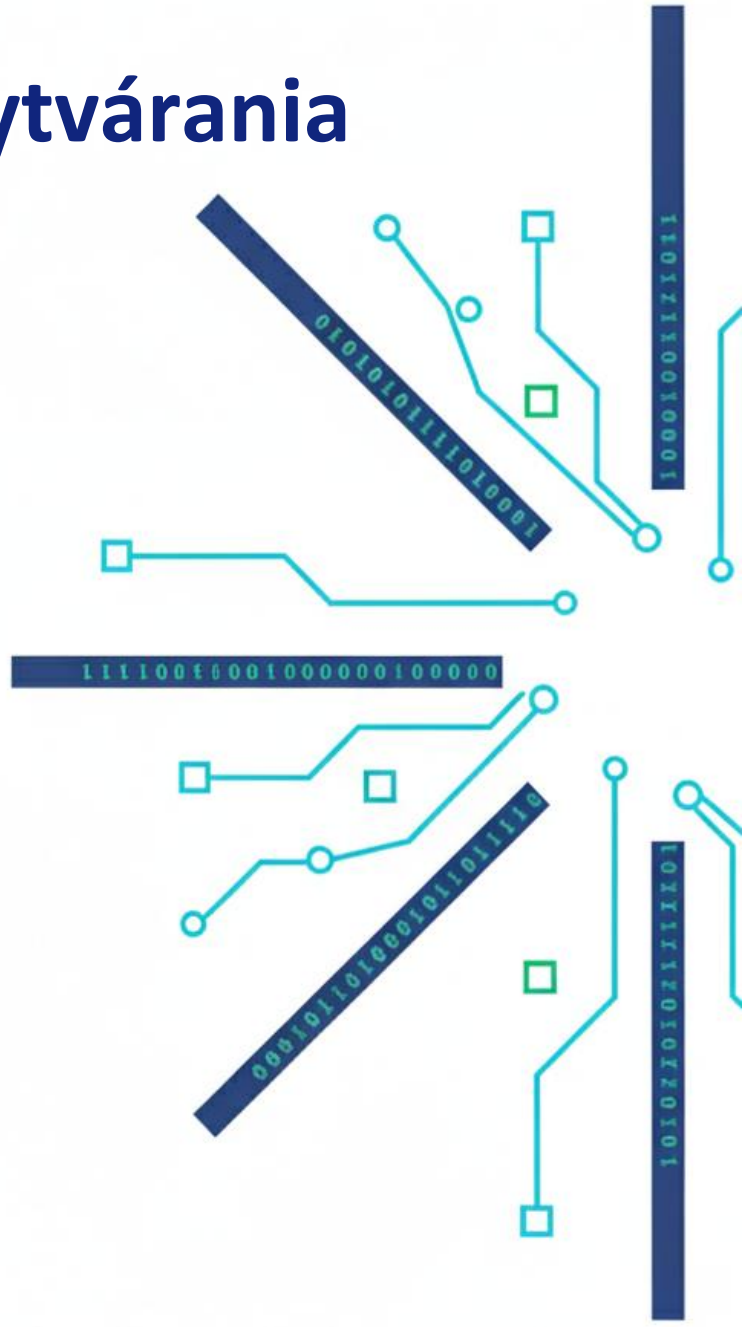
D13 – Metóda vytvárania datasetov



Projekt Automatizácia digitálnej forenzie a odpovede na incidenty (ADFIR) financovaný Európskou úniou – Next GenerationEU prostredníctvom Plánu obnovy a odolnosti Slovenskej republiky pod číslom projektu č. 09-I05-03-V02-00079.

Osnova

1 Popis projektu

Projekt **Automatizácia digitálnej forenzie a odpovede na incidenty** (ďalej len **„ADFIR“**) je financovaný **Európskou úniou – Next GenerationEU prostredníctvom Plánu obnovy a odolnosti Slovenskej republiky** pod číslom projektu č. 09-I05-03-V02-00079. Tento projekt sa zaoberá jednou z kľúčových výziev v oblasti kybernetickej bezpečnosti a informačnej bezpečnosti – ako spracovať obrovské množstvo digitálnych dôkazov, ktoré vznikajú počas incidentov kybernetickej bezpečnosti alebo forenzných vyšetrovaní. V súčasnosti je tento proces veľmi náročný z hľadiska ľudských zdrojov a času. Automatizácia pomocou metód strojového učenia môže preto výrazne **zlepšiť kvalitu digitálnej forenznej analýzy** a skrátiť čas potrebný na jej vykonanie. Celkovo to umožňuje bezpečnostným tímom efektívnejšie reagovať na kybernetické hrozby. Hlavné prínosy tohto projektu sú:

- **Urýchlené riešenie incidentov v oblasti kybernetickej bezpečnosti**. Projekt ADFIR zavádza automatizované prístupy k zberu, spracovaniu a analýze digitálnych stôp. V dôsledku toho môžu bezpečnostné tímy rýchlejšie identifikovať príčiny incidentov a prijať účinné opatrenia na ich riešenie.
- **Zníženie pracovnej záťaže forenzných analytikov**. Rutinné a časovo náročné úlohy spojené so spracovaním digitálnych stôp budú nahradené automatizovanými metódami. To umožní analytikom sústrediť sa na zložitejšie prípady a strategické rozhodovanie.
- **Vyššia kvalita a konzistentnosť výstupov**. Použitie jednotných metodík a nástrojov zaručuje, že spracované digitálne stopy budú presnejšie, konzistentnejšie a ľahšie overiteľné. To výrazne znižuje riziko chýb spôsobených ľudskými faktormi.
- **Možné využitie v trestnom konaní.** Výstupy projektu budú vyvinuté v súlade s právnymi požiadavkami a normami, čo umožní, aby boli digitálne stopy akceptované ako relevantné dôkazy pre vyšetrovanie a súdne konania.

2 Úvod

Vývoj metód strojového učenia a pokročilá analýza údajov má tiež významný vplyv na oblasť digitálnej forenznej analýzy. Automatizované spracovanie veľkého množstva forenzných údajov, identifikácia vzorov správania, korelácia udalostí a detekcia anomálií však vyžadujú vysoko kvalitné, dobre štruktúrované a reprezentatívne súbory údajov. Spôsob, akým sa tieto súbory údajov vytvárajú, má zásadný vplyv na použiteľnosť a spoľahlivosť výsledných modelov.

Vývoj metód strojového učenia a pokročilá analýza údajov majú tiež významný vplyv na oblasť digitálnej forenznej analýzy. Automatizované spracovanie veľkého množstva forenzných údajov, identifikácia vzorov správania, korelácia udalostí a detekcia anomálií však vyžadujú vysoko kvalitné, dobre štruktúrované a reprezentatívne súbory údajov. Spôsob, akým sa tieto súbory údajov vytvárajú, má zásadný vplyv na použiteľnosť a spoľahlivosť výsledných modelov.

Jednou z hlavných výziev v oblasti digitálnej forenznej analýzy je **nedostatok kvalitných dátových súborov**, ktoré spĺňajú niekoľko kľúčových požiadaviek, ako je dostupnosť komplexných prípadových štúdií a dostatočný počet relevantných digitálnych artefaktov [1,2]. Tento problém je zdôraznený v niekoľkých vedeckých publikáciách, ktoré poukazujú na potrebu dobre štruktúrovaných dátových súborov vhodných na forenzné účely a zároveň poukazujú **na celkový nedostatok takýchto zdrojov** v tejto oblasti [1,2,3].

