browser, dar şi un API[23].

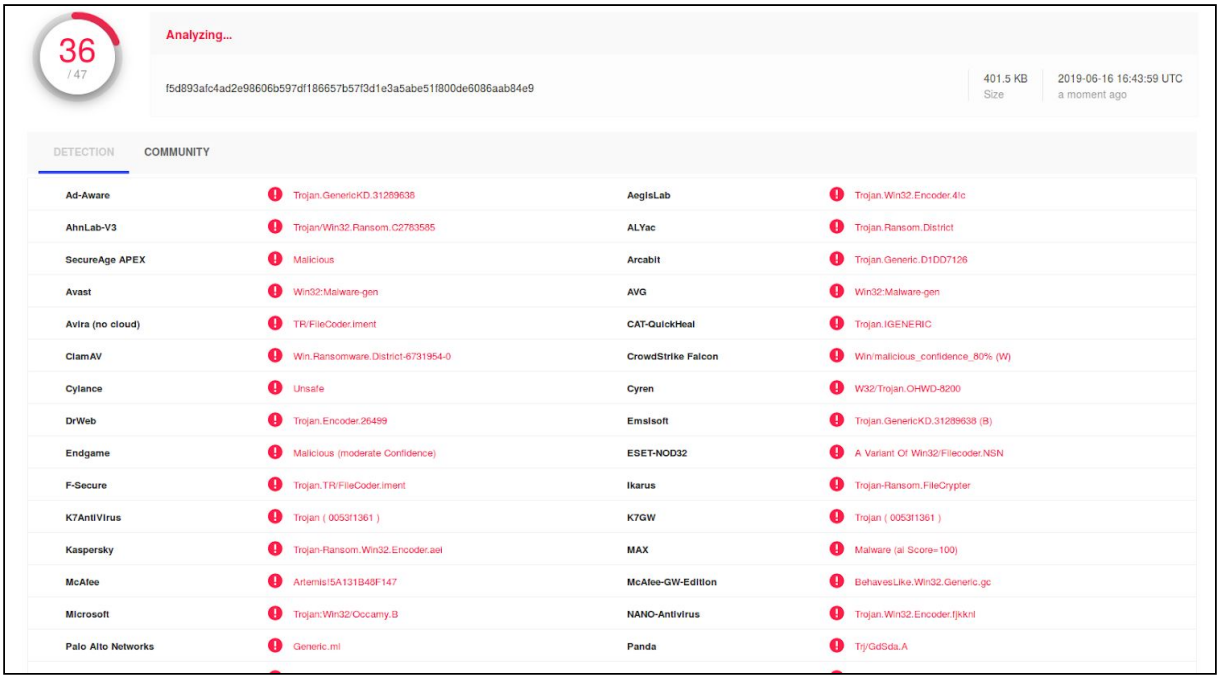


Figura 4.4.1. Analiza unui fişier malițios din platforma VirusTotal

5. Analiza automată a comportamentului dinamic al viruşilor

În această lucrare vom prezenta aplicaţia creată cu scopul de a încerca să demonstreze necesitatea şi utilitatea instrumentelor de analiză dinamică de malware. Deoarece analiza statică poate fi obţinută foarte uşor cu ajutorul multitudinii de utilitare disponibile pe Internet, aplicaţia dezvoltată în acest articol exploatează tehnicile de analiză dinamică deoarece se doreşte extragerea cât mai multor informaţii valoroase. De asemenea, pentru a uşura utilizarea aplicaţiei, aceasta incorporează atât o interfaţă web cât şi una de tip consolă. Interfaţa web ajută utilizatorul în crearea şi execuţia unui nou experiment în urma căruia are posibilitatea să vizualizeze un raport care încapsulează toate rezultatele pe care aplicaţia le extrage după terminarea experimentului. Interfaţa de tip consolă, una nu foarte atractivă pentru persoanele neexperimentate în tehnicile dezvoltării aplicaţiilor, este dezvoltată cu scopul de a ajuta utilizatorul să execute mai multe experimente succesive fără a fi nevoie de intervenţia utilizatorului pentru configurarea fiecărui experiment.

