UNIVERZITA PAVLA JOZEFA ŠAFÁRIKA
PRÍRODOVEDECKÁ FAKULTA

IDENTIFIKÁCIA FÁZ ÚTOKU V POČÍTAČOVEJ SIETI

2024 **Bc. Marek DORKO**

UNIVERZITA PAVLA JOZEFA ŠAFÁRIKA
PRÍRODOVEDECKÁ FAKULTA

IDENTIFIKÁCIA FÁZ ÚTOKU V POČÍTAČOVEJ SIETI

DIPLOMOVÁ PRÁCA

Študijný program:	Informatika
Pracovisko (katedra/ústav):	Ústav informatiky
Vedúci diplomovej práce:	doc. RNDr. JUDr. Pavol Sokol, PhD.
Konzultant diplomovej práce: (nepovinný)	RNDr. Tomáš Bajtoš

Košice 2024

Bc. Marek DORKO

Zadanie záverečnej práce

Zadanie záverečnej práce (ďalej len „zadanie“) je dokument, ktorým vysoká škola stanoví študentovi študijné povinnosti v súvislosti s vypracovaním záverečnej práce. Zadanie spravidla obsahuje: typ záverečnej práce, názov záverečnej práce, meno, priezvisko a tituly študenta, meno, priezvisko a tituly školiteľa, v prípade externého školiteľa meno, priezvisko a tituly konzultanta, školiace pracovisko, meno, priezvisko a tituly vedúceho pracoviska, anotáciu záverečnej práce, jazyk, v ktorom sa práca vypracuje, dátum schválenia zadania.

Poďakovanie (nepovinné)

Na tomto mieste môže byť vyjadrenie poďakovania napr. vedúcemu práce resp. konzultantom za pripomienky a odbornú pomoc pri vypracovaní práce. Nie je zvykom ďakovať za rutinnú kontrolu, menšiu spoluprácu alebo všeobecné rady. Vyjadrenie poďakovania v prípade využitia inej práce sa uskutočňuje formou citácie na konci hlavného textu práce a odkazy na citáciu sa musia uviesť aj na zodpovedajúcich miestach v texte.

Abstrakt v štátnom jazyku

Činnosť útočníkov v rámci kybernetických útokov je možné rozdeliť do niekoľkých fáz, resp. krokov. V rámci tomto článku sa bližšie venujeme identifikácie jednotlivých fáz útoku. Podstatná časť článku sa venuje popisom prístupu k jednotlivým fázam útoku. Pre tento popis sa zameriavame na najčastejšie využívané rámce, ako je Cyber kill chain, Mandiant model a Mitre ATT&CK. Súčasne v rámci článku bližšie popisujeme skryté Markovove modely, ktoré predstavujú jeden z možných prístupov, ako riešiť identifikáciu fáz útoku v počítačovej sieti.

Kľúčové slová: kybernetická bezpečnosť, fázy útoku, Mitre ATT&CK, skryté Markovove modely

Abstrakt v cudzom jazyku

Text abstraktu v svetovom jazyku je potrebný pre integráciu do medzinárodných informačných systémov (napr. The Network Digital Library of Theses and Dissertations). Ak nie je možné jazykovú verziu umiestniť na jednej strane so slovenským abstraktom, je potrebné umiestniť ju na samostatnú stranu (cudzojazyčný abstrakt nemožno deliť a uvádzať na dvoch stranách).

Obsah

Zoznam ilustrácií (nepovinné)

Zoznam tabuliek (nepovinné)

Zoznam skratiek a značiek

μ **micro**, 10^{-6}

SI **S**ystème **I**nternational

V **volt**, základná jednotka napätia v sústave SI

Slovník termínov

Dizertácia je rozsiahla vedecká rozprava, v ktorej sa na základe vedeckého výskumu a s použitím (využitím) bohatého dokladového materiálu ako i vedeckých metód rieši zložitý odborný problém.

Font je súbor, obsahujúci predpisy na zobrazenie textu v danom písme, napr. na tlačiarni. To čo vidíme je písmo; font je súbor a nevidíme ho.

Meter (m) je vzdialenosť, ktorú svetlo vo vákuu prejde za časový interval 1/299 792 458 sekundy.

Proces je postupnosť či rad časovo usporiadaných udalostí tak, že každá predchádzajúca udalosť sa zúčastňuje na determinácii nasledujúcej udalosti.

Úvod

Kybernetické útoky ohrozujú používateľov a organizácie už od vzniku internetu. Spolu s počítačovými sieťami sa však stali oveľa viac zložitejšími. V dnešnej dobe potrebujú útočníci vykonať aj niekoľko krokov, aby dosiahli svoj konečný cieľ. Množinu takýchto krokov označujeme ako viacstupňový útok alebo scenár útoku. Ich viacstupňový charakter bráni detekcii narušenia systému, keďže na pochopenie stratégie útoku a identifikáciu hrozby je potrebná korelácia viac ako jednej akcie. Od začiatku roku 2000 sa komunita výskumníkov v oblasti bezpečnosti pokúsila navrhnúť riešenie na detekciu tohto druhu hrozby a predikovať ďalšie kroky útoku.

Pokročilí útočníci postupujú krok za krokom pri ich pokusoch o vykonanie útoku na systém. Je to hlavne z dôvodu, že obete zvolené útočníkmi sú zvyčajne stredné alebo veľké organizácie so zložitou topológiou siete a rôznymi vrstvami zabezpečenia. Vzhľadom na to, že najdôležitejšie dáta z hľadiska hodnoty informácie sú umiestnené v menej dostupných častiach siete, bolo by viac-menej nemožné uskutočniť úspešný prienik do systému pomocou jednokrokového útoku. Ďalším dôvodom je to, že ak sa útok rozloží na niekoľko krokov, tak je nenápadnejší a ťažšie identifikovateľný obeťou, najmä ak niektoré z krokov sami o sebe nepredstavujú riziko pre systém.

V rámci práce budeme uvažovať o reprezentácii postupu útočníka pomocou MITRE ATT&CK rámca [1]. Tento rámec obsahuje taktiky a v rámci nich jednotlivé techniky. Hlavnou myšlienkou práce je uľahčiť prácu forenznému a inému analytikovi, ktorý dostane k dispozícii údaje z jednotlivých zariadení v rámci počítačovej siete organizácie. Jeho úlohou je rekonštruovať činnosť útočníka. Inými slovami, identifikovať jednotlivé fázy. Táto úloha je reprezentovaná v rámci cieľa práca - navrhnúť a implementovať model identifikácie fáz útoku v počítačovej sieti vrátane vyhodnotenia efektívnosti tohto modelu.

Aby sme vedeli navrhnúť a implementovať vyššie uvedený cieľ, je potrebné analyzovať jednotlivé prístupy k identifikácii fáz útokov v počítačovej sieti. Vyššie sme uviedli jeden z týchto prístupov (MITRE ATT&CK rámec), od ktorého budeme vychádzať. V rámci práce chceme porovnať viacero týchto prístupov (napr. Kill chain model). V tejto práci budeme pracovať s dátami zozbieranými v rámci súťaže Guardians 2021 [2]. Tento dataset predstavuje reálne zozbierané dáta zo stredne veľkej organizácie (vydavateľstvo novín), na ktorú bol odsimulovaných niekoľko útokov vrátane

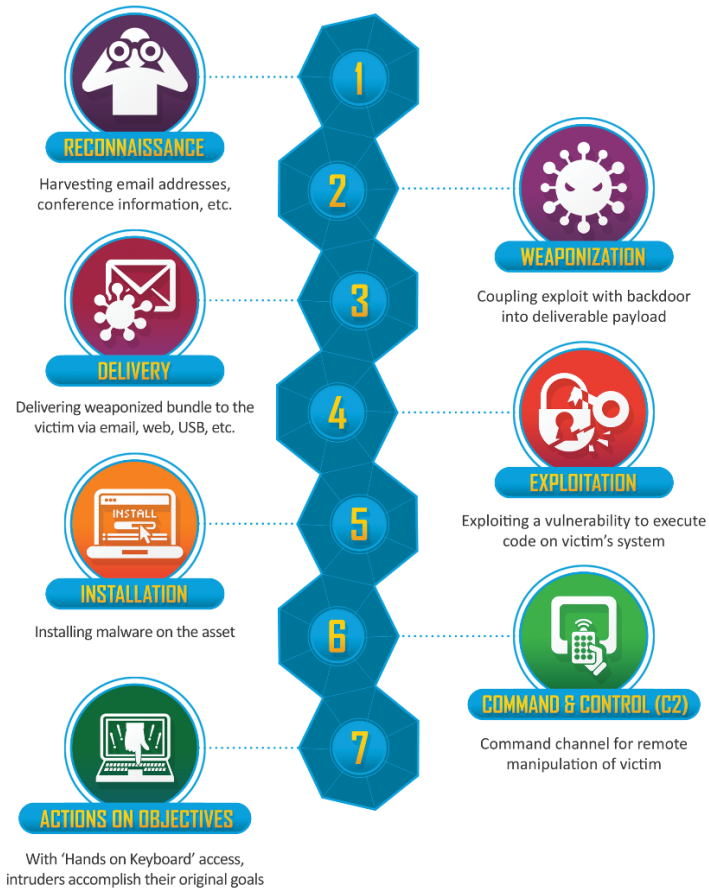
viacstupňových útokov. Dataset obsahuje dáta z koncových staníc, emailového servera, webového servera a ďalších zdrojov. Cieľom práce je prostredníctvom tohto datasetu identifikovať konkrétne digitálne stopy (najmä záznamy), ktoré reprezentujú jednotlivé fázy/kroky útočníka v čase. Zároveň by sme takto chceli zistiť, akú taktiku a techniky spadajúce pod túto taktiku, útočník využil.