Z hľadiska učenia modelov a aplikácie strojového učenia v digitálnej forenznej analýze možno identifikovať tri základné typy dátových súborov:

1. **Reálne súbory údajov** získané priamo z riešenia skutočných bezpečnostných incidentov,
2. **Vytvorené súbory údajov zo simulovaných útokov a techník útočníkov**, vykonávané v kontrolovanom prostredí,
3. **Databázy pochádzajúce zo súťaží CTF (Capture The Flag)**, v ktorých sa simulujú útoky a následne sa pripravujú analytické úlohy a otázky.

Každý z týchto prístupov má svoje výhody, obmedzenia a špecifiká, ktoré je potrebné zohľadniť pri navrhovaní metodiky výskumu a interpretácii výsledkov.

2.1 Reálne dátové súbory zo skutočných bezpečnostných incidentov

Reálne údaje zo skutočných bezpečnostných incidentov sú z hľadiska autenticity a realizmu **najcennejším zdrojom** pre digitálnu forenznú analýzu. Zachytávajú skutočné správanie útočníkov, reálne chyby konfigurácie systému, nepredvídateľné kombinácie udalostí a komplexnú dynamiku incidentov v produkčnom prostredí. Z tohto dôvodu poskytujú ideálny základ pre vývoj a testovanie forenzných analytických metód.

Používanie takýchto údajov však prináša **významné problémy a obmedzenia**. Reálne dátové súbory často pokrývajú len **obmedzený rozsah techník, taktík a postupov útočníkov (TTP)**, ktoré sa vyskytli v danom bezpečnostnom incidente. Nie sú teda reprezentatívnym prierezom

celého spektra možných útokov, ale skôr špecifickým a kontextovo viazaným prípadom. Okrem toho sa reálne bezpečnostné incidenty často líšia v kvalite a úplnosti dostupných artefaktov, pretože údaje môžu byť poškodené, vymazané alebo neúplné.

Najväčšou prekážkou je však **praktická nemožnosť zdieľania týchto údajov**. Tieto údaje sú citlivé a môžu obsahovať osobné údaje a iné citlivé informácie (napr. duševné vlastníctvo, obchodné tajomstvá) a ich použitie je obmedzené legislatívnymi rámcami, ochranou osobných údajov, obchodnými tajomstvami a internými bezpečnostnými politikami organizácií. Z tohto dôvodu je ich použitie na akademický výskum a vývoj, replikovateľnosť experimentov a porovnateľnosť výsledkov výrazne obmedzené. Potvrdzuje to aj článok [4], v ktorom autori vo svojich zisteniach poukazujú na to, že pri vytváraní súboru údajov je dôležité zabezpečiť, aby boli súbory údajov nielen správne vytvorené, ale aby aj odrážali scenáre, ktoré sa môžu vyskytnúť v reálnych situáciách. Zdôrazňujú tiež, že vytváranie a zdieľanie súborov údajov môže byť náročné z dôvodu právnych obmedzení alebo správy a ochrany údajov. To isté uvádzajú aj Breitinger a Jotterand [3], ktorí dospeli k záveru, že vytváranie a zdieľanie dátových súborov je nevyhnutné pre pokrok a umožnenie porovnania výsledkov. Zdôraznili tiež potrebu opatrnosti v súvislosti s konkrétnymi zákonmi, ako sú autorské práva alebo licencovanie.

2.2 Databázy zo simulovaných útokov a techník útočníkov

Druhým prístupom k vytváraniu dátových súborov je **cielená simulácia útokov a techník útočníkov** v kontrolovanom a izolovanom prostredí. V tomto prípade sa útoky vykonávajú vedome a systematicky. Útoky sa často vykonávajú na základe známych rámcov, ako je MITRE ATT&CK, s cieľom generovať forenzné artefakty zodpovedajúce konkrétnym technikám a fázam útoku.

Hlavnou výhodou tohto prístupu je **kontrola nad scenárom**. Výskumník alebo vývojár presne vie, ktoré techniky boli použité, v akom poradí a na aký účel. To umožňuje presné označovanie údajov, vytváranie vyvážených dátových súborov a systematické testovanie schopnosti modelov detekovať konkrétne typy správania. Simulované dátové súbory sú vhodné aj na vytváranie referenčných údajov pre experimentálne porovnania a validačné štúdie.

Na druhej strane majú simulované útoky aj svoje obmedzenia. Aj pri vysokej úrovni odbornosti môže byť simulácia **zjednodušená** a nemusí zachytiť všetky nepredvídateľné aspekty skutočných incidentov, ako sú chyby útočníkov, kombinované útoky viacerých aktérov alebo dlhodobé kampane s nízkou intenzitou. Simulované útoky však predstavujú dôležitý kompromis medzi realizmom a kontrolovateľnosťou údajov.

2.3 Databázy zo súťaží CTF

Tretia kategória pozostáva z **dátových súborov zo súťaží CTF (Capture the Flag)**, ktoré sa už dlho používajú vo vzdelávaní a odbornej príprave v oblasti kybernetickej bezpečnosti. Úlohy CTF sú zvyčajne navrhnuté tak, aby simulovali konkrétny bezpečnostný incident alebo jeho časť, a účastníci odpovedajú na vopred definované otázky analýzou digitálnych stôp.

Používanie dátových súborov CTF má určité **vnútorné obmedzenia**. Scenáre CTF sú často umelo vytvorené, zamerané na konkrétny typ útoku a zvyčajne ich vykonáva jediný útočník. Takýto model nemusí plne odrážať realitu, kde môže byť zapojených viacero útočníkov, môžu sa odohrávať viacfázové útoky a výsledok vyšetrovania nie je vopred známy. Okrem toho údaje CTF implicitne predpokladajú existenciu riešenia, zatiaľ čo bezpečnostné incidenty v reálnom svete sa vyznačujú vysokou mierou neistoty a nejednoznačnosti.

Napriek týmto obmedzeniam majú súťaže CTF **významné paralely so skutočnými bezpečnostnými incidentmi**. Aj v scenároch CTF útočníci používajú špecializované nástroje, techniky a taktiky, ktoré sú často identické alebo veľmi podobné tým, ktoré sa používajú v praxi. Proces skúmania údajov a digitálnych stôp, korelácie artefaktov a rekonštrukcie udalostí je porovnateľný so skutočnou digitálnou forenznou analýzou.

Hoci sú digitálne stopy v úlohách CTF umelo vytvorené, samotný **proces analýzy údajov a extrakcie forenzných artefaktov** je do veľkej miery rovnaký ako pri vyšetrovaní reálnych incidentov. Okrem toho štruktúrovaná povaha úloh CTF, často založená na otázkach a odpovediach, môže podporovať systematický analytický prístup, ktorý je prenosný do reálneho sveta a môže viesť k rýchlejšej identifikácii a riešeniu problémov.