5.1. Dezvoltarea aplicaţiei

Pentru dezvoltarea aplicaţiei am ales folosirea limbajului de programare Python pentru avantajele şi *framework-urile* numeroase, pe care acest limbaj le oferă. Un avantaj important pe care limbajul îl oferă este portabilitatea, pentru că dezvoltând un software în Python, acesta poate fi utilizat cu uşurinţă pe un al sistem care rulează alt sistem de operare. Un alt beneficiu de care se poate beneficia este *framework-ul* bine dezvoltat şi documentat, VirtualBox-Python, care ajută în comunicarea cu mediul în care malware-ul îşi dezvăluie intenţiile maliţioase.

Mediul virtual în care experimentele se realizează este VirtualBox. VirtualBox este un hypervizor open-source dezvoltat de compania Oracle. VirtualBox poate fi instalat pe sistemele de operare Windows, macOS, Linux etc. care suportă crearea şi administrarea maşinilor virtuale precum Windows, Linux, Solaris etc. şi care are un API documentat pentru exploatarea facilităţilor puternice ale hypervizorului de către alte aplicaţii.

Deoarece analiza dinamică de malware presupune monitorizarea unor factori din interiorul unui mediu iniţial intact sau neinfectat, avem nevoie de câteva tool-uri care ne vor ajuta în extragerea indicatorilor care vor descriere comportamentul maliţios. Astfel, am ales să

studiem modificările pe care fişierului analizat le întreprinde asupra cheilor de regiştri şi a fişierelor de pe disc. Pe lângă aceşti factori pe care malware-ul ar putea să îi modifice în timpul execuţiei, procesele care rulează pe sistemul virtual la momentul rulării acestuia pot influenţa comportamentul malware-ului, iar o analiză a proceselor ar îmbunătăţii rezultatele experimentului.

„Hollow Hunter" este un program creat special pentru analiza proceselor de pe un sistem, identificarea şi extragerea indicatorilor maliţioşi precum modificări ale antetului fişierelor PE (eng. Portable Executable (trad. executabil portabil)), shellcode sau patch-uri în memorie. „Portable executable" este un format de fişier folosit în executabile, DLL-uri (eng. Dynamic Link Library), etc. din versiunile sistemului de operare Windows. Formatul PE este o structură de date necesară sistemului de operare Windows pentru încărcarea în memorie a fişierului care urmează să fie executat.

„ProcDump" este un executabil al cărui scop este acela de a monitoriza un proces specificat şi crearea unei capturi de memorie atunci când procesul examinat satisface anumite criterii: o anumită perioadă de timp s-a scurs, pragul de solicitare a procesorului s-a depăşit, etc.

Tehnicile prezentate mai sus vor fi folosite în examinarea şi monitorizarea comportamentului maliţios pentru o mai bună înţelegere a acestuia.

5.2. Implementare

Deoarece aplicaţia suportă atât o interfaţă web, cât şi una tip consolă, aceasta a fost gândită într-un mod în care frontend-ul să fie separat de partea de backend. Aşadar, librăria *vboxmachine* este interfaţa care trebuie folosită pentru a interacţiona cu nucleul aplicaţiei.

Interfaţa web este dezvoltată cu scopul de a face crearea experimentelor şi vizualizarea rapoartelor mai uşoară. Pe de altă parte, interfaţa de tip consolă a fost dezvoltată pentru a facilita execuţia secvenţială a mai multor experimente, care pot fi declarate într-un fişier de configurare cu structura ca în figura 5.2.1 sau date ca şi parametri în linia de comanda. Dacă se specifică un fişier de configurare, dar acesta nu conţine atribute necesare rulării, acestea vor fi luate din parametrii liniei de comandă, altfel este afişat un mesaj de eroare.