1 Modely životného cyklu útoku

Jedným z cieľovej tejto práce je analyzovať prístupy k identifikácii fáz útoku v počítačovej siete. V tejto časti si jednotlivé prístupy bližšie popíšeme.

1.1 Cyber kill chain

Rámec Cyber kill chain, vyvinutý spoločnosťou Lockheed Martin, je súčasťou Intelligence Driven Defense modelu na identifikáciu a prevenciu kybernetických útokov. Model identifikuje, čo všetko musia útočníci splniť, aby dosiahli svoj cieľ. Pozostáva zo siedmich krokov: prieskum, zbrojenie, doručenie, zneužitie, inštalácia, velenie a riadenie a pôsobenie na ciele, ktoré zvyšujú viditeľnosť útoku a umožňujú analytikom lepšie pochopiť taktiky, techniky a postup útočníka. Jednotlivé fázy tohto modelu popisuje nasledujúci obrázok [3].



Obr. 1 Cyber kill chain model

Prvým krokom je prieskum, v ktorom sa primárne zameriava na zhromažďovanie informácií. To môže byť vykonávané na vyhľadávaním na konkrétnom cieli, zbieraním a kompilovaním údajov z online dostupných zdrojov, ako sú napríklad stránky sociálnych sietí.

V tejto fáze je tiež možné zbierať technické údaje spustením skenovania portov na cieľovej webovej lokalite a odhaliť tak potenciálne zastarané bežiace služby. Ďalšou možnosťou ako zozbierať údaje je zvyčajne pomocou phishingových emailov.

Po zozbieraní údajov nasleduje fáza zbrojenia, v ktorej môže útočník vytvoriť potenciálny malvér na základe analýzy zhromaždených dát. Napríklad, ak útočník vie, že v prostredí organizácie beží zraniteľná verzia Adobe Reader, môže vytvoriť malvér, ktorý túto zraniteľnosť bude zneužívať.

Fáza doručenia zvyčajne zahŕňa špeciálne vytvorené phishingové emaily, ktoré sa posielajú konkrétnym zamestnancom v organizácii. Takéto emaily často obsahujú priložené dokumenty programu Word s makrami alebo PDF súbory, ktoré obsahujú vložený odkaz, ktorý presmeruje obete na webové stránky útočníka, následne z ktorých je možné automaticky stiahnuť malvér.

Zneužitie je štvrtou fázou v útočnom cykle a kde sa vypláca fáza zbrojenia. Priložený súbor zvyčajne využije cielenú zraniteľnosť a môže sa pokúšať o stiahnutie ďalšieho softvéru na zariadenie obete pre vytvorenie perzistencie, čo už môžeme hovoriť o fáze inštalácie. Tá zahŕňa aj eskaláciu privilégií, interné skenovania na nájdenie špecifických nainštalovaných aplikácií, vytváranie naplánovaných úloh, registráciu služieb či úpravu registrov na zabezpečenie, že aplikácia prežije reštart zariadenia. Útočníci potom nastavia spojenie so serverom velenia a riadenia (Command and Control), kde budú pokračovať vo svojej činnosti.

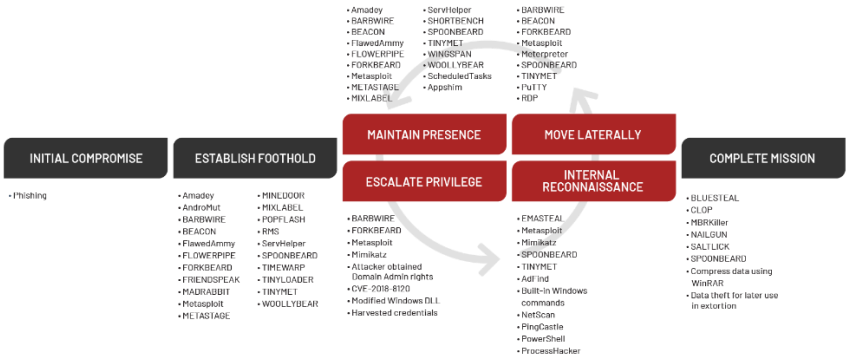
Po dokončení týchto šiestich fáz, útočníci prejdú k exfiltrácii údajov na ich servery. Nie je neobvyklé, že útočníci pokračujú vykonaním „bočného“ pohybu (Lateral Movement) s cieľom nájsť ďalšie systémy na kompromitáciu a zopakovať ich fázy znovu.

1.2 Mandiant model

Firma Mandiant pre kybernetickú bezpečnosť vytvorila model životného cyklu útoku, ktorý zahŕňa podrobné fázy o tom, ako fungujú APT (Advanced Persistent Threats) útočníci, resp. skupiny útočníkov a ako sa pohybujú v sieti s cieľom exfiltrácie údajov.

Na účely „lovu“ hrozieb poskytuje jedinečné príklady správania a nástrojov, na ktoré sa APT útočníci spoliehajú pri útoku na cieľ. Mandiant model je znázornený na Obr. 2 [4].

APT skupina [5] je zoskupenie kybernetických útočníkov, ktorí sa zameriavajú na pokročilé pretrvávajúce hrozby. Obvykle ide o ľudí z radov štátnych organizácií alebo organizácií, ktoré pracujú na objednávku štátu. Zameriavajú sa na cielené a sofistikované kybernetické operácie v snahe preniknúť do systémov vysoko postavených cieľov (vládne organizácie, korporácie) a nepozorovane v nich ostať dlhší čas. Účelom je zvyčajne dlhodobá kybernetická špionáž a odcudzenie citlivých údajov. APT skupiny disponujú širokou škálou poznatkov, pokročilými nástrojmi a technikami, vďaka ktorými dokážu zneužívať zero-day zraniteľnosti.



Obr. 2 Mandiant model

Mandiant model začína fázou počiatočnej kompromitácie (initial compromise), v ktorej sa zvyčajne vyskytuje phishingový email so škodlivou prílohou alebo odkazom. V porovnaní s predchádzajúcim modelom (Cyber kill chain), Mandiant model konsoliduje fázy zbrojenie, doručenie, zneužitie a inštalácia do fázy počiatočnej kompromitácie.

Zriadenie opory (establish foothold) je druhou fázou v tomto modeli, ktorá zahŕňa inštaláciu zadných dvierok (backdoor) po tom, čo doručený email bol spustený obeťou. Účelom zadných dvierok je vytvorenie spojenia s koncovým bodom.

Vo fáze eskalácie privilégií (escalate privileges) útočník získava ďalší prístup k podnikovým systémom a dátam. Útočníci často eskalujú svoje privilégiá prostredníctvom získavania poverení, zaznamenávania stlačenia kláves alebo rozvrátenia autentifikačných systémov.

Ďalším krokom je interný prieskum (internal reconnaissance), kde útočník skúma prostredie organizácie, aby lepšie porozumel infraštruktúre, ukladaniu informácií, ktoré ho zaujímajú a rolám a zodpovednostiam kľúčových jednotlivcov. V tomto čase útočníci minimalizujú akékoľvek abnormálne aktivity, a preto využívajú primárne vstavané príkazy operačného systému na preskúmanie kompromitovaných systémov.

Nasleduje fáza „bočný“ pohyb (move laterally), v ktorej útočník používa účty získané v predchádzajúcej fáze a presúva sa na ďalšie systémy v rámci kompromitovaného prostredia. Bežné techniky bočného pohybu zahŕňajú prístup k sieťovým zdieľaným súborom, vzdialené vykonávanie príkazov alebo prístup k systémom prostredníctvom protokolov vzdialeného prihlásenia, ako sú Remote Desktop Services (RDS) alebo Secure Shell (SSH).

Vo fáze udržiavania prítomnosti (maintain presence) útočník zabezpečuje nepretržitý prístup k prostrediu inštaláciou viacerých variantov zadných vrátok alebo prístupom k službám vzdialeného prístupu, ako je napríklad podniková virtuálna privátna sieť (VPN).

Poslednou fázou je dokončenie misie (complete mission), kde útočník dosiahne ciele, ako je krádež duševného vlastníctva, finančných údajov alebo informácií umožňujúcich identifikáciu osôb. V iných prípadoch môže byť cieľom misie narušenie systémov alebo služieb alebo zničenie údajov v prostredí.

Pochopenie krokov, ktoré útočníci podniknú, je dôležité na vytvorenie plánu na predchádzanie takýmto útokom a na zmiernenie rizík.

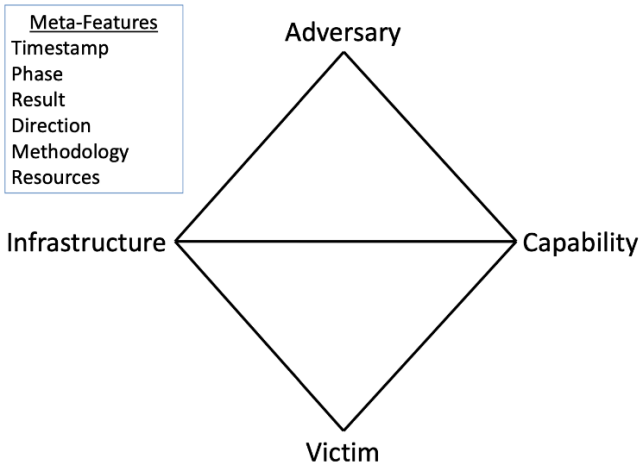
1.3 Diamantový model

Diamantový model [8] je jedným z nových modelov pre analyzovanie narušenia systému, kde útočník útočí na obeť prioritne v závislosti od dvoch kľúčových motivácií, na rozdiel od používania sérii krokov, ako je Kill chain model alebo graf útoku.