2.4 Motivácia pre výber zdrojov údajov

Vzhľadom na vyššie uvedené skutočnosti sme sa rozhodli zamerať túto štúdiu predovšetkým na **súbory údajov vytvorené na základe simulovaných techník útočníkov a súbory údajov pochádzajúce zo súťaží CTF**.
Hlavným dôvodom je obmedzená dostupnosť údajov zo skutočných bezpečnostných incidentov a praktická nemožnosť ich zdieľania v akademickom a výskumnom prostredí.

Zameranie na **diskové obrazy operačného systému Windows** vychádza z rozsahu a zložitosti digitálnej forenznej analýzy ako disciplíny. Jednotlivé subdomény, ako napríklad sieťová forenzná analýza, forenzná analýza Windows, Linux alebo mobilných zariadení, sa výrazne líšia v štruktúre údajov, používaných nástrojoch a analytických postupoch. Každá z týchto oblastí vyžaduje špecifické metódy predspracovania a analýzy údajov, čo znemožňuje univerzálny prístup.

Pomocou simulovaných útokov a dátových súborov CTF v prostredí Windows je možné vytvoriť **reprodukovateľné, zdieľateľné a metodicky kontrolovateľné dátové súbory,** ktoré poskytujú vhodný základ pre výskum a vývoj metód strojového učenia v digitálnej forenznej analýze.

3 Metóda pre vytváranie forenzných dátových súborov na základe techník útočníkov

Na účely školenia a validácie modelov strojového učenia (ML) určených na automatizáciu forenznej analýzy bol vytvorený dátový súbor, ktorý verne odráža správanie moderných útočníkov v fáze po zneužití. Na rozdiel od syntetických dátových súborov generovaných náhodne sme v projekte ADFIR použili kombináciu automatizovanej simulácie a manuálneho vykonávania pokročilých ofenzívnych TTP (taktiky, techniky a postupy – TTP).

Zber údajov sa vykonával na virtualizovanej pracovnej stanici s označením RD05W-W11-1, ktorá beží na systéme Windows 11 (verzia 23H2) a na testovacie účely hostuje agenta Cymulate. Základná infraštruktúra využíva hypervízor VMware ESXi. Testovacie prostredie simuluje štandardnú štruktúru firemnej domény, kde koncový bod spravujú dva redundantné doménové radiče so systémom Windows Server 2022 Standard.

Na generovanie syntetických, ale realistických simulácií útokov bola v experimentálnom prostredí použitá platforma Cymulate [5]. Cymulate je nástroj na simuláciu narušení a útokov (BAS), ktorý je navrhnutý na nepretržité overovanie kybernetickej odolnosti prostredníctvom automatizovaných simulácií útokov.

Na rozdiel od tradičného penetračného testovania umožňuje Cymulate vykonávať komplexné scenáre útokov v kontrolovanom prostredí bez narušenia prevádzky. Pre tento výskum je kľúčové, že platforma generuje vysoko verné digitálne stopy na úrovni operačného systému aj siete.

Na účely výskumu boli nevyhnutné nasledujúce funkcie platformy:

- Zosúladenie s MITRE ATT&CK®: Cymulate simuluje taktiky, techniky a postupy (TTP) reálnych protivníkov zmapovaných v rámci MITRE ATT&CK. Tým sa zabezpečuje, že generované údaje odrážajú aktuálne trendy v oblasti kybernetických hrozieb (napr. skupiny APT, ransomware a pokročilé trójske kone).
- Generovanie forenzných artefaktov: Hoci sú útoky simulované, interakcia s operačným systémom je autentická. Vykonanie týchto simulácií zanecháva špecifické stopy potrebné na trénovanie forenzného automatizačného nástroja, vrátane:
    - Záznamy v systémových protokoloch (protokoly udalostí Windows).
    - Úpravy registrov systému Windows a súborového systému.
    - Zvyškových údajov vo volatilnej pamäti (RAM).
    - Vzory sieťovej komunikácie (C2 prevádzka).

3.1 Vytvorenie dátového súboru

Na vytvorenie súboru údajov vhodného na trénovanie modelov strojového učenia je najlepšie zaviesť kontrolovaný proces generovania zameraný na produkciu realistických forenzných artefaktov. Vstupom do tejto fázy je čisté, inštrumentované virtuálne prostredie, zatiaľ čo

výstupom sú surové forenzné obrazy diskov a výpisy pamäte obsahujúce vzory neškodných aj škodlivých aktivít.

Proces generovania údajov bol navrhnutý tak, aby simuloval kompletný životný cyklus kybernetického útoku, a bol vykonaný prostredníctvom nasledujúcich metodických krokov:

- koordinácia viacfázových scenárov útokov,
- vstrekovanie pokročilých techník úniku,
- časové anotácie (generovanie základných údajov) a
- získavanie forenzných údajov.

Koordinácia viacfázových scenárov útoku: Jadro dátového súboru bolo generované vykonaním pokročilých scenárov útoku, ktoré napodobňujú taktiky, techniky a postupy (TTP) sofistikovaných aktérov hrozieb. Simulácia sledovala lineárny postup kill-chain, od počiatočného prístupu (napr. doručenie nákladu prostredníctvom downloaderov) až po vykonanie, vytvorenie perzistencie, eskaláciu privilégií a prístup k povereniam. Tento štruktúrovaný prístup zabezpečuje, že súbor údajov zachytáva sekvenčné závislosti a kauzálne vzťahy medzi rôznymi forenznými artefaktmi.

Vstrekovanie pokročilých techník úniku: Aby simulácia odrážala zložitosť moderných hrozieb, zahŕňala techniky špeciálne navrhnuté na minimalizáciu forenzných stôp a obídenie statickej detekcie. Patrilo sem použitie binárnych súborov „Living off the Land“ (LotL), metód vykonávania bez súborov (napr. reflexné vstrekovanie DLL, načítanie zostavy .NET v pamäti) a zamlžovanie argumentov príkazového riadku. Tieto techniky zabezpečujú, že výsledný súbor údajov preveruje detekčné schopnosti modelu nad rámec jednoduchého porovnávania podpisov.

Časové anotácie (generovanie základných údajov): Pre proces riadeného učenia bolo kľúčové zaznamenávanie presných časových značiek pre každú jednotlivú fázu simulácie útoku. Tieto časové okná tvoria „základné údaje“, ktoré definujú presné obdobia, počas ktorých boli do systému zavedené škodlivé artefakty. Tieto anotácie sa následne používajú v procese spracovania údajov na označenie segmentov superčasovej osi, čo umožňuje výpočet metrík presnosti a spätného vyhľadávania počas hodnotenia modelu.

Získavanie forenzných údajov: Po dokončení simulačných scenárov bol stav virtuálneho prostredia zmrazený a bolo vykonané úplné forenzné získavanie údajov. Tento proces priniesol surový bitový obraz nevolateľného úložiska (diskový obraz). Tieto zdroje surových údajov slúžia ako primárny vstup pre rámec forenznej extrakcie (napr. Athena), ktorý spája fázu fyzickej simulácie s fázou logického spracovania údajov opísanou v nasledujúcej časti.

3.2 Techniky útoku použité pri vytváraní súboru údajov

Základná vrstva sieťovej a systémovej aktivity bola generovaná pomocou platformy Cymulate, ktorá bola použitá na vytvorenie realistického pozadia a simuláciu štandardných vektorov križovania.