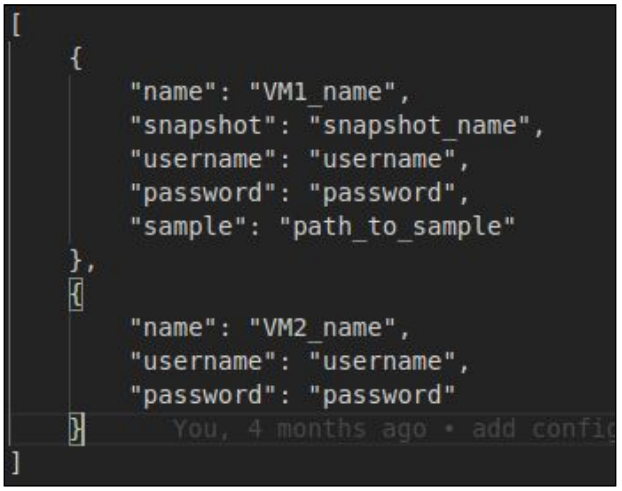


Figura 5.2.1. Fişier de configurare

```
alin@alin-laptop:~/work/licenta/project⟫ ./app.py -h
usage: app.py [-h] [-c CONFIG] -vm VM_NAME [-ss SNAPSHOT] -u USERNAME -p
              PASSWORD -s SAMPLE

optional arguments:
  -h, --help            show this help message and exit
  -c CONFIG, --config CONFIG
                        set configuration file with VMs
  -vm VM_NAME, --vm_name VM_NAME
                        set VM name
  -ss SNAPSHOT, --snapshot SNAPSHOT
                        set which snapshot to revert
  -u USERNAME, --username USERNAME
                        set username for VM
  -p PASSWORD, --password PASSWORD
                        set password for VM
  -s SAMPLE, --sample SAMPLE
                        set which sample to deploy on VM
alin@alin-laptop:~/work/licenta/project⟫ ./app.py -c config.json -s malware/sample.sample
```

Figura 5.2.2. Rularea cu un fişier de comandă şi un parametru

Pachetul *vboxmachine* foloseşte interfaţa de comunicare cu *hypervizorul* VirtualBox implementată cu ajutorul framework-ului VirtualBox-Python, care implementează toate componentele necesare pentru comunicarea VirtualBox cu o aplicaţie creată în Python. Clasa care creează obiectul pentru comunicare este *VBoxMachine* care are nevoie în mod obligatoriu de câteva detalii despre maşina virtuală pe care experimentul o să urmeze să fie rulat:

- numele maşinii aşa cum este afişat de VirtualBox - caracterele aldine prezentate în figura 5.2.3;

Figura 5.2.3. Numele, starea şi snapshot-ul maşinilor virtuale curente

- snapshotul la care maşina va fi refăcută înainte de pornirea experimentului - dacă nu este definit se va folosit primul snapshot capturat de pe maşină;
- numele de utilizator şi parola maşinii - necesare pentru copierea, rularea şi extragerea datelor capturate în timpul experimentului;
- calea spre fişierul maliţios care va fi executat în mediul virtual - această cale poate să fie una relativă la directorul curent sau una absolută.

Următorii parametrii pot să fie specificaţi, în caz contrar se va folosi valoarea lor implicită:

- timpul de aşteptare pentru rularea experimentului - implicit 30, este timpul pe care aplicaţia îl va aştepta după pornirea malware-ului în maşina, acest timp nu include iniţializarea şi pornirea serviciilor care vor monitoriza activitatea maşinii;
- tipul rulării maşinii virtuale - implicit „headless”, este metoda cu care VirtualBox va porni maşina virtuală în cauză; de menţionat este faptul că există trei metode prin care o maşină virtuală poate fi pornită: „gui” (această metodă necesită existenţa unei metode de afişare a ferestrelor, precum sistemele de operare de care nu implementează o interfaţă grafică), „headless” (folosită implicit de aplicaţie şi care nu presupune existenţa unei interfeţe grafice), „sdl” (folosită în general pentru depanare) şi „emergencystop” (valoare rezervată);
- numele arhivei cu informaţi de pe maşina virtuală - numele folosit pentru crearea arhivei care va fi extrasă de pe maşina şi care va conţine toate fişierele de monitorizare din timpul experimentului, implicit „extraction”;
- numele malware-ului - nu este decât denumirea sub care v-a fi copiat fişierul maliţios declarat mai sus pe maşina victimă, implicit „a”.