Tento model pozostáva zo štyroch základných prvkov, ako sú útočník, infraštruktúra, schopnosti a obeť. Útočník je herec (alebo skupina hercov), ktorí zaútočia na obeť po analýze ich schopnosti proti obeti. Spočiatku útočník začína so žiadnymi znalosťami o schopnostiach obete. Po analýze schopnosti obete, útočník môže zistiť, že on/ona má väčšiu schopnosť ako obeť, a teda či zaútočiť alebo nie. Tento model je dôležitý pri riešení pokročilejších útočníkov

ako sú tí, ktorí už získali určitú kontrolu nad sieťou. Útočník tiež analyzuje infraštruktúru technických a logických schopností veliť a ovládať ktorúkoľvek sieť obete.

Diamantový model môžeme vidieť na nasledujúcom obrázku.



Obr. 3 Diamantový model

1.4 MITRE ATT&CK model

MITRE ATT&CK (Adversarial Tactics, Techniques and Common Knowledge) [1] je celosvetovo dostupná základňa obsahujúca taktiky a techniky protivníkov. Je založená na pozorovaniach z reálneho sveta. Používa sa ako základ pre vývoj rôznych modelov hrozieb a metodológií v súkromnom sektore, vo vláde a v komunite produktov a služieb kybernetickej bezpečnosti.

ATTA&CK organizuje techniky do súboru taktík, ktoré pomáhajú vysvetliť a poskytnúť kontext pre techniku. Každá technika obsahuje informácie, ktoré sú relevantné pre červený tím (red team) alebo penetračného testera na pochopenie podstaty toho, ako technika funguje a tiež pre obrancu na pochopenie kontextu alebo artefaktov generovaných používanou technikou.

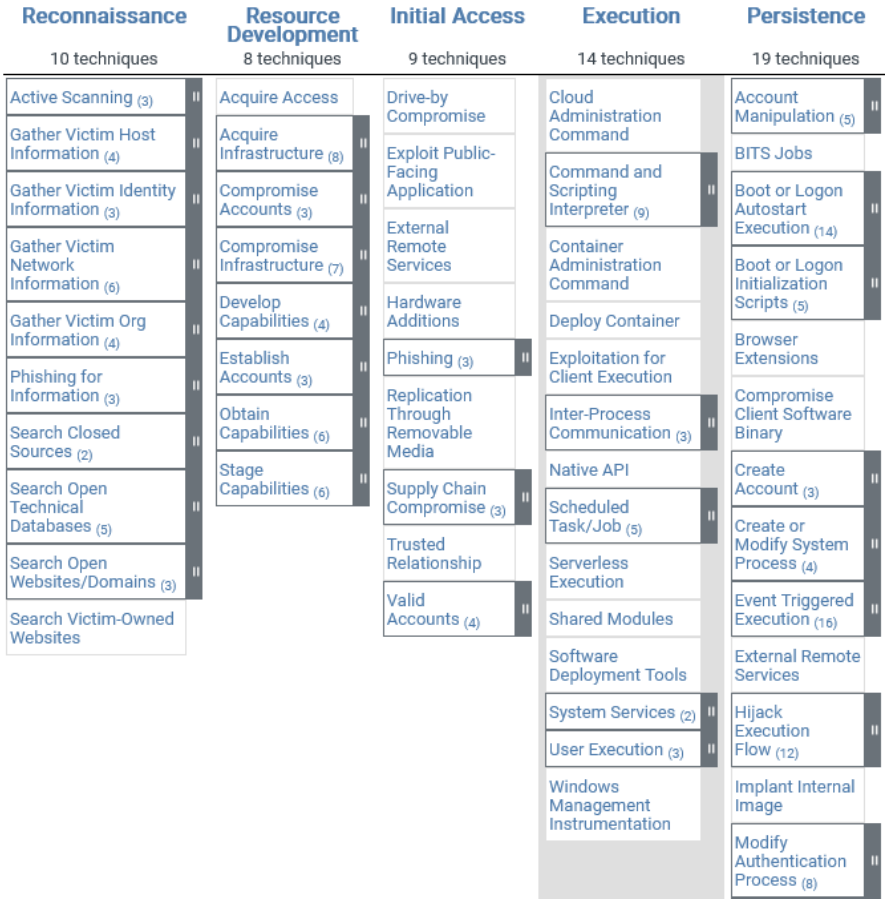
Taktika je taktickým cieľom útočníka. Taktiky slúžia ako užitočné kontextové kategórie pre jednotlivé techniky a pokrývajú štandardné zápisy vyššej úrovne pre veci, ktoré robia

útočníci počas operácie, ako je pretrvávanie, objavovanie informácií, „bočný“ pohyb, spúšťanie súborov a exfiltrácia údajov.

Techniky predstavujú „ako“ útočník dosiahne taktický cieľ vykonaním akcie. Napríklad, útočník môže získať prístupové údaje, aby získal prístup k užitočným prístupovým údajom v rámci siete, ktoré možno neskôr použiť na bočný pohyb. Techniky môžu tiež predstavovať „čo“ útočník získa vykonaním akcie. Spôsobov alebo techník na dosiahnutie taktických cieľov môže byť veľa, takže v každej kategórii taktiky existuje viacero techník.

1.4.1 ATT&CK rámec

Vzťah medzi taktikou a technikami je možné vizualizovať v ATT&CK rámci. Napríklad pod taktikou pretrvávanie (toto je cieľ útočníka – zotrvať v cieľovom prostredí) existuje séria techník vrátane Manipulácia s účtom (Account manipulation), Rozšírenia prehliadača (Browser extensions) či Naplánovaná úloha (Scheduled task/job). Každá z nich je samostatnou technikou, ktorú môžu útočníci použiť na dosiahnutie cieľa vytrvalosti/pretrvávania. Časť ATT&CK matice je možné vidieť na Obr. 4.



Obr. 4 Časť ATT&CK rámca

ATT&CK rámec je pravdepodobne najrozšírenejším aspektom ATT&CK, pretože sa bežne používa na zobrazenie vecí, ako je obranné pokrytie prostredia, detekčné schopnosti v bezpečnostných produktoch, výsledky incidentu a pod. Rámec pozostáva z nasledujúcich štrnástich taktík:

- Prieskum (Reconnaissance),
- Rozvoj zdrojov (Resource Development),
- Počiatočný prístup (Initial Access),
- Spustenie (Execution),
- Pretrvávanie (Persistence),
- Eskalácia privilégií (Privilege Escalation),
- Obranný únik (Defense Evasion),
- Prístup k prihlasovacím údajom (Credential Access),
- Objavenie (Discovery),
- „bočný pohyb“ (Lateral Movement),
- Zbierka (Collection),
- Príkaz a ovládanie (Command and Control),
- Exfiltrácia (Exfiltration),
- Dopad (Impact).

Pri každej technike v rámci ATT&CK rámca je tiež uvedené aké skupiny túto techniku využívajú. Skupiny [6] sú klastre aktivít, ktoré sú v bezpečnostnej komunite sledované pod spoločným názvom. Analytici sledujú tieto klastre pomocou rôznych analytických metodológií a pojmov, ako sú skupiny hrozieb, skupiny činností a aktéri hrozieb. Niektoré skupiny majú viacero mien spojených s podobnými aktivitami, pretože rôzne organizácie sledujú podobné aktivity pod rôznymi názvami. Definície skupín organizáciami sa môžu čiastočne prekrývať so skupinami určenými inými organizáciami a môžu sa nezhodovať v konkrétnej činnosti. Príklad takejto skupiny uvádzame na Obr. 5.

G0073	APT19	Codoso, C0d0so0, Codoso Team, Sunshop Group	APT19 is a Chinese-based threat group that has targeted a variety of industries, including defense, finance, energy, pharmaceutical, telecommunications, high tech, education, manufacturing, and legal services. In 2017, a phishing campaign was used to target seven law and investment firms. Some analysts track APT19 and Deep Panda as the same group, but it is unclear from open source information if the groups are the same.

Obr. 5 Skupina APT19

Jednotlivé stĺpce predstavujú ID skupiny, názov, pridružené skupiny a popis na čo sa skupina zameriava.

Ďalšou veľmi užitočnou informáciou, ktorá je pri jednotlivých technikách uvedená je softvér [7], ktorý útočníci využívajú. Softvér je rozdelený na:

- Nástroj – komerčný, open-source, verejne dostupný softvér, ktorý by mohol byť použitý obrancom, penetračným testerom alebo útočníkom. Táto kategória zahŕňa softvér, ktorý sa nenachádza v podnikovom systéme, ako aj softvér bežne dostupný ako súčasť operačného systému. Ako príklady takýchto nástrojov môžeme uviesť PsExec, Metasploit, Mimikatz, ako aj nástroje systému Windows, ako sú Net, netstat, Tasklist, atď.
- Malvér – komerčný softvér s vlastným uzavretým zdrojovým kódom alebo aj softvér s otvoreným zdrojovým kódom, ktorý je určený na použitie útočníkmi na škodlivé účely. Príklady takéhoto škodlivého softvéru zahŕňajú PlugX, CHOPSTICK, atď.