V rámci nástroja Cymulate bol vykonaný pokročilý scenár simulujúci aktivity skupiny APT-19.

3.2.1 Fázy prípravy útoku APT-19

Pokročilý scenár APT-19 začína vykonaním na úrovni služby: útočník nainštaluje a spustí službu založenú na PowerShell, pripraví užitočné zaťaženie pomocou Invoke-WebRequest a spustí kód PowerShell zakódovaný v Base64, aby sa skryl.

- **Sťahovač: Invoke-WebRequest (PowerShell)**
  Táto simulácia útoku stiahne súbor z určenej adresy URL do konkrétnej cesty v počítači pomocou cmdletu Invoke-WebRequest v PowerShell. Tento cmdlet umožňuje používateľovi načítať obsah webovej stránky alebo súboru z webového servera a môže sa použiť na stiahnutie súboru zo vzdialeného servera do lokálneho počítača.

- **Vykonanie príkazu PowerShell zakódovaného v Base64**
  Táto simulácia útoku vytvorí škodlivý kód PowerShell, ktorý je zakódovaný pomocou Base64. Tento typ útoku bežne používajú útočníci na získanie prístupu do systému. Po úspešnom vykonaní kód PowerShell vykoná zakódovaný príkaz, ktorý vypíše správu „Hey, Atomic!“ do štandardného výstupu.

- **Vykonanie príkazu ako služby**
  Táto simulácia útoku vytvorí službu, ktorá umožňuje vykonanie akéhokoľvek príkazu. Keď sa pokúsite spustiť príkaz, napríklad PowerShell, služba nahlási neúspešné dokončenie, aj keď je kód vykonaný správne. Ak je príkaz úspešný, príkaz sc.exe create vytvorí novú službu a Powershell.exe vytvorí nový súbor s názvom art-marker.txt.

- **Vstrekovanie procesu: Reflektívne vstrekovanie DLL**
  Reflexné vstrekovanie DLL je metóda útoku na systém vstrekovaním škodlivého kódu do bežiaceho procesu. Táto technika umožňuje útočníkovi vstreknúť dynamickú knižnicu (DLL) do cieľového procesu bez potreby najprv zapísať DLL na disk. Namiesto toho sa DLL načíta priamo z pamäte, čím sa obíde tradičné bezpečnostné mechanizmy, ktoré môžu byť nasadené. Po vložení škodlivej DLL sa dá použiť na vykonanie škodlivého kódu alebo získanie prístupu k citlivým informáciám.

- **Inštalácia služby pomocou PowerShell**
  Táto simulácia útoku inštaluje lokálnu službu pomocou PowerShell. Po úspešnom spustení PowerShell stiahne súbor AtomicService.exe z Github. Potom použije príkazy New-Service a Start-Service na spustenie služby.

- **Skryté okno**
  Táto simulácia útoku spustí skryté okno PowerShell, ktoré vykoná spustiteľný súbor bez vedomia používateľa. To sa dosiahne odovzdaním argumentu -WindowStyle

Hidden príkazu PowerShell, čím sa parameter WindowStyle nastaví na skrytý. Tým sa okno PowerShell a súvisiace procesy efektívne skryjú pred používateľom.

- **Úprava registru SecurityHealth pomocou CMD**
  Táto simulácia útoku zahŕňa úpravu kľúča RUN v registri lokálneho počítača, aby sa spustiteľný súbor programu Windows Defender spúšťal pri štarte systému. To je možné vykonať len vtedy, ak je príkazový riadok (CMD) spustený s oprávneniami správcu.

- **Úprava registra: Skrytie prípon súborov**
  Tento príkaz nastaví položku registra pod HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced, ktorá nariadi programu Windows Explorer skryť prípony všetkých súborov.

- **Kľúče spúšťania registra: Perzistencia**
  Táto simulácia útoku vytvorí nové kľúče registra na miestach, ktoré sa bežne používajú na udržanie perzistencie (trvalej prítomnosti), napríklad v registri systému Windows. Hodnoty týchto kľúčov sa potom nastavia tak, aby odkazovali na užitočné dáta, ktoré chce útočník spustiť. To umožňuje, aby užitočné dáta zostali trvalo prítomné v systéme aj po jeho reštartovaní.

- **Tok vykonania únosu: Zachytenie cesty (PATH)**
  Tento test simulácie útoku stiahne škodlivý binárny súbor do určenej cesty. Následne nastaví premennú prostredia PATH tak, aby presmerovala alias (net.exe) na cestu k škodlivému binárnemu súboru. Tým sa nastaví nová cesta s vyššou prioritou ako akákoľvek iná cesta v hodnote PATH. Pri použití príkazu net sa namiesto pôvodného binárneho súboru spustí škodlivý binárny súbor. Predvolený binárny súbor by mal vygenerovať reťazec „This is a test binary by Cymulate“ (Toto je testovací binárny súbor od Cymulate).

- **LOLBIN Download – Vzdialené spustenie s ADS**
  Interpretátor príkazového riadku v systéme Windows pridáva obsah do alternatívneho dátového toku (ADS). Príklad škodlivého použitia: Môže sa použiť na obídenie obranných protiopatrení alebo na skrytie ako mechanizmus perzistencie.

3.2.2 Fázy vykonania útoku APT-19

Do tohto prostredia boli potom ručne vložené špecifické sekvencie vykonávania zamerané na perzistenciu, eskaláciu privilégií a krádež identity (prístup k povereniam/krádež poverení). Scenár útoku prebiehal v nasledujúcich fázach:

1. **Perzistencia:** Na udržanie trvalého prístupu bol použitý SharPersist (súbor nástrojov C# na perzistenciu v systéme Windows), ktorý bol v tomto prípade spustený z C:\Windows\Tasks\SharPersist.exe. Pomocou tohto nástroja boli vytvorené

mechanizmy perzistencie pomocou naplánovaných úloh, služieb a kľúčov registru. Vytvorená perzistencia zabezpečila prítomnosť zadných vrátok, umiestnených v C:\Windows\Tasks\mssvc\mssvc.exe.

2. **Vymenovanie prostredia a eskalácia privilégií:** Po zabezpečení trvalého prístupu bolo vykonané vyšetrovanie zraniteľnosti systému. Na tento účel boli použité dva doplnkové nástroje:

   - **SharpUp** (C:\Windows\Tasks\SharpUp.exe): Implementácia C# navrhnutá na rýchlu identifikáciu konfiguračných chýb (napr. modifikovateľných služieb), ktoré umožňujú eskaláciu na úroveň správcu.
   - **winPEAS.ps1** (C:\Windows\Tasks\winPEAS.ps1): komplexný skript PowerShell, ktorý vykonal hlbokú analýzu systému. Tento nástroj generuje značný počet artefaktov (čítanie súborov, prístup k registrom), čím vytvára špecifický „hlučný“ vzor v dátovom súbore.

   Oba nástroje vykonávajú rozsiahlu sadu kontrol (napr. konfigurácie služieb, prístupové práva, registre, súborový systém) pri bežnom používaní, čo sa zvyčajne prejavuje zvýšeným počtom systémových dotazov a „hlučnejším“ profilom v protokoloch v porovnaní s minimalistickým ručným výpočtom.