În urma validării acestor parametrii (parametrii precum numele şi parola de utilizator se vor putea verifica decât în timpul rulării), un experiment se poate crea şi se pot începe etapele de rulare a experimentului care cuprinde etapele care se pot vizualiza în figura 5.2.4.

```
vbx.restore_snapshot()
vbx.launch()
vbx.deploy_necessary_files()
vbx.launch_client_app()
vbx.power_off()
```

Figura 5.2.4. Etapele necesare rulării unui experiment

Astfel, există cinci faze care trebuie să se execute pentru ca un experiment să fie terminat cu succes: faza de restaurare a snapshot-ului, pornirea maşinii virtuale, copierea fişierelor necesare rulării experimentului, pornirea execuţiei utilitarului pe maşina gazdă precum şi extragerea rezultatelor, şi oprirea maşinii. Erorile care pot fi întâmpinate în timpul execuţiei unui experiment pot fi de tipul: maşina virtuală selectată este blocată de un alt proces care lucrează cu aceasta şi astfel nu se poate accesa imaginea cu drepturi de scriere; numele de utilizator şi parola specificate la pornirea experimentului sunt incorecte şi de aceea nu s-a putut crea sesiunea de comunicare cu maşina; drepturile utilizatorului logat nu sunt suficiente pentru copierea pe maşină a fişierelor necesare experimentului (fişierele care se vor folosi la execuţia experimentului sunt salvate în directorul *C:\maltest*) etc.

Fiecare operaţie executată de aplicaţie către maşina virtuală este sincronă, adică odată ce operaţia a fost apelată se aşteaptă ca aceasta să se execute complet sau să fie aruncată o eroare, caz în care execuţia se opreşte afişându-se eroarea respectivă.

Pentru că majoritatea aplicaţiilor care analizează comportamentul de malware folosesc imagini virtuale predefinite, aplicaţia este implementată astfel încât permite utilizatorului să creeze maşinile virtuale cu configuraţia dorită. Multe tipuri de malware au posibilitatea să infecteze prin reţeaua Internet şi alte calculatoare conectate, de aceea este recomandat ca orice conexiune la Internet să fie dezactivată, figura 5.2.5, şi eliminate directoarelor partajate cu drept de scriere sau de citire, figura 5.2.6.