Pri technikách sú uvádzané aj zmiernenia. Tie predstavujú bezpečnostné koncepty a triedy technológií, ktoré môžu byť použité na zabránenie úspešnému vykonaniu techniky alebo pod-techniky.

2 Súčasné prístupy k identifikácii fáz útokov

V tejto kapitole si predstavíme niektoré existujúce prístupy k identifikácii viacfázových útokov, ktoré si bližšie popíšeme.

Autori v [13] navrhli riešenie na detekciu fáz útoku, ktoré využíva model sequence-to-sequence (seq2seq). Hlavnou myšlienkou je zakódovať sekvenciu alertov do vektora latentných znakov pomocou LSTM (Long short-term memory) siete a potom dekódovať tento vektor na sekvenciu predikovaných útočných fáz pomocou inej LSTM siete. Vďaka encoder-decoder spolupráci mohli autori oddeliť lokálne obmedzenie medzi pozorovanými alertami a potenciálnymi fázami útoku, a teda boli schopní plne poznať všetky alerty na zistenie fáz v sekvencii. Pomocou LSTM je možné sa naučiť „zabudnúť" na irelevantné alerty a mať tak viac možností „zapamätať si" dlhodobú závislosť medzi rôznymi štádiami pre detekciu sekvencie.

Štúdia [14] poukazuje na nový IDS (Intrusion Detection System), ktorý využíva kontextové informácie vo forme Pattern-of-Life (PoL) a informácie súvisiace s odborným posudkom na správanie sa siete. Tento IDS sa zameriava na detekciu viacfázových útokov v reálnom čase bez predchádzajúceho tréningového procesu. Výsledky autorov ukazujú, že použitie kontextovej informácie zlepšujú efektivitu IDS vylepšením miery detekcie viacfázových útokov v reálnom čase o 58%.

Strojové učenie a MITRE ATT&CK rámec využili autori v práci [15] pre skorú detekciu viacfázových útokov v reálnom čase. Najprv vyvinuli run-time engine, ktorý obdrží upozornenie, ak je škodlivý spustiteľný súbor stiahnutý cez prehliadač alebo spustením nového procesu v systéme. Po upozornení engine vyextrahuje atribúty zo spustiteľného súboru na učenie. Následne autori využili MITRE ATTA&CK rámec, vyvinutý na základe skutočných pozorovaní kybernetických útokov, ktorý najlepšie vystihuje viacfázový útok vzhľadom na taktiky, techniky a postupy útočníka (TPP – Tactics, Techniques and Procedure), aby detegovali škodlivý spustiteľný súbor a zároveň predikovali fázy, ktoré malvér spustí počas útoku. Model bol otestovaný na 6000 vzorkách malvéru a dosahuje úspešnosť 98%.

Autori v [16] prišli s metódou syntetizovania grafov scenárov zo stavových automatov. Smer siete využili na odvodenie potenciálnych fáz útoku z jednotlivých meta-alertov. Výsledný model obsahuje scenáre útoku v kill chain stavovom automate. Algoritmus poskytuje grafické zhrnutie útoku, kde vrcholy reprezentujú hostiteľov a hrany škodlivú aktivitu. Vyhodnotenie tohto prístupu vykonali vložením APT kampane

do dát zo sieťovej prevádzky, kde bola aj neškodná aktivita. Následne je vygenerovaná množina grafov APT scenárov, ktorá obsahuje aj vloženú kampaň redukovaním množiny alertov až o tri rády. Toto zníženie umožňuje analytikom efektívne triediť potenciálne incidenty.

V práci [17] autori navrhli nástroj na detekciu viacstupňových útokov, ktorý pozostáva z fázy generovania pravidiel na detekciu viacstupňových útokov a fázy samotného detegovania viacstupňových útokov. Po porovnaní prichádzajúcich dát zo sieťovej prevádzky s vygenerovanými detekčnými pravidlami boli nájdené rôzne vzory viacstupňových útokov bez vopred pozorovaných detailov správania sa jednej útočnej fázy. Na datasete DARPA LLS DDoS ukázali, že všetky možné vzory viacstupňových útokov boli správne detegované. Taktiež otestovanie na datasete CTU-13 dosiahlo maximálne F1 skóre 0,938.

Hidden Markov Model (HMM) je ďalším prístupom k detekcii viacstupňových útokov, ktorému sa venujú autori v [18]. Popisujú využitie HMM na detekciu komplexných útokov, ktoré pozostávajú z niekoľkých krokov, ktoré sa môžu vyskytnúť počas dlhšieho časového obdobia. Autori poukazujú na užitočnosť HMM v prípade, že je zachované poradie vykonaných akcií, teda pre prípad, kde jedna akcia musí predchádzať alebo nasledovať inej akcii, aby bola efektívna. V porovnaní s ďalšími dvoma technikami strojového učenia (rozhodovacie stromy a neurónové siete), HMM fungujú vo všeobecnosti lepšie ako rozhodovacie stromy a podstatne lepšie ako neurónové siete pri detekcii komplexných útokov.

2.1 Skrytý Markovov Model (Hidden Markov Model - HMM)

Skryté Markovove modely sa objavovali vo viacerých štúdiách a často moderné výskumy zaoberajúce sa detekciou komplexných útokov obhajujú použitie tohto prístupu. Preto sme sa v tejto podkapitole rozhodli bližšie popísať Skrytý Markovov Model.

Skryté Markovove modely [9] sa vo veľkej miere používajú na určovanie počítačových systémov, ktoré sú pod viacnásobným Multi-Stage Network Attack (MSA), avšak získanie optimálnych parametrov tréningu modelu zostáva výzvou.

Markovove modely fungujú lepšie pre nepozorovateľné stavy a prechody, čím sa eliminuje potreba úplných informácií pri predikcii a detekcii útoku. Markovove reťaze a Skrytý Markovov Model sú dva hlavné príklady Markovových modelov.

V kontexte sieťových útokov je útočná plocha modelovaná na základe Markovovej vlastnosti, ktorá definuje budúci stav ako funkciu len aktuálneho stavu. V praxi HMM vyžaduje dataset na trénovanie.

Skrytý Markovov Model je dvojúrovňový stochastický proces, kde prvá úroveň predstavuje stavy modelovaného systému, ktoré nie sú pozorovateľné. Druhá úroveň predstavuje pozorovania/emisie získané zo systému, čo v sieti môžu byť výstrahy IDS indikujúce stav systému. HMM je bežne reprezentovaný ako trojica (A, B, π), kde A je matica prechodu stavov, B je matica pravdepodobnosti pozorovania a π je počiatočný vektor pravdepodobnosti. Autori v [10, 11] dokonca definujú 6-ticu (S, V, A, B, π, O). Tri dodatočné parametre sú množina systémových stavov S, množina odlišných symbolov pozorovania V a sekvencia pozorovaní O.

Stručný popis šiestich HMM parametrov:

- Množina systémových stavov. Je to konečná množina N stavov reprezentovaná ako $S = \{s_1, s_2, \ldots, s_N\}$. Stav v čase t je označovaný ako q_t,
- Množina odlišných pozorovaní. Množina V obsahuje M odlišných symbolov pozorovania a je daná ako $V = \{v_1, v_2, \ldots, v_M\}$,
- Sekvencia pozorovaní O. Táto množina má premenlivú dĺžku T a predstavuje sekvenciu pozorovaní, ktorej prvky patria do množiny symbolov odlišných pozorovaní V. Množina je označená ako $O = \{o_1, o_2, \ldots, o_T\}$,
- Matica prechodov stavov A. Je to matica rozmeru NxN, v ktorej súčet riadkov je rovný jednej, teda $\Sigma\, a_{ij} = 1, \forall\, i, j \in [1, N]$

$$A = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1N} \\ a_{21} & a_{22} & \ldots & a_{2N} \\ \ldots & \ldots & \ldots & \ldots \\ a_{N1} & a_N & \ldots & a_{NN} \end{bmatrix}$$

  Každý prvok A, a_{ij} označuje pravdepodobnosť prechodu zo stavu i do j, teda $\forall\, t \in [1, T]$, $a_{ij} = P(q_{t+1} = s_j \mid q_t = s_i)$,
- Matica pravdepodobností pozorovaní B. Táto matica má rozmer NxN. $\Sigma\, b_j(v_k) = 1, \forall\, j \in [1, N]$ a $k \in [1, M]$

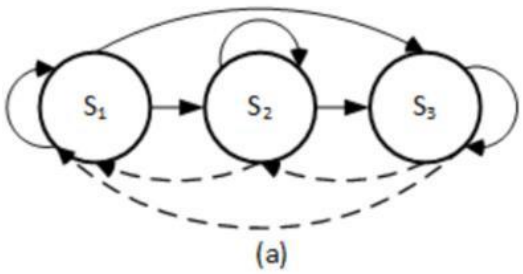
$$B = \begin{bmatrix} b_1(1) & b_1(2) & \dots & b_1(M) \\ b_2(1) & b_1(2) & \dots & b_2(M) \\ \dots & \dots & \dots & \dots \\ b_N(1) & b_N(2) & \dots & b_N(M) \end{bmatrix}$$

Každý prvok B $b_j(v_k)$ označuje pravdepodobnosť pozorovania symbolu v_k v stave j, teda $\forall t$, $b_j(v_k) = P(o_t = v_k \mid q_t = s_j)$,

- Vektor pravdepodobností počiatočného stavu π. Toto je reprezentované ako riadková matica pravdepodobností úvodných stavov v čase t = 1, napríklad $\pi_i = P(q_1 = s_i)$. Podobne ako A a B, $\Sigma\, \pi_i = 1$, $\forall\, i \in [1, N]$. $\pi = [\pi_1, \pi_2, \dots, \pi_N]$.