3. **Prístup k povereniam:** SafetyKatz a SharpKatz sa používali na extrakciu autentifikačných materiálov (kľúče/tiket NTLM a Kerberos) z procesu LSASS.

   - C:\Windows\Tasks\SafetyKatz.exe – Nástroj C# zo súboru GhostPack, ktorý funguje ako **obal okolo známeho hackerského nástroja Mimikatz.** SafetyKatz vytvorí minidump pamäte procesu LSASS pomocou funkcie **MiniDumpWriteDump** do *súboru C:\Windows\Temp\debug.bin*. Následne **načíta upravený Mimikatz** prostredníctvom PELoader, vykoná sekurlsa::logonpasswords a sekurlsa::ekeys nad dumpom a po dokončení súbor odstráni.

   - C:\Windows\Tasks\SharpKatz.exe – port C# vybraných príkazov Mimikatz.

4. **Obchádzanie ochrany:** Nástroj SharpKiller (port „AMSI-Killer“ do .NET) bol použitý na obídenie AMSI (AntiMalware Scan Interface) opravou kontroly v pamäti, aby AMSI neblokoval škodlivý obsah.
   - C:\Windows\Tasks\SharpKiller.exe

3.3 Hrozby

Vytvorená dátová sada je relevantná pre súčasný výskum v oblasti kybernetickej bezpečnosti, pretože odráža posun v paradigme útokov od tradičných spustiteľných súborov k technikám označovaným ako **„Living off the Land“ (LotL)** a **„.NET Tradecraft“.**

3.3.1 Detekcia „bezsúborových“ a .NET hrozieb

Tradičné antivírusové systémy sa dlho zameriavali na podpisy súborov na disku. Nástroje použité v našej dátovej sade (SharpUp, SharpKatz, SharPersist) sa často načítajú priamo do pamäte prostredníctvom *techniky Reflective DLL Injection* alebo *Assembly Loading*, čím sa minimalizuje stopa na disku. Pre modely ML to predstavuje výzvu identifikovať anomálie v správaní procesov a volaniach API, namiesto toho, aby sa spoliehali na statické atribúty súborov. Dataset poskytuje trénovacie dáta na detekciu škodlivého kódu bežiaceho v kontexte legitímneho bežného jazykového prostredia (CLR).

3.3.2 Zneužitie dôveryhodných ciest a kamufláž

Umiestnenie nástrojov do adresára C:\Windows\Tasks nie je náhodné. Tento adresár často prehliadajú správcovia, pretože obsahuje veľa legitímnych systémových súborov a zároveň majú bežní používatelia v určitých konfiguráciách aj oprávnenie na zápis do neho. Kombinácia tohto miesta s názvami ako mssvc.exe simuluje sofistikované maskovanie, ktoré je typické pre skupiny Advanced *Persistent Threat (APT*). ML model vyškolený na týchto údajoch sa musí naučiť rozlišovať medzi legitímnym procesom a jeho falošným náprotivkom na základe kontextových metadát (rodičovský proces, čas spustenia, absencia digitálneho podpisu), nielen na základe názvu.

3.3.3 Obchádzanie moderných ochranných mechanizmov

Zahrnutie techniky obchádzania AMSI (Sharp-Killer) do súboru údajov je nevyhnutné na simuláciu sofistikovaného protivníka. Ak model ML nedokáže detekovať samotný akt deaktivácie telemetrie (AMSI), stáva sa slepým voči nasledujúcim fázam útoku. Tento súbor údajov vám umožňuje trénovať modely na detekciu takzvaných „pre-exploit“ aktivít, pri ktorých útočník manipuluje s integritou samotného monitorovacieho systému.

3.3.4 Praktická použiteľnosť

Popísané techniky v súčasnosti aktívne využívajú nielen štátom podporované subjekty, ale aj prevádzkovatelia ransomware ako služby (napr. skupiny Conti, LockBit), ktorí používajú nástroje ako *SharpUp* a varianty *Mimikatz* na laterálny pohyb v sieti pred zašifrovaním údajov. Výskum založený na tejto dátovej sade má preto priamy vplyv na zlepšenie detekčných schopností proti najrozšírenejším kybernetickým hrozbám súčasnosti.

4 Metóda vytvárania forenzných dátových súborov z CTF

Táto kapitola predstavuje možné zdroje údajov a metodiky na vytváranie dátových súborov pre digitálnu forenznú analýzu. Kapitola opisuje, aké typy zdrojov – vrátane súťaží CTF, portálov referenčných dátových súborov a syntetických bezpečnostných dátových súborov – sú vhodné na získanie relevantných artefaktov. Rovnako rozoberá dva prístupy k vytváraniu dátových súborov: použitie forenzných nástrojov a vytváranie dátových súborov pomocou vložených údajov.

4.1 Možné zdroje údajov

Ako už bolo spomenuté, **súťaže CTF** sú vhodným zdrojom na vytváranie dátových súborov pre digitálnu forenznú analýzu, keďže prístup k reálnym údajom z bezpečnostných incidentov môže byť problematický a obmedzený. Údaje z reálnych bezpečnostných incidentov často nie sú verejne dostupné a ich získanie môže byť náročné z dôvodu potreby autorizácie, právnych obmedzení alebo ochrany osobných údajov.

Na účely vytvárania dátových súborov sa preto používajú kombinované zdroje na simuláciu alebo priamy prístup k relevantným artefaktom a scenárom. Nasledujúce zdroje údajov sú praktické a osvedčené možnosti, ktoré sa bežne používajú vo výskume a výučbe digitálnej forenznej analýzy:

CTF a tréningové dátové súbory

- **DFIR CTF archív** – zbierka rôznych CTF úloh a dátových súborov pre DFIR školenia (obrazy diskov, artefakty) – jasne dostupná prostredníctvom portálov, ako sú DFIR Training / DFIR CTF archívy [6].
- **Digital Corpora** – voľne dostupná zbierka obrazov diskov, scenárov a artefaktov, vrátane rôznych scenárov CTF a scenára M57 Patents (obrazy diskov, pamäť, PCAP) [7].
- **Honeynet Project Forensic Challenges** – súťaže a forenzné výzvy (výzvy a balíky dátových súborov na precvičovanie forenzných techník) [8].

Portály a projekty referenčných dátových súborov

- **Projekt EVIDENCE** – európsky projekt zameraný na výmenu a štandardizáciu digitálnych dôkazov v praxi; podkladové a rámcové materiály pre prácu s reálnymi dôkazmi (prístup k údajom môže byť obmedzený) [9].

4.2 Vytváranie dátových súborov pomocou forenzných nástrojov

Dataset pre výskum v oblasti digitálnej forenznej analýzy je možné vytvoriť priamo pomocou špecializovaných forenzných nástrojov. Dáta extrahované z diskových obrazov jednotlivých scenárov CTF je možné spracovať pomocou nástroja **Kroll Artifact Parser and Extractor (KAPE)** [10]. KAPE je jedným z široko používaných nástrojov v oblasti digitálnej forenznej analýzy a umožňuje systematickú, opakovanú a automatizovanú extrakciu a analýzu forenzných artefaktov z operačného systému Windows. Vďaka podpore modulárnych profilov (cieľov a

modulov) je možné presne definovať, ktoré artefakty sa majú získať zo systému a ako sa majú spracovať do štruktúrovanej formy vhodnej na ďalšiu analýzu.