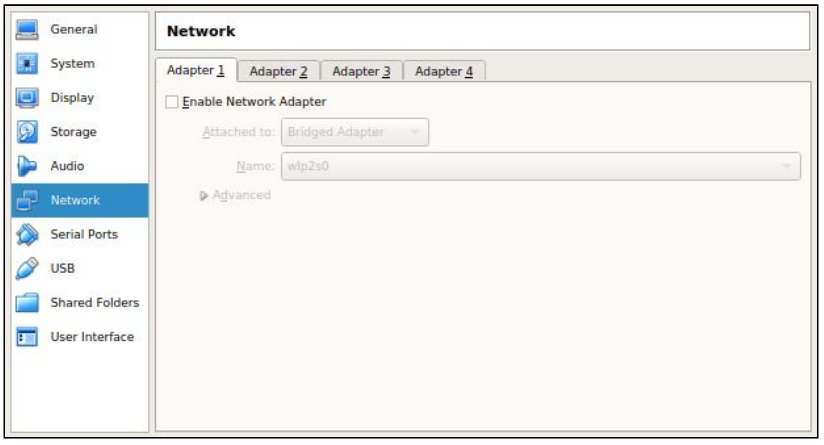


Figura 5.2.5. Adaptoarele de reţea configurate

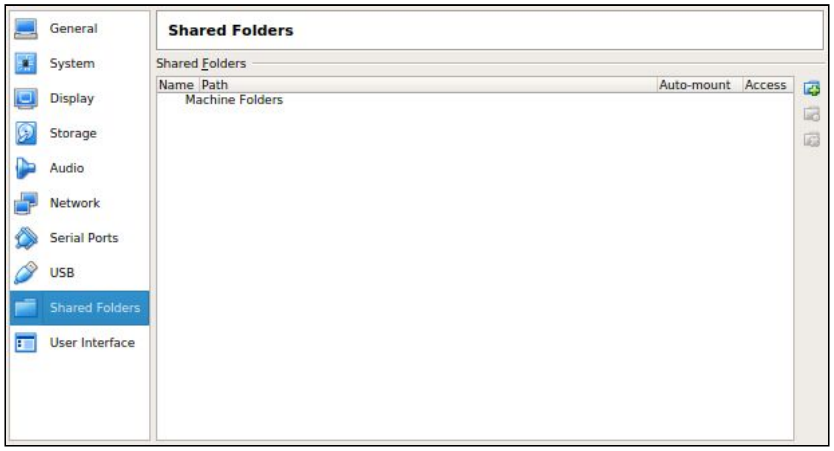


Figura 5.2.6. Directoarele partajate ale unei maşini virtuale

O parte importantă din analiza efectuată de aplicaţie este aplicaţia care execută, monitorizează şi asigură terminarea experimentului şi extragerea fişierelor de monitorizare. Aplicaţia *ClientApp* are ca scop pornirea tuturor utilitarelor de monitorizare a activităţii maliţioase din interiorul mediului de test, pornirea execuţiei malware-ului, precum şi oprirea acestuia.

Componentele de monitorizare a schimbărilor sistemului pe care este executat fişierul maliţios joacă de asemenea un rol important în dezvoltarea aplicaţiei. Aşadar, au fost dezvoltate două utilitare, care înregistrează modificările aduse sistemului de la începutul experimentului până la final. Prima aplicaţie este *FolderTool*, care cu ajutorul funcţiilor sistem din sistemul de operare Windows, capturează adăugări, modificări sau ştergeri ale fişierelor de pe disc. O a doua aplicaţie este *RegistryTool*, cu ajutorul căreia înainte de începerea procesului de infecţie, aplicaţia efectuează o captură a întregii ierarhii a regiştrilor Windows. La finalul experimentului efectuează o a doua captură pe care o compară cu prima, modificările înregistrate fiind salvate pe disc.

5.3. Manual de utilizare

Înainte de execuţia propriu-zisă a aplicaţiei, utilizatorul trebuie să creeze mediile de test pe care se vor executa în siguranţă experimentele. Aşadar, utilizatorul este nevoie să creeze o maşină virtuală de Windows, să asigure instalarea componentei *VirtualBox GuestAddition* - componentă care face posibilă comunicarea cu maşina - să creeze un cont de utilizator cu o parolă pentru mediul de lucru şi să facă o captură a sistemului în timpul rulării. După ce această cerinţă a fost satisfăcută, se poate începe utilizarea aplicaţiei de analiză. Aşa

cum am mai menţionat, se poate utiliza interfaţa *web* (figura 5.3.1) sau cea de tip consolă (figura 5.3.2). Pentru rularea de experimente este necesară furnizarea informaţiilor despre maşina virtuală, precum şi a fişierului care va fi executat şi monitorizat.