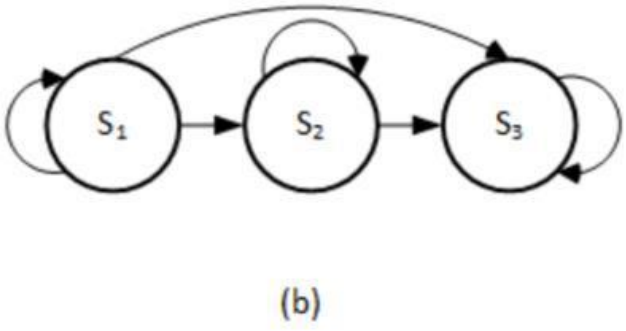
2.1.1 Typy Skrytých Markovových Modelov

Vo všeobecnosti existujú dva typy HMM: ergodic a left-right. Autori v [12] definujú ergodic HMM ako plne prepojený HMM, kde každý stav môže prejsť do akéhokoľvek možného stavu ako je znázornené na Obr. 6. Tento typ sa konkrétne nevzťahuje na viacfázové útoky, keďže sa len zriedka očakáva, že by útočník kompromitoval, resp. ohrozoval cieľ návratom do predchádzajúceho stavu alebo stavov, a nie pokračovaním smerom k cieľu.



Obr. 6 Ergodic HMM

Left-right typ HMM, znázornený na Obr. 7, nepovoľuje prechody do predchádzajúcich stavov, čo je typické vo viacfázových útokoch. Problém s týmto typom je, že $a_{ij} = 0$, $\forall\, j < i$ a nulové hodnoty sú obmedzením pre Baum-Welch algoritmus a Viterbiho implementáciách dekódovania, pretože sa očakáva, že prechod stavu bude $0 < a_{ij} \leq 1$. Tento problém možno vyriešiť priradením veľmi malých hodnôt namiesto núl a následne škálovanie každého riadku tak, aby súčet riadku bol rovný nule.



Obr. 7 Left-right HMM

2.1.2 Tri základné problémy HMM

Trénovanie (učenie), dekódovanie a hodnotenie sú tri základné problémy HMM. Prvý z nich je najnáročnejším problémom, pretože zohráva kľúčovú úlohu pri definovaní HMM. Nasleduje stručný popis týchto problémov a ako sa bežne riešia:

- Trénovanie (učenie): Toto nastaví všetky parametre HMM λ, na maximalizovanie pravdepodobnosti pozorovania P (O | λ). Baum-Welch algoritmus sa najčastejšie používa na riešenie tohto problému.
- Dekódovanie: Stavy HMM nie sú pozorovateľné; avšak pozorovania, ktoré sú generované v stave sú evidentné. Problém dekódovania sa snaží nájsť najviac pravdepodobnú cestu stavov (Viterbiho cesta), ktorá môže byť prechodná vzhľadom na sekvenciu pozorovaní O a HMM parametre λ. Na vyriešenie tohto problému sa zvyčajne používa Viterbiho dekódovanie.
- Hodnotenie: Vzhľadom na HMM je nevyhnutné určiť pravdepodobnosť generovanej pozorovanej sekvencie podľa toho modelu. Dopredné a spätné algoritmy sú bežne používané na túto úlohu.

3 Návrh riešenia

V tejto kapitole si popíšeme nami navrhnuté riešenie problému identifikácie fáz útoku. Najprv si povieme o generovaní dát, s ktorými sme následne pracovali.

3.1 Príprava virtuálneho prostredia

Prvým krokom v našej práci bolo vytvorenie čistého a izolovaného prostredia, teda virtuálneho stroja, kde sme si mohli vygenerovať dáta. Rozhodli sme sa pre nástroj VirtualBox od spoločnosti Oracle. Výhodou je možnosť vytvárania snímok systémov a tiež to, že ide o voľne dostupný virtualizačný nástroj. Na virtuálny stroj sme nainštalovali operačný systém Windows 10. Keďže v tejto práci sme využívali forenzné artefakty ako zdroj dát, potrebovali sme aj nástroj, ktorý zo systému vyextrahuje potrebné artefakty. Takýmto nástrojom je Kape nástroj. Je to efektívny a do veľkej miery konfigurovateľný forenzný nástroj, ktorý v systéme nájde cenné forenzné artefakty a následne zanalyzuje v priebehu niekoľkých minút [Kape web]. Do nášho virtuálneho prostredia sme teda nainštalovali Kape nástroj a taktiež Invoke-AtomicRedTeam modul na spúšťanie testov, ktorý si popíšeme v nasledujúcej podkapitole. Takýto stav systému sme považovali ako základný a čistý stav, na ktorý budeme simulovať útoky.

Návrh nášho riešenia v podstate spočíva v mapovaní forenzných artefaktov na MITRE ATT&CK techniky. V nultom kroku máme k dispozícii čistý virtuálny stroj, z ktorého pomocou nástroja Kape vyextrahujeme zvolené forenzné artefakty. Z týchto artefaktov získame využitím nástrojov od Erica Zimmermana CSV výstupy forenzných artefaktov pre čistý systém. Tento proces je znázornený na obrázku 8.

Obr. 8 Grafické znázornenie návrhu riešenia - 0. krok

V ďalšom kroku pre každú techniku z vybranej množiny techník vykonáme nasledovný proces. Na čistý systém spustíme Atomic test (popísaný nižšie), ktorý simuluje danú techniku. Z takéhoto stavu systému znova pomocou nástroja Kape vyextrahujeme zvolené artefakty. Následne pomocou nástrojov od Erica Zimmermana

získame modifikované CSV výstupy forenzných artefaktov pre danú techniku, čo je možné pozorovať aj z obrázka 9.

Obr. 9 Grafické znázornenie návrhu riešenia - 1. krok

V poslednom kroku ostáva porovnať získane artefakty čistého systému s artefaktami so systémov, na ktoré boli vykonané techniky. Obrázok 10 popisuje grafické znázornenie tohto kroku.

Obr. 10 Grafické znázornenie návrhu riešenia - 2. krok

To by znamenalo získanie informácií o tom, aké artefakty, ktorá technika po sebe zanecháva a možnosť vygenerovania podobnej tabuľky ako je vidieť v nasledujúcom obrázku.

Techniky/ artefakty	Register	Eventlog	Prefetch	LNK	...
T1.1	rozdiel	rozdiel	...		
T2.1	...				
...					

Obr. 11 Tabuľka rozdielov v artefaktoch medzi čistým systémom a systémom po vykonaní techniky

Nasledujúca fázou je vytvorenie pravidiel na hľadanie týchto rozdielov identifikujúcich konkrétnu techniku. Otestovanie by spočívalo v hľadaní rozdielov pomocou pravidiel na systéme, na ktorom boli vykonané viaceré techniky. Pravidlá nám asociujú techniku, ktorá bola pri útoku použitá.

3.2 Invoke-AtomicRedTeam

Atomic Red Team [19] je knižnica testov mapovaných na MITRE ATT&CK rámec. Je to open source projekt, ktorý je vyvinutý RedCanaryCo a komunitou. Bezpečnostné tímy môžu používať Atomic Red Team na rýchle, prenosné a reprodukovateľné testovanie svojich prostredí. Jednoduché testy sú cielené, majú málo závislostí a sú definované v štruktúrovanom formáte, ktorý vedia spracovať automatizačné nástroje.

Všetky testy sú uložené v jednom adresári nazvanom „atomics“ a sú rozdelené v adresároch s názvami podľa MITRE ATT&CK techník, ktoré reprezentujú. Každý adresár reprezentujúci techniku obsahuje:

- YAML test súbor,
- Markdown test súbor, ktorý je pre človeka čitateľnejší,
- Voliteľný adresár „src“, ktorý obsahuje potrebné závislosti,
- Voliteľný adresár „bin“, ktorý obsahuje potrebné binárne závislosti.

Vykonanie resp. spustenie testu je možné dvomi spôsobmi. Manuálny spôsob spočíva v otvorení Markdown súboru, kde je test definovaný a spustením príkazov, ktoré sú uvedené v časti „Attack Commands“. Druhý spôsob túto činnosť automatizuje. Invoke-AtomicRedTeam je Powershell modul na vykonávanie testov definovaných v „atomics“ adresári. My sme si tento automatizačný nástroj nainštalovali aj spolu s „atomics“ adresárom.

Medzi základné príkazy nástroja Invoke-AtomicRedTeam môžeme zaradiť

Invoke-AtomicTest T1003 -ShowDetailsBrief

Tento príkaz s prepínačom -ShowDetailsBrief zobrazí zoznam dostupných testov pre danú techniku (T1003). Rovnaký príkaz s prepínačom -ShowDetails zobrazí podrobnosti o teste, vrátane príkazov na útok, vstupných parametrov a potrebných pre-rekvizít pre danú techniku. Ďalším prepínačom je -CheckPrereqs, ktorý skontroluje, či náš systém spĺňa všetky pre-rekvizity vyžadované na spustenie testu. Toto overenie má zmysel pred samotným spustením testu. Ak zistíme, že systém nemá potrebné pre-rekvizity, môžeme použiť prepínač -GetPrereqs, ktorý sa ich pokúsi získať.