Okrem nástroja KAPE je možné na vytvorenie dátového súboru použiť aj samostatné forenzné nástroje zo sady **Eric Zimmerman Tools** [11]. Tieto nástroje sa v praxi často používajú buď priamo, alebo ako moduly analýzy integrované do KAPE. Umožňujú podrobnú analýzu konkrétnych typov forenzných artefaktov a generujú výstupy najčastejšie vo forme súborov CSV, ktoré sú vhodné na následné spracovanie pomocou metód analýzy dát a strojového učenia.

Proces vytvárania súboru údajov pomocou forenzných nástrojov zahŕňa niekoľko systematických krokov, ktoré zabezpečujú extrakciu, analýzu a vyhodnotenie forenzných údajov. Každý krok je navrhnutý tak, aby zachoval integritu pôvodných údajov a aby výsledný súbor údajov obsahoval analyticky relevantné informácie vhodné na ďalšiu digitálnu forenznú analýzu a experimenty s analýzou údajov alebo strojovým učením.

Na vytvorenie súboru údajov boli použité nasledujúce kroky:

1) pripojenie obrazu disku,
2) extrakcia forenzných artefaktov pomocou forenzných nástrojov (napr. KAPE),
3) parsovanie forenzných artefaktov,
4) čistenie a príprava výstupných údajov,
5) automatické a manuálne obohatenie výstupných údajov (označenie striebrom a zlatom)

V nasledujúcich podkapitolách poskytujeme podrobnejšie komentáre k jednotlivým častiam.

4.2.1 Pripojenie obrazu disku

Spracovanie údajov pozostávalo z viacerých krokov. V prvom kroku bol disk namontovaný v režime len na čítanie pomocou nástrojov na montáž diskových obrazov (napr. Arsenal Image Mounter [12]). Tento prístup zaručuje integritu pôvodných údajov a eliminuje riziko neúmyselnej modifikácie počas analýzy, čo je kľúčový princíp digitálnej forenznej metodiky.

4.2.2 Extrakcia forenzných artefaktov pomocou forenzných nástrojov

Po úspešnom pripojení diskového obrazu sa na pripojenom súborovom systéme spustí nástroj KAPE. Počas fázy extrakcie je možné použiť vstavané „cieľové objekty“, ktoré umožňujú cielenú extrakciu relevantných forenzných artefaktov. Konkrétne sa odporúča použiť cieľové objekty pre nasledujúce artefakty:

- $MFT (Master File Table),
- $J (USN Journal),
- SRUM (monitor využívania systémových zdrojov),
- Registry hives,
- Amcache,

- LNK súbory a zoznamy skokov,
- Súbory predbežného načítania,
- Protokoly udalostí systému Windows,
- koš a
- Časová os Windows.

Tieto artefakty sú dôležitými zdrojmi informácií o aktivite používateľov, spúšťaní aplikácií, operáciách so súbormi a udalostiach v systéme, a preto sú kľúčové pre rekonštrukciu incidentov a analýzu potenciálne škodlivých aktivít. Viac informácií o forenzných artefaktoch v operačnom systéme Windows nájdete v dodávke – D12 Metóda automatizovaného zberu digitálnych dôkazov.

4.2.3 Analýza forenzných artefaktov

Na vyššie uvedené forenzné artefakty je možné použiť nasledujúce moduly nástroja KAPE a príslušné nástroje od Erica Zimmermana:

- **AmcacheParser.mkape** – pomocou nástroja **AmcacheParser** [13].
- **AppCompatCacheParser.mkape** – použitie nástroja **AppCompatCacheParser** [14].
- **EvtxECmd.mkape** – použitie nástroja **EvtxECmd** [15].
- **JLECmd.mkape** – použitie nástroja **JLECmd** [16].
- **LECmd.mkape** – použitie nástroja **LECmd** [17].
- **MFTECmd.mkape** – použitie nástroja **MFTECmd** [18].
- **PECmd.mkape** – použitie nástroja **PECmd** [19].
- **RBCmd.mkape** – použitie nástroja **RBCmd** [20].
- **RECmd_AllBatchFiles.mkape** – pomocou nástroja **RECmd** (spracovanie všetkých dávkových súborov) [21].
- **SBECmd.mkape** – pomocou nástroja **SBECmd** [22].
- **WxTCmd.mkape** – pomocou nástroja **WxTCmd** [23].
- **SrumECmd.mkape** – pomocou nástroja **SrumECmd** [24].

Tieto parsery transformujú surové forenzné artefakty do štruktúrovanej podoby, čo má za následok vytvorenie súborov CSV, ktoré reprezentujú jednotlivé typy artefaktov a ich atribúty.

4.2.4 Čistenie a príprava výstupných údajov

Výstupné súbory sú súbory CSV obsahujúce vybrané forenzné artefakty, ktorých štruktúra a počet sa môžu líšiť v závislosti od:

- verzie operačného systému Windows,
- nastavení systému,
- počtu používateľských účtov (viac používateľov vedie k väčšiemu počtu výstupných súborov),
- dostupnosti konkrétnych mechanizmov (napr. SRUM nie je prítomný vo všetkých verziách operačného systému).

Konzolové protokoly, ktoré sú štandardne generované nástrojom KAPE počas behu, by mali byť z výstupov odstránené, pretože nepredstavujú analyticky relevantné údaje na účely vytvorenia súboru údajov. Odporúča sa tiež vylúčiť výstup parseru PECmd vo forme časovej osi, pretože štandardný výstup CSV tohto nástroja obsahuje podrobnejšie a analyticky hodnotnejšie informácie.

4.2.5 Automatické a manuálne obohacovanie výstupných údajov

Po extrakcii a analýze forenzných artefaktov nasleduje fáza obohacovania výstupných údajov, ktorá je založená na identifikácii časových okien, počas ktorých došlo k škodlivej činnosti. Časové okná pre jednotlivé súbory údajov sa určujú predovšetkým na základe verejne dostupných popisov scenárov CTF (CTF write-ups), ktoré špecifikujú priebeh a časový rámec incidentu, alebo na základe manuálnej forenznej analýzy konkrétnych prípadov CTF.

Všetky výstupy z nástrojov Eric Zimmerman Tools sa potom automaticky analyzujú. V rámci tejto analýzy sa v každom súbore CSV identifikujú atribúty obsahujúce časové značky (tzv. strieborné označenie). Na základe ich porovnania s definovaným časovým oknom incidentu sa každý riadok údajov vyhodnotí takto:

- hodnota **1,** ak aspoň jeden časový údaj v riadku spadá do identifikovaného časového okna bezpečnostného incidentu,
- hodnota **0,** ak žiadny časový údaj v riadku nepatrí do tohto časového okna.