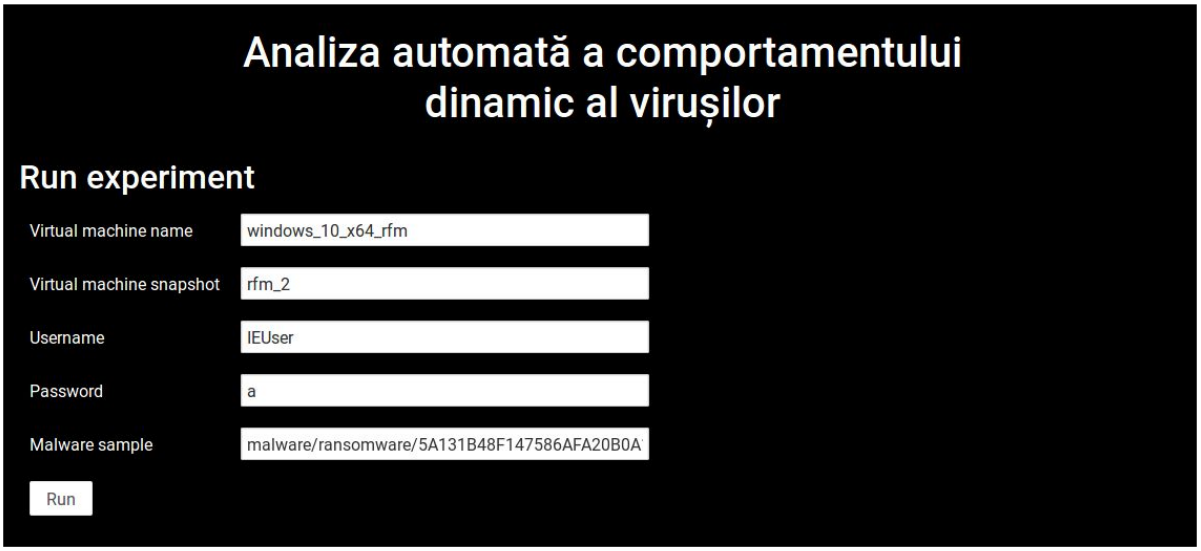


Figura 5.3.1. Crearea unui experiment din interfaţa web

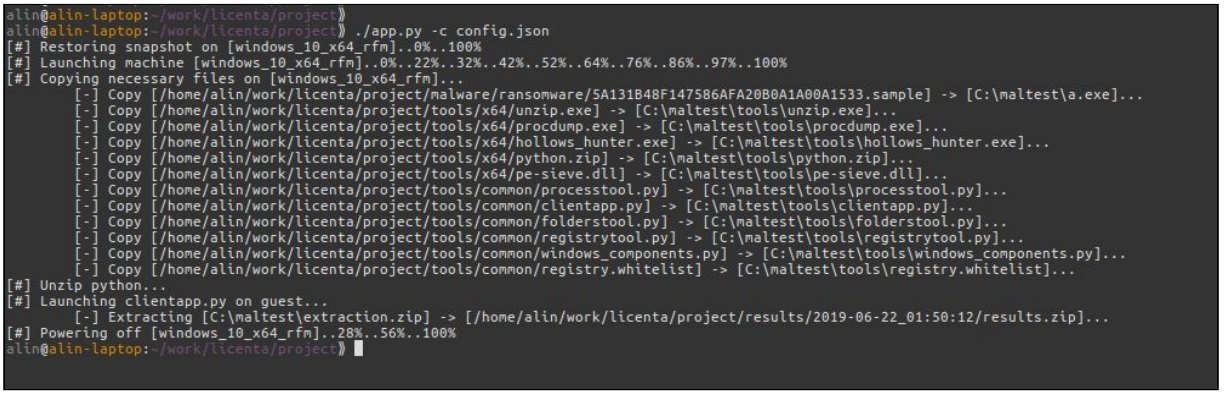


Figura 5.3.2. Crearea şi execuţia unui experiment din consolă

Dacă experimentul s-a terminat cu succes în directorul *results* a fost generat un subdirector cu datele extrase din mediul de test. Pentru vizualizarea rapoartelor este recomandată utilizarea interfeţei web, deoarece oferă o interfaţă plăcută cu toate informaţiile într-o singură pagină, împărţite pe secţiuni, ca în figura 5.3.3.

Figura 5.3.3. Vizualizarea unui raport

5.4. Extinderi posibile

Pentru o mai bună analiză a activităţii maliţioase a tipurilor de malware unele îmbunătăţiri ar putea fi implementate.

Pentru că există foarte multe aplicaţii pentru crearea şi configurarea de maşini virtuale o, precum VMWare Player, Virtual Iron, Microsoft Virtual Server, primă îmbunătăţire o constituie extinderea posibilităţilor de virtualizare pe care aplicaţia să le folosească pentru rularea de experimente. Un pas în plus ar fi integrarea cu hypervizoare precum ESXi - dezvoltat de compania VMWare - deoarece această soluţie de virtualizare este folosită în special de către companii, astfel aplicaţia putând avea o prezentă mai bună în mediul corporate.