Na spustenie testu môžeme použiť príkaz

Invoke-AtomicTest T1218.010 -TestNumbers 1,2

Prepínač -TestNumbers v tomto prípade určuje, ktoré testy z danej techniky chceme vykonať.

3.3 Simulácia útokov (vykonávanie techník pomocou Atomic Red Team testov)

Techniky, ktoré sme vykonávali na naše testovacie virtuálne prostredie sme zvolili podľa toho, ako často jednotlivé techniky útočníci využívajú. Na začiatok sme vybrali päť techník, ktoré si bližšie popíšeme.

3.3.1 Technika T1003.002

Technika OS Credential Dumping s pod-technikou Security Account Manager [20] má označenie T1003.002. Využitím tejto techniky sa môžu útočníci pokúsiť extrahovať prihlasovacie údaje zo Security Account Manager (SAM) databázy buď pomocou techník v pamäti alebo Windows registrov, kde je SAM databáza uložená. SAM je databázový súbor, ktorý obsahuje lokálne účty pre hostiteľa. Typicky ide o tie, ktoré sú dostupné pomocou príkazu „net user“. Enumerácia SAM databázy vyžaduje prístup na úrovni SYSTEM.

Niektoré nástroje, ktoré môžu byť použité na získanie SAM súboru pomocou techník v pamäti:

- pwdumpx.exe
- gsecdump
- Mimikatz
- Secretsdump.py

Alternatívne, SAM súbor môže byť vyextrahovaný z Windows registrov:

- reg save HKLM\sam sam
- reg save HKLM\system system

Testy pre túto techniku sa nachádzajú v adresári /atomics/T1003.002. V tomto adresári je Markdown súbor, kde je popísaných sedem testov, ktorými môžeme vykonať techniku T1003.002. V našom prípade sme použili test #1, ktorý má názov Registry dump of SAM, creds, and secrets.

Príkazy na vykonanie útoku vyzerajú nasledovne:

reg save HKLM\sam %temp%\sam

reg save HKLM\system %temp%\system

reg save HKLM\security %temp%\security

Po úspešnom vykonaní testu nájdeme v adresári %temp% tri súbory s názvami sam, system a security. Tento test je možné vykonať na operačnom systéme Windows.

3.3.2 Technika T1053.005

Technika Scheduled Task/Job s pod-technikou Scheduled Task [21] má označenie T1053.005. Útočníci môžu zneužiť Plánovač úloh v systéme Windows na vykonanie plánovania úloh pre počiatočné alebo opakované spustenie škodlivého kódu. Existuje niekoľko spôsobov ako získať prístup k Plánovaču úloh v systéme Windows. Nástroj schtasks je možné spustiť priamo v príkazovom riadku alebo Plánovač úloh je možné taktiež otvoriť prostredníctvom grafického používateľského rozhrania. V niektorých prípadoch môžu útočníci použiť .NET obaľovač (wrapper) pre Windows Plánovač úloh. Alternatívne môžu využiť Windows netapi32 knižnicu na vytvorenie plánovanej úlohy.

Útočník môže využiť Windows Plánovač úloh na spúšťanie programov pri štarte systému alebo na pravidelnej naplánovanej báze pre trvalosť. Plánovač úloh možno tiež zneužiť na vykonávanie vzdialeného spustenia v rámci laterálneho pohybu a/alebo na spustenie procesu v kontexte špecifikovaného účtu (napríklad SYSTEM). Útočníci využívajú Plánovač úloh aj na potenciálne maskovanie jednorazového spustenia pri podpísaných/dôveryhodných systémových procesoch.

Útočníci môžu vytvoriť aj „skryté“ naplánované úlohy, ktoré nemusia byť viditeľné pre antivírusové nástroje. Konkrétne, útočník môže skryť naplánovanú úlohu z schtasks /query a Plánovača úloh vymazaním asociovanej hodnoty Security Descriptor (SD) v registroch. Vymazanie tejto hodnoty v registri vyžaduje SYSTEM oprávnenia.

Testy pre túto techniku sa nachádzajú v adresári /atomics/T1053.005. K dispozícii je desať testov, ktorými môžeme techniku vykonať. V práci sme použili test #1, ktorý má názov Scheduled Task Startup Script. Tento test spustí spustiteľný súbor pri prihlásení používateľa alebo spustení systému. Po vykonaní sa zobrazia správy o úspešnom naplánovaní dvoch úloh.

Príkazy na vykonanie útoku vyzerajú nasledovne:

schtasks /create /tn "T1053_005_OnLogon" /sc onlogon /tr "cmd.exe /c calc.exe"

schtasks /create /tn "T1053_005_OnStartup" /sc onstart /ru system /tr "cmd.exe /c calc.exe"

Tento test je možné vykonať na operačnom systéme Windows.

3.3.3 Technika T1082

Technika System Information Discovery [22] má označenie T1082. Útočník sa môže pokúsiť získať podrobné informácie o operačnom systéme, hardvéri, vrátane verzií, záplat, rýchlych opráv servisných balíkoch a architektúry. Útočníci môžu využiť tieto informácie na formovanie následného správania, vrátane toho, či útočník úplne infikuje cieľ a/alebo sa pokúsi o konkrétne akcie.

Na získanie detailných informácií o systéme možno použiť nástroje ako Systeminfo alebo útočníci s prístupmi bežného používateľa môžu spustiť príkaz df -aH na získanie informácií o aktuálne pripojených diskoch a súvisiaceho voľného priestoru. Tiež je možné využiť Command Line Interface (CLI) na sieťových zariadeniach na získanie systémových informácií (napr. show version). Zisťovanie systémových informácií v kombinácii s informáciami získanými z iných foriem objavovania a prieskumu môže viesť k rozvoju a utajovaniu útoku.

Poskytovatelia cloudových služieb ako AWS, GCP a Azure umožňujú prístup k informáciám o inštanciách a virtuálnych strojoch prostredníctvom rozhrania API. Úspešne autentifikované API volania môžu vrátiť údaje ako je platforma operačného systému a stav konkrétnej inštancie alebo zobrazenie modelu virtuálneho počítača.

Testy pre techniku T1082 sa nachádzajú v adresári /atomics/T1082. Je možné využiť 33 testov. Použili sme test #1 s názvom System Information Discovery.

Príkazy na vykonanie útoku vyzerajú nasledovne:

systeminfo

reg query HKLM\SYSTEM\CurrentControlSet\Services\Disk\Enum

Po úspešnom vykonaní sa zobrazia systémové a časové informácie. Test je možné vykonať na operačnom systéme Windows.

3.3.4 Technika T1543.003

Technika Create or Modify System Process s pod-technikou Windows Service [23] má označenie T1543.003. Útočníci môžu vytvárať alebo upravovať Windows služby tak, aby opakovane spúšťali škodlivé programy ako súčasť pretrvávania. Pri spustení systému Windows sa spúšťajú programy alebo aplikácie nazývané služby, ktoré vykonávajú systémové funkcie na pozadí. Informácie o konfigurácii Windows služieb, vrátane cesty k spustiteľnému súboru služby, sú uložené vo Windows registroch.

Útočníci môžu nainštalovať novú službu alebo modifikovať existujúcu službu tak, aby sa spustila pri spustení systému. Konfigurácie služieb je možné nastaviť alebo upraviť pomocou systémových nástrojov (napr. sc.exe), priamou úpravou hodnoty vo Windows registri alebo interakciou s API rozhraním systému Windows.

Útočníci môžu tiež využiť služby na inštaláciu a spustenie škodlivých ovládačov. Napríklad, po presunutí súboru ovládača (napr. .sys) na disk, môže byť tento súbor načítaný a registrovaný natívnymi API funkciami ako je CreateServiceW(). Útočník môže tieto ovládače zneužiť ako Rootkit, aby zakryl prítomnosť škodlivej aktivity v systéme.

Služby môžu byť vytvorené s oprávneniami správcu, ale sú vykonávané pod systémovými oprávneniami, takže útočník môže službu použiť na eskaláciu privilégií.

Testy pre techniku T1543.003 sa nachádzajú v adresári /atomics/T1543.003. K dispozícii je šesť testov, z ktorých sme v práci použili test #1 s názvom Modify Fax service to run Powershell.

Tento test dočasne upraví službu Fax zmenou cesty k jej spustiteľnému súboru na PowerShell a potom vráti zmenu do pôvodného stavu. Po úspešnom vykonaní sa spustí PowerShell.

Príkazy na vykonanie útoku vyzerajú nasledovne:

```
sc config Fax binPath= "C:\windows\system32\WindowsPowerShell\v1.0\powershell.exe -noexit -c \"write-host 'T1543.003 Test'\""
sc start Fax
```

Test je možné spustiť na operačnom systéme Windows.

3.3.5 Technika 1569.002

Technika System Services s pod-technikou Service Execution [24] nesie označenie T1569.002. Útočníci môžu zneužiť správcu riadenia Windows služieb na vykonávanie škodlivých príkazov. Správca riadenia služieb v systéme Windows (services.exe) je rozhranie na správu a manipuláciu so službami. Je prístupný používateľom prostredníctvom komponentov grafického používateľského rozhrania ako aj systémových nástrojov, ako sú sc.exe a Net.