Pri vyhodnocovaní jednotlivých dátových súborov sa berú do úvahy aj ďalšie dôležité časové míľniky známe zo scenárov CTF. Zvyčajne ide o čas inštalácie operačného systému, začiatok a koniec incidentu (časové rámce incidentu) a čas zabavenia systému alebo zberu digitálnych dôkazov. Informácie o inštalácii systému sa používajú na skrátenie údajov, pretože záznamy pred týmto časom by nemali byť v systéme prítomné a ich časové značky boli s vysokou pravdepodobnosťou upravené. Od času inštalácie systému až do začiatku incidentu sa záznamy hodnotia ako 0, počas incidentu ako 1 a po jeho skončení až do času zabezpečenia opäť ako 0. Záznamy po zabezpečení systému sa opäť odstránia z dátového súboru, pretože po tomto bode by sa do systému nemali pridávať žiadne nové legitímne artefakty a ich časové značky by sa nevyhnutne zmenili.

Príklady časových rámcov pre niektoré prípadové štúdie CTF sú uvedené v tabuľke 1.

Prípad	**Inštalácia OS**	**Časové rámce incidentov**	**Zbieranie digitálnych stôp**
Prípad úniku údajov NIST	22. 3. 2015 14:34:26	23. 3. 2015, 13:29 – 23. 3. 2015, 16:43, 24. 3. 2015 09:26 – 24. 3. 2015 17:06, 25. 3. 2015 10:46 – 25. 3. 2015 11:30	23. 4. 2015 10:58:22
SSS - DC	17. 9. 2020, 16:43:59	19. 9. 2020, 02:19 – 19. 9. 2020, 02:35	19. 9. 2020 04:51:27

SSS – Desktop	18. 9. 2020, 05:47:03	19.09.2020 02:35 – 19.09.2020 08:52	18. 9. 2020, 22:18:11
Magnet CTF 2019	28. 7. 2018 07:27:53	18. 3. 2019 18:35:00 – 18. 3. 2019 19:08:57	20. 3. 2019 21:29:33
Magnet CTF 2020	14. 2. 2020 02:10:21		22. 4. 2020 21:55:30
Magnet CTF 2022	4. 2. 2022, 07:05:47	10. 2. 2022 18:23 - 12. 2. 2022 02:20	13. 2. 2022 15:52:20

Tabuľka 1 – Časové rámce pre vybrané CTF

Výnimkou z tohto postupu sú výstupy, ktoré neobsahujú žiadne časové atribúty (napr. niektoré súbory CSV extrahované z registrov Windows). V týchto prípadoch nie je možné určiť vzťah záznamu k incidentu na základe času, a preto sa týmto záznamom priraďuje hodnota **N/A (neplatí)**.

Okrem automatizovaného hodnotenia času je vhodné doplniť súbor údajov ručným označením (tzv. gold labeling). V prípade scenárov CTF sa vykonáva podrobná ručná forenzná analýza, na základe ktorej sa záznamy, ktoré priamo súvisia s bezpečnostným incidentom s vysokou mierou istoty, označia hodnotou 1. Ostatné záznamy nie sú jasne vyhodnotené, pretože niektorým z nich nie je možné priradiť hodnotu 0 (jasne nesúvisiace s bezpečnostným incidentom), zatiaľ čo iné musia byť označené ako „neznáme“, pretože nebolo možné s istotou určiť ich vzťah k bezpečnostnému incidentu.

To znamená, že záznamy môžu byť hodnotené viacerými spôsobmi: buď automaticky na základe ich klasifikácie v časovom rámci incidentu (strieborné označenie), alebo manuálne na základe posúdenia, či konkrétny záznam z forenzného výstupu je relevantný alebo irelevantný pre daný bezpečnostný incident (zlaté označenie).

4.3 Vytvorenie vloženej dátovej sady

Na vstupe do potrubia na vytváranie dátových súborov založených na vkladaní používame výstup z **forenzného rámca Plaso** [25], konkrétne **superčasovú os**, ktorá predstavuje chronologicky usporiadané zoskupenie forenzných artefaktov extrahovaných z obrazu disku. Superčasová os integruje udalosti z viacerých zdrojov (napr. metadáta súborového systému, položky registru, protokoly udalostí, artefakty prehliadača) a poskytuje jednotný časový pohľad na činnosť systému.

Aby bola táto surová forenzná časová os vhodná pre analýzu založenú na strojovom učení, bol použitý viacfázový transformačný proces, ktorý pozostával z nasledujúcich krokov:

1) označenie údajov podľa časových okien incidentov uvedených v tabuľke (ADD),
2) skrátenie času inštalácie systému a zálohovania obrazu,
3) vytvorenie delty pre každé dva záznamy,
4) výber stĺpcov,

5) kombinovanie vybraných stĺpcov do textového poľa,
6) filtrovanie škálovača na všetkých deltách,
7) trénovanie tokenizéra na všetkých textoch,
8) rozdelenie na okná,
9) vytvorenie vložiek z textu v oknách a
10) vrátenie vložiek, delty, štítky ako trojice.

4.3.1 Označovanie údajov podľa časových okien incidentov (ADD)

Každý záznam supertimeline bol anotovaný na základe vopred definovaných **časových okien incidentov**, ktoré boli odvodené z predchádzajúcich znalostí o časoch vykonania útoku (skutočné údaje). Tieto časové okná predstavujú obdobia, počas ktorých došlo k známym škodlivým aktivitám. Záznamy spadajúce do týchto okien boli zodpovedajúcim spôsobom označené, čo umožnilo supervidované alebo polo-supervidované učenie. Tento krok vytvára priamu časovú väzbu medzi forenznými artefaktmi a fázami útoku.

4.3.2 Orezávanie artefaktov inštalácie systému a zálohovania obrazu

Aby sme znížili šum a zabránili skresleniu v súbore údajov, odstránili sme segmenty časovej osi, ktoré zodpovedali **inštalácii operačného systému**, počiatočnej konfigurácii a **forenznému získavaniu obrazov alebo zálohovaniu**. Tieto fázy zvyčajne generujú husté, ale irelevantné forenzné artefakty, ktoré môžu dominovať v časovej osi a skresľovať časové vzory nesúvisiace so správaním útočníka.

4.3.3 Výpočet delty s logaritmickým škálovaním

Pre každú dvojicu po sebe idúcich záznamov v superčasovej osi sme vypočítali **časovú delta**, definovanú ako rozdiel medzi ich časovými značkami. Táto delta zachytáva časovú dynamiku aktivity systému skôr ako absolútne časové hodnoty.
Aby sme zmiernili vplyv extrémnych rozdielov v časových značkách (napr. nečinnosť systému, obdobia vypnutia), delty sme transformovali pomocou **logaritmického škálovania**, ktoré stabilizuje variabilitu a zlepšuje robustnosť pre následné modely ML.

4.3.4 Výber vlastností (stĺpcov)

Z každého záznamu Plaso sme vybrali podmnožinu sémanticky významných atribútov, ktoré sa bežne používajú vo forenznej interpretácii:

- **užívateľ** – kontext užívateľa spojený s udalosťou
- **hostiteľ** – systém alebo názov hostiteľa
- **desc** – ľudsky čitateľný popis artefaktu
- **MACB** – sémantika aktivity súboru (modifikovaný, prístupný, zmenený, vytvorený)
- **sourcetype** – pôvod artefaktu (napr. NTFS, Registry, EVTX)

- **type** – typ artefaktu alebo udalosti

Tento výber vyvažuje bohatstvo kontextu s redukciou dimenzionality.