O altă îmbunătăţire o poate constitui extinderea comportamentului urmărit în interiorul maşinii virtuale. De exemplu, monitorizarea traficului de reţea, identificarea indicatorilor de compromis pentru o mai bună detecţie, analiza apelurilor către funcţiile de sistem efectuate de fişierul maliţios sunt doar câteva din utilitarele de monitorizare care pot fi folosite în analiza dinamică.

Unele tipuri de malware încearcă să determine dacă sunt examinate prin verificarea proceselor care rulează şi determinarea utilitarelor care ar putea realiza acest lucru; astfel comportamentul malware-ului nefiind arătat. De aceea, o tehnică prin care se poate evita acest impediment îl constituie schimbarea numelor/iconiţelor utilitarelor de monitorizare pentru a face verificarea proceselor mult mai dificilă. În unele cazuri, există posibilitatea ca fişierul maliţios să oprească procesul care l-a creat sau procesele care monitorizează activitatea, astfel extragerea rezultatelor în urma experimentului nefiind posibilă sau fiind inutilă.

În capitolul Malware am vorbit despre *Rootkit*. Deoarece acest malware va ascunde informaţiilor referitoare la modificările pe care fişierul maliţios le-a făcut, examinarea discurilor virtuale din afara maşinii de test ar putea identifica fişiere sau date ascunse, care ar fi fost greu sau chiar imposibil de descoperit.

Capitolul Implementare, ne recomandă ca la crearea maşinii virtuale de test să dezactivăm conexiunea la Internet. Din nefericire, multe fişiere maliţioase încearcă să verifice dacă au acces la Internet sau să se conecteze la un calculator la distantă pentru a primii comenzi pentru continuarea execuţiei. Astfel, o primă soluţie pentru această problemă poate fi simularea unei reţele virtuale şi a unor servicii precum DNS, servere FTP pentru comunicare astfel încât malware-ul să îşi desfăşoare activitatea maliţioasă. O altă abordare a acestei probleme o poate constitui permiterea accesului la Internet într-un mod controlat, dar astfel încât malware-ul să nu poată infecta alte calculatoare din reţeaua la care este conectată.

6. Concluzii

Pentru ca specialiştii din domeniul securităţii să poată face faţă numărului mare de ameninţări cu malware care apar în fiecare zi, este nevoie de folosirea resurselor disponibile într-un mod eficient. Analiza automată de malware este o necesitate în verificarea acestor fişiere, deoarece în urma analizei automate este nevoie de intervenţia unui specialist din domeniul doar în cazuri speciale.

Cele două metode de analiză a unui fişier maliţios, analiza statică şi dinamică, pot ajuta un specialist să verifice şi categorizeze ameninţările de natură cibernetică care se ivesc zi de zi. În anumite situaţii, analiza dinamică este mult mai eficientă decât cea statică, deoarece aceasta poate detecta şi analiza comportamentul pe care fişierul maliţios îl desfăşoară în interiorul unei maşini virtuale.

Aplicaţia dezvoltată în această lucrare efectuează monitorizarea dinamică a comportamentului unui malware cu ajutorul utilitarelor dezvoltate: aplicaţie pentru verificarea modificărilor aduse cheilor de regiştri, cât şi aplicaţia de monitorizare a fişierelor de pe maşina atacată. Un raport este generat la finalul fiecărui experiment, raport pe care utilizatorul îl poate vizualiza în aplicaţia web.

7. Bibliografie

[1] Bert Rankin: A Brief History of Malware — Its Evolution and Impact, https://www.lastline.com/blog/history-of-malware-its-evolution-and-impact/

[2] CrowdStrike: Falcon Sandbox FAQ, https://www.crowdstrike.com/endpoint-security-products/falcon-sandbox-malware-analysis/falcon-sandbox-faq/

[3] Egele M., Scholte T., Kirda E., and Kruegel C: A survey on automated dynamic malware-analysis techniques and tools. ACM Comput., 2012

[4] Ekta G., Divya B., Sanjeev S: Malware Analysis and Classification: A Survey, PEC University of Technology, Chandigarh - India, 2014

[5] Fred Cohen: Computer Viruses - Theory and Experiments http://all.net/books/virus/index.html