PsExec môže byť použitý na vykonávanie príkazov prostredníctvom dočasnej Windows služby vytvorenej pomocou API správcu riadenia služieb. Nástroje ako PsExec a sc.exe môžu akceptovať vzdialené servery ako argumenty a možno ich použiť na vykonávanie vzdialeného spúšťania.

Útočníci môžu využiť tieto mechanizmy na spustenie škodlivého obsahu. To je možné vykonať spustením novej alebo upravenej služby. Táto technika je používaná v spojení s technikou Windows Service, ktorú sme popisovali v predchádzajúcej podkapitole, počas pretrvávania služby alebo eskalácii privilégií.

Testy pre techniku T1569.002 sa nachádzajú v adresári /atomics/T1569.002. K dispozícii je šesť testov, z ktorých sme použili test #1, ktorý má názov Execute Command as a Service.

Tento test vytvorí službu špecifikujúcu ľubovoľný príkaz a vykoná ho. Pri vykonávaní príkazov ako je PoweShell, bude služba hlásiť, že sa nespustila správne, aj keď sa kód spustí správne.

Test vyžaduje vstupy od požívateľa, ktoré za nás dosadí automatizačný nástroj Invoke-AtomicRedTeam. Potrebné vstupy pre test môžeme vidieť v nasledujúcej tabuľke.

Názov	**Popis**	**Typ**	**Hodnota**
service_name	názov služby, ktorá sa vytvorí	string	ARTService
executable_command	príkaz, ktorý sa spustí ako služba	string	%COMSPEC% /c powershell.exe -nop -w hidden -command New-Item -ItemType File C:\art-marker.txt

Tab. 1 Vstupy pre test #1 techniky T1569.002

Príkazy na vykonanie útoku vyzerajú nasledovne:

sc.exe create #{service_name} binPath= "#{executable_command}"

sc.exe start #{service_name}

sc.exe delete #{service_name}

Po úspešnom vykonaní testu sa vytvorí nová služba pomocou sc.exe, ktorá spustí powershell.exe na vytvorenie nového súboru art-marker.txt.

Test je možné spustiť na operačnom systéme Windows.

Sumarizácia techník sa nachádza v Tab. 2.

	Nad-technika	Taktika	Platforma
T1003.002	T1003	Prístup k prihlasovacím údajom	Windows
T1053.005	T1053	Spustenie, Pretrvávanie, Eskalácia privilégií	Windows
T1082	-	Objavenie	IaaS, Linux, Windows, macOS
T1543.003	T1543	Pretrvávanie, Eskalácia privilégií	Windows
T1569.002	T1569	Spustenie	Windows

Tab. 2 Sumarizácia techník

Túto základnú množinu piatich techník sme použili v našej práci. Každú techniku sme vykonali pomocou prislúchajúceho testu na čistý systém. Našim cieľom bolo získať informácie o tom ako sa jednotlivé techniky prejavia v systéme Windows. Po každom spustení techniky sme zo systému vyextrahovali množinu forenzných artefaktov, ktoré sme mohli následne analyzovať. Po extrahovaní artefaktov sme naše virtuálne prostredie obnovili do pôvodného čistého stavu a tento proces zopakovali na ďalšej technike.

3.4 Výber forenzných artefaktov a ich extrahovanie zo systému

V tejto podkapitole si popíšeme forenzné artefakty, ktoré sme sa rozhodli zo systému zozbierať a následne v ďalšom kroku analyzovať. Vybrali sme forenzné artefakty ako sú EventLogs, Master File tabuľka, Prefetch, LNK a Jumplist súbory, RecycleBin a Registre.

Všetky spomínané forenzné artefakty sme zo systému získali pomocou nástroja Kape. Tento nástroj nám však vyextrahuje aj binárne súbory, ktoré nie sú pre človeka čitateľné. Z tohto dôvodu sme použili Eric Zimmerman nástroje, ktoré jednotlivé forenzné artefakty zanalyzujú a rozparsujú do textových .csv súborov. Nástroje, ktoré sme použili:

- EvtxECmd.exe – parser na EventLogs (.evtx) so štandardizovaným CSV, XML a JSON výstupom,
- JLECmd.exe – parser na Jumplist,
- LECmd.exe – parser na LNK súbory,
- MFTECmd.exe – parser na Master File tabuľku ($MFT),
- PECmd.exe – parser na Prefetch,
- RBCmd.exe – parser na RecycleBin,
- RECmd.exe – parser na Windows registre

3.4.1 Forenzný artefakt EventLogs

Windows event log [25] je hĺbkový záznam udalostí týkajúcich sa systému, zabezpečenia a aplikácií uložených v operačnom systéme Windows. Tieto udalostí možno použiť na sledovanie problémov so systémom a niektorými aplikáciami a na predpovedanie budúcich problémov. Pomáhajú správcom siete sledovať potenciálne hrozby a problémy, ktoré potenciálne znižujú výkon. Operačný systém Windows ukladá tieto záznamy v štandardnom formáte, ktorý umožňuje jasné pochopenie informácií. Každý záznam má nasledovné atribúty:

- Názov – predstavuje názov záznamu udalosti, do ktorého sa budú zapisovať udalostí z rôznych komponentov logovania v systéme. Udalosti sa bežne zaznamenávajú pre systém, zabezpečenie a aplikácie.
- Dátum/čas udalosti – zahŕňa dátum a čas kedy k udalosti došlo.

- Kategória úlohy – identifikuje typ zaznamenaného záznamu udalosti. Vývojári aplikácií môžu tiež definovať kategórie úloh, ktoré budú slúžiť ako dodatočné informácie o udalosti.
- ID udalosti – toto identifikačné číslo pomáha správcom siete jednoznačne identifikovať konkrétnu zaznamenanú udalosť.
- Zdroj – názov programu alebo softvéru, ktorý spôsobuje záznam.
- Úroveň – predstavuje závažnosť zaznamenanej udalosti. Zahŕňa úrovne ako information, error, verbose, warning a critical.
- Používateľ – Meno používateľa, ktorý sa prihlásil do počítača so systémom Windows, keď došlo k udalosti.
- Počítač – názov počítača zaznamenávajúceho udalosť.

Udalostí sú rozdelené do štyroch kategórií, ako je aplikácia, bezpečnosť, nastavenie a systém.

- Systém – obsahuje udalosti súvisiace so systémom a jeho komponentami. Zlyhanie pri zavádzaní ovládača na spustenie systému je príkladom takejto udalosti na úrovni systému.
- Aplikácia – udalosti súvisiace so softvérom alebo aplikáciou uloženou na Windows systéme sa zaznamenávajú kategórie Aplikácia. Napríklad, problém pri spustení programu Microsoft PowerPoint sa zaznamená do tejto kategórie.
- Bezpečnosť – obsahuje udalosti súvisiace s bezpečnosťou systému. Udalosť sa zaznamená prostredníctvom Windows procesu auditovania. Príkladom môže byť neúspešné a úspešné prihlásenia, vymazanie súborov a pod.
- Nastavenie – obsahuje udalosti, ktoré sa vyskytnú počas inštalácie operačného systému Windows.

Udalosti sú v systéme Windows uložené v adresári C:\Windows\system32\config\.

3.4.2 Forenzný artefakt MFT (Master File Table)

Vo forenznej analýze je MFT [26] kľúčovou súčasťou operačného systému Windows. Ide o databázu, ktorá obsahuje informácie o každom súbore a adresári na pevnom disku. MFT sleduje umiestnenie súboru na pevnom disku a spravuje ďalšie

atribúty. Obsahuje metadáta o každom súbore ako je jeho názov, veľkosť, dátum vytvorenia a prístupové práva. Tieto údaje sú rozhodujúce pre akékoľvek forenzné vyšetrovanie.

MFT je nevyhnutný na zabezpečenie integrity a spoľahlivosti súborového systému. Sleduje všetky zmeny vykonané v súboroch a adresároch na disku, takže ak sa niečo pokazí, operačný systém môže použiť informácie v MFT na obnovenie súborového systému do predchádzajúceho stavu. Tieto zmeny poskytujú forenznému vyšetrovateľovi kritické informácie.

Okrem vyššie spomínaných atribútov MFT obsahuje aj príponu súboru, dátum úpravy, dátum posledného prístupu k súboru, dátum úpravy MFT záznamu a ďalšie informácie.

3.4.3 Prefetch

Windows od verzie Windows XP vytvára prefetch súbor [27] pri každom prvom spustení aplikácie. Tento súbor obsahuje údaje, ktoré operačný systém potrebuje na zrýchlenie načítania aplikácie pri každom jej spustení. Toto tiež pomáha rýchlejšiemu načítaniu systému Windows.

Prefetch súbory sú zvyčajne uložené v adresári C:\Windows\Prefetch\. Adresár obsahuje súbory s príponou .pf a každý súbor zodpovedá konkrétnemu programu alebo aplikácii, ktorá bola spustená na počítači.

3.4.4 LNK súbory

Súbory s príponou .lnk sa často označujú ako „súbory odkazov, resp. linkovacie súbory“ alebo skratky na pracovnej ploche [28]. Tieto súbory sú často spojené so systémom Microsoft Windows a zvyčajne poukazujú na spustiteľné súbory .exe, ktoré sú umiestnené na inom mieste v počítači používateľa. Keď klikneme na odkaz .lnk, spustí sa program priradený k súboru .exe, na ktorý odkaz odkazuje.