4.3.5 Textové zlúčenie vybraných atribútov

Vybrané stĺpce boli zlučené do jedného **textového znázornenia** na jeden záznam časovej osi. Táto transformácia prevádza heterogénne forenzné metadáta do jednotného textového formátu, čím sa stáva kompatibilným s **modelmi vkladania založenými na spracovaní prirodzeného jazyka (NLP)**.

V tejto fáze boli záznamy pochádzajúce zo všetkých obrazov diskov a experimentov zlúčené do **jedného konsolidovaného súboru údajov**, čím sa zabezpečila konzistentná tokenizácia a vkladací priestor vo všetkých vzorkách.

4.3.6 Prispôsobenie delta škálovača

Na všetky vypočítané hodnoty delta v celom súbore údajov bol prispôsobený globálny škálovač. Tento škálovač bol neskôr jednotne aplikovaný počas trénovania a hodnotenia modelu, čím sa zabezpečila konzistentná normalizácia časových charakteristík v rôznych vzorkách a scenároch.

4.3.7 Trénovanie tokenizéra na forenznom texte

Tokenizér bol trénovaný na kompletnom korpuse forenzných textových reprezentácií. Trénovanie tokenizéra na doménovo špecifickom forenznom jazyku (napr. cesty registrov, názvy spustiteľných súborov, popisy udalostí) umožňuje vkladacímu modelu lepšie zachytiť sémantiku digitálnych forenzných artefaktov v porovnaní s generickými tokenizérmi.

4.3.8 Rozdelenie superčasovej osi na okná

Kontinuálna supertimeline bola segmentovaná na **okná s pevnou dĺžkou**, z ktorých každé reprezentuje krátky časový kontext systémovej aktivity. Windowing umožňuje modelu naučiť sa lokálne vzorce správania a časové korelácie medzi udalosťami, namiesto toho, aby každý záznam spracovával izolovane.

Každé okno možno interpretovať ako mikrosekvenciu forenznej aktivity zodpovedajúcu potenciálnej fáze útoku alebo neškodnej prevádzke systému.

4.3.9 Generovanie vloženia z textu v okne

Pre každé okno boli vloženia generované z agregovaného textového obsahu pomocou vyškoleného tokenizátora a modelu vkladania. Tieto vloženia kódujú sémantické a kontextové informácie forenzných udalostí do hustých vektorových reprezentácií vhodných pre následné úlohy ML, ako je klasifikácia, zhlukovanie alebo detekcia anomálií.

4.3.10 Konštrukcia výstupu

Konečný dataset pozostáva z **trojíc**:

- **Vloženia** reprezentujúce sémantický kontext udalosti,
- **Časové delty** zachytávajúce dynamiku časovania,
- **Štítky** odvodené z anotácií okien incidentov.

Tento štruktúrovaný výstup umožňuje hybridné modelovacie prístupy, ktoré spoločne využívajú **sémantické**, **časové** a **dozorné** informácie, čím poskytujú robustný základ pre pokročilú forenznú analýzu a experimenty s detekciou útokov.

5 Bibliografia

[1] Grajeda, C., Breitinger, F., & Baggili, I. (2017). Availability of datasets for digital forensics—and what is missing. *Digital Investigation, 22*(Suppl.), S94–S105. https://doi.org/10.1016/j.diin.2017.06.004

[2] Luciano, L., Conti, M., Dragoni, N., Massacci, F., & Smith, C. (2018). Digital forensics in the next five years. In *Proceedings of the 13th International Conference on Availability, Reliability and Security (ARES 2018)*. ACM. https://doi.org/10.1145/3230833.3232813

[3] Breitinger, F., & Jotterand, A. (2023). Sharing datasets for digital forensics: A novel taxonomy and legal concerns. *Forensic Science International: Digital Investigation, 45*, 301562. https://doi.org/10.1016/j.fsidi.2023.301562

[4] Horsman, G., & Lyle, J. R. (2021). Dataset construction challenges for digital forensics. *Forensic Science International: Digital Investigation, 38*, 301264. https://doi.org/10.1016/j.fsidi.2021.301264

[5] Cymulate Ltd. (n.d.). *Platforma Cymulate*. Dostupné na https://cymulate.com/platform/ (prístupné 15. decembra 2025).

[6] DFIR Training. (n.d.). *Archív DFIR CTF: Testovacie obrázky*. Dostupné na https://www.dfir.training/downloads/test-images (prístupné 15. decembra 2025).

[7] Digital Corpora. (n.d.). *Diskové obrázky*. Dostupné na https://digitalcorpora.org/category/disk-images/ (prístupné 15. decembra 2025).

[8] Projekt Honeynet. (bez dátumu). *Forenzné výzvy*. Dostupné na https://www.honeynet.org/challenges/ (prístupné 15. decembra 2025).

[9] EVIDENCE Project Consortium. (n.d.). *Projekt EVIDENCE*. Dostupné na https://www.evidenceproject.eu/ (prístupné 15. decembra 2025).

[10] Kroll, LLC. (n.d.). *Kroll Artifact Parser and Extractor (KAPE)*. Dostupné na https://www.kroll.com/en/services/cyber/incident-response-recovery/kroll-artifact-parser-and-extractor-kape (prístupné 15. decembra 2025).

[11] Zimmerman, E. (n.d.). *Nástroje Erica Zimmermana*. Dostupné na https://ericzimmerman.github.io/#!index.md (prístupné 15. decembra 2025).

[12] Arsenal Recon. (n.d.). *Arsenal Image Mounter*. Dostupné na https://arsenalrecon.com/products/arsenal-image-mounter (prístupné 15. decembra 2025).

[13] Zimmerman, E. (n.d.). *AmcacheParser*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/AmcacheParser (prístupné 15. decembra 2025).

[14] Zimmerman, E. (n.d.). *AppCompatCacheParser*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/AppCompatCacheParser (prístupné 15. decembra 2025).

[15] Zimmerman, E. (n.d.). *EvtxECmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/evtx (prístupné 15. decembra 2025).

[16] Zimmerman, E. (n.d.). *JLECmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/JLECmd (prístupné 15. decembra 2025).
[17] Zimmerman, E. (n.d.). *LECmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/LECmd (prístupné 15. decembra 2025).
[18] Zimmerman, E. (n.d.). *MFTECmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/MFTECmd (prístupné 15. decembra 2025).
[19] Zimmerman, E. (n.d.). *PECmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/PECmd (prístupné 15. decembra 2025).
[20] Zimmerman, E. (n.d.). *RBCmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/RBCmd (prístupné 15. decembra 2025).
[21] Zimmerman, E. (n.d.). *RECmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/RECmd (prístupné 15. decembra 2025).
[22] Zimmerman, E. (n.d.). *SBECmd*. Dostupné na https://download.ericzimmermanstools.com/SBECmd.zip (prístupné 15. decembra 2025).
[23] Zimmerman, E. (n.d.). *WxTCmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/WxTCmd (prístupné 15. decembra 2025).
[24] Zimmerman, E. (n.d.). *SrumECmd*. Repozitár GitHub. Dostupné na https://github.com/EricZimmerman/Srum (prístupné 15. decembra 2025).
[25] Plaso Team. (n.d.). *Plaso documentation*. Dostupné na https://plaso.readthedocs.io/en/latest/ (prístupné 15. decembra 2025).