[6] Gongjun Yan: Providing Location Security in Vehicular Ad Hoc Networks, Old Dominion University, 2010

[7] Hybrid Analysis: Frequently Asked Questions (FAQ) https://www.hybrid-analysis.com/faq

[8] John Brunner: The Shockwave Rider, Harper & Row, United Kingdom, 1975

[9] John Von Neumann, Arthur W. Burks: Theory of Self-Reproducing Automata, University of Illinois Press, Urbana and London, 1966

[10] Lawrence A.: SNDBOX - an AI Powered Malware Analysis Site is Launched www.bleepingcomputer.com/news/security/sndbox-an-ai-powered-malware-analysis-site-is-launched/

[11] M.U Farroq, Muhammad Waseem, Anjum Khairi, Sadia Mazhar: A Critical Analysis on the Security Concerns of Internet of Things (IoT), International Journal of Computer Applications, 2015

[12] Mary Landesman: What Is Antivirus Software? https://www.lifewire.com/what-is-antivirus-software-152947

[13] Nate Lord: What is Cyber Security? Definition, Best Practices & More https://digitalguardian.com/blog/what-cyber-security

[14] Newsapexs: SNDBOX: AI-POWERED ONLINE AUTOMATED MALWARE ANALYSIS PLATFORM
https://newsapexs.com/sndbox-ai-powered-online-automated-malware-analysis-platform/
[15] Nikola Milošević: History of malware
https://arxiv.org/pdf/1302.5392.pdf
[16] Palo Alto Networks: WHAT IS CYBERSECURITY?
https://www.paloaltonetworks.com/cyberpedia/what-is-cyber-security
[17] Rajesh Kumar Goutam: Importance of Cyber Security, University of Lucknow, Lucknow, 2015
[18] Rayne Reid, Johan Van Niekerk: From Information Security to Cyber Security Cultures; Organizations to Societies, Nelson Mandela Metropolitan University, 2014
[19] Rossouw von Solms, Johan van Niekerk: From information security to cyber security, Nelson Mandela Metropolitan University, Elsevier, 2013
[20] Savan Gadhiya, Kaushal Bhavsar: Techniques for Malware Analysis
https://pdfs.semanticscholar.org/21e4/44fe5ddb9d48777faa53c3fa7d52002cb204.pdf
[21] Tim Fisher: What Is Malware and What Can It Do?
https://www.lifewire.com/what-is-malware-2625933
[22] TrustedSec: Malware Analysis is for the (Cuckoo) Birds
https://www.trustedsec.com/2018/05/malware-cuckoo-1/
[23] VirusTotal: How it works
https://support.virustotal.com/hc/en-us/articles/115002126889-How-it-works
[24] Whitman ME, Mattord HJ: Principles of information security, Thompson Course Technology, 2009
[25] Wikipedia: Advanced persistent threat
https://en.wikipedia.org/wiki/Advanced_persistent_threat
[26] Wikipedia: Botnet
https://en.wikipedia.org/wiki/Botnet
[27] Wikipedia: Computer virus
https://en.wikipedia.org/wiki/Computer_virus
[28] Wikipedia: Computer worm
https://en.wikipedia.org/wiki/Computer_worm

[29] Wikipedia: Cyber spying
https://en.wikipedia.org/wiki/Cyber_spying
[30] Wikipedia: Cyberterrorism
https://en.wikipedia.org/wiki/Cyberterrorism
[31] Wikipedia: Cyberwarfare
https://en.wikipedia.org/wiki/Cyberwarfare
[32] Wikipedia: Malware
https://en.wikipedia.org/wiki/Malware
[33] Wikipedia: Malware analysis
https://en.wikipedia.org/wiki/Malware_analysis
[34] Wikipedia: Ransomware
https://en.wikipedia.org/wiki/Ransomware
[35] Wikipedia: Rootkit
https://en.wikipedia.org/wiki/Rootkit
[36] Wikipedia: Trojan horse (computing)
https://en.wikipedia.org/wiki/Trojan_horse_%28computing%29
[37] Wikipedia: VirusTotal
https://en.wikipedia.org/wiki/VirusTotal