Umiestnenie týchto súborov sa líši od verzií systému Windows. Pre Windows 7 až 10 sa tieto súbory nachádzajú v adresári:

C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Recent.

Windows XP tieto súbory ukladá do adresára:

\Documents and Settings\UserName\Recent a

\Documents and Settings\UserName\Application Data\Microsoft\Office\Recent.

Napokon Windows Vista má súbory uložené v adresári:

\Users\UserName\AppData\Roaming\Microsoft \Windows\Recent a

\Users\UserName\AppData\Roaming\Microsoft\Office\Recent.

3.4.5 Jump list

Jump list [29] je funkcia, ktorá prišla so systémom Windows 7 a novšími verziami. Umožňuje používateľom zobraziť súbory, ku ktorým bolo naposledy pristupované v nainštalovaných programoch. Zoznamy týchto súborov umožňujú rýchlejší prístup k naposledy pristupovaným súborom.

Umiestenie Jumplist artefaktu je v adresári:

C:\Users\{username}\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations\. Súbory majú názov vo formáte {id}.automaticDestinations-ms, kde id predstavuje ID aplikácie.

V adresári C:\Users\{username}\AppData\Roaming\Microsoft\Windows\Recent\CustomDestination\ sú súbory vytvorené aplikáciou, zvyčajne preto, že so súborom sa stalo niečo dôležité (napr. súbor mohol byť označený ako obľúbený).

3.4.6 Registre Windows

Forenzné artefakty vo Windows registroch [30] môžu poskytnúť cenné informácie forenzným vyšetrovateľom. Register je hierarchická databáza, ktorá ukladá nastavenia na nízkej úrovni pre operačný systém, aplikácie a používateľov. Obsahuje informácie o nainštalovanom softvéri, konfiguráciách systému, používateľských nastaveniach a hardvéri. Medzi kľúčové položky registrov patria:

- Software – softvérový register HKEY_LOCAL_MACHINE\SOFTWARE obsahuje informácie o nainštalovanom softvéri, vrátane inštalačnej cesty, verzie a vydavateľa. To je užitočné pri identifikácii, aký softvér bol nainštalovaný v systéme a kedy bol nainštalovaný.

- System – systémový register HKEY_LOCAL_MACHINE\SYSTEM obsahuje informácie o konfiguráciách hardvéru a systému, vrátane ovládačov zariadení a konfigurácií služieb. To môže byť užitočné pri identifikácii hardvéru a ako bol nakonfigurovaný.
- Security – register zabezpečenia HKEY_LOCAL_MACHINE\SECURITY obsahuje informácie súvisiace so zabezpečením, vrátane používateľských účtov, skupinových politík zoznamov riadenia prístupu. To je užitočné pri identifikácii používateľských účtov a ich povolení v systéme.
- User profiles – každý používateľ v systéme má v registri HKEY_USERS svoj vlastný pod-register, ktorý obsahuje informácie o jeho nastaveniach a preferenciách, vrátane pozadia pracovnej plochy, internetovej histórie a iných údajov špecifických pre požívateľa. To môže byť užitočné pri identifikácii aktivít konkrétnych požívateľov v systéme.
- SAM – SAM (Security Account Manager), správca zabezpečenia účtov je register HKEY_LOCAL_MACHINE\SAM. Obsahuje informácie o lokálnych používateľských účtoch, vrátane hashovaných hesiel. To je užitočné na prelomenie hesiel a identifikáciu používateľských účtov v systéme.

Umiestnenie Windows registrov závisí od používanej verzie systému Windows. Od verzie Windows Vista jednotlivé registre nájdeme na týchto miestach:

- HKEY_LOCAL_MACHINE\SOFTWARE:
  %SystemRoot%\System32\Config\Software
- HKEY_LOCAL_MACHINE\SYSTEM:
  %SystemRoot%\System32\Config\System
- HKEY_LOCAL_MACHINE\SECURITY:
  %SystemRoot%\System32\Config\Security
- HKEY_USERS:
  %SystemRoot%\Users
- HKEY_LOCAL_MACHINE\SAM:
  %SystemRoot%\System32\Config\SAM

%SystemRoot% je premenná prostredia, ktorej hodnota je C:\Windows.

Súbory registrov sú binárne súbory, majú príponu .hiv a typicky majú názov podľa registra, ktorý reprezentujú (napr. SOFTWARE.hiv).

Našim cieľom je na základe tejto množiny artefaktov, ktorú sme získali zo systému určiť, ktorá technika bola pri útoku použitá, teda asociovať vzniknuté artefakty s použitou technikou útočníka. Rozhodli sme sa najprv analyzovať Eventlogs a MFT ako dva hlavné artefakty pomocou neurónových sietí. Cieľom je naučiť neurónovú sieť rozpoznávať na základe Event logov, prípadne kombinácie s MFT, ktoré budú na vstupe neurónovej siete, o akú techniku ide.

Záver

V článku sme sa zamerali na analýzu a porovnanie prístupov na identifikáciu fáz útoku. Uviedli sme Cyber kill chain, Mandiant model a napokon ATT&CK rámec a v každom popísali jednotlivé kroky.

V druhej kapitole sme uviedli niekoľko prác, ktoré sa venujú problematike identifikácie viacfázových útokov. Popísali sme existujúce riešenia a v skratke sme sa venovali skrytým Markovovym modelom, ktoré sa často používajú na riešenie tohto problému.

V ďalšej kapitole sme sa zaoberali návrhom nášho riešenia. Popísali sme jednotlivé techniky, ktoré sme využili a tiež forenzné artefakty, ktoré sme zo systému extrahovali.

Zoznam použitej literatúry

1. ATT&CK rámec [online]. [cit. 2023-06-23]. Dostupné na: https://attack.mitre.org/
2. Guardians [online]. [cit. 2023-06-23]. Dostupné na: https://www.guardians.sk/
3. Cyber kill chain [online]. [cit. 2023-06-23]. Dostupné na: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html
4. Mandiant model [online]. [cit. 2023-06-23]. Dostupné na: https://www.mandiant.com/resources/insights/targeted-attack-lifecycle
5. APT skupina [online]. [cit. 2023-06-23]. Dostupné na: https://www.eset.com/cz/aptskupina/
6. ATT&CK skupiny [online]. [cit. 2023-06-23]. Dostupné na: https://attack.mitre.org/groups/
7. ATT&CK softvér [online]. [cit. 2023-06-23]. Dostupné na: https://attack.mitre.org/software/
8. AL-MOHANNADI, Hamad, et al. Cyber-attack modeling analysis techniques: An overview. In: 2016 IEEE 4th international conference on future internet of things and cloud workshops (FiCloudW). IEEE, 2016. p. 69-76.
9. CHADZA, Timothy; KYRIAKOPOULOS, Konstantinos G.; LAMBOTHARAN, Sangarapillai. Analysis of hidden Markov model learning algorithms for the detection and prediction of multi-stage network attacks. Future generation computer systems, 2020, 108: 636-649.
10. THANTHRIGE, Udaya Sampath K.; SAMARABANDU, Jagath; WANG, Xianbin. Intrusion alert prediction using a hidden Markov model. arXiv preprint arXiv:1610.07276, 2016.
11. BROGI, Guillaume; BERNARDINO, Elena Di. Hidden Markov models for advanced persistent threats. International Journal of Security and Networks, 2019, 14.4: 181-190.

12. SHAWLY, Tawfeeq, et al. Architectures for detecting interleaved multi-stage network attacks using hidden Markov models. IEEE Transactions on Dependable and Secure Computing, 2019, 18.5: 2316-2330.
13. ZHOU, Peng, et al. Detecting multi-stage attacks using sequence-to-sequence model. *Computers & Security*, 2021, 105: 102203.
14. APARICIO-NAVARRO, Franciso J., et al. Multi-stage attack detection using contextual information. In: *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*. IEEE, 2018. p. 1-9.
15. TAKEY, Yuvraj Sanjayrao, et al. Real time early multi stage attack detection. In: 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE, 2021. p. 283-290.
16. WILKENS, Florian, et al. Multi-stage attack detection via kill chain state machines. In: *Proceedings of the 3rd Workshop on Cyber-Security Arms Race*. 2021. p. 13-24.
17. SHIN, Jinmyeong, et al. Unsupervised multi-stage attack detection framework without details on single-stage attacks. *Future Generation Computer Systems*, 2019, 100: 811-825.
18. OURSTON, Dirk, et al. Applications of hidden markov models to detecting multi-stage network attacks. In: *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*. IEEE, 2003. p. 10 pp.
19. Atomic Red Team [online]. [cit. 2024-01-22]. Dostupné na: https://atomicredteam.io/
20. T1003.002 [online]. [cit. 2024-01-22]. Dostupné na: https://attack.mitre.org/techniques/T1003/002/
21. T1053.005 [online]. [cit. 2024-01-22]. Dostupné na: https://attack.mitre.org/techniques/T1053/005/
22. T1082 [online]. [cit. 2024-01-22]. Dostupné na: https://attack.mitre.org/techniques/T1082/
23. T1543.003 [online]. [cit. 2024-01-22]. Dostupné na: https://attack.mitre.org/techniques/T1543/003/
24. T1569.002 [online]. [cit. 2024-01-22]. Dostupné na: https://attack.mitre.org/techniques/T1569/002/
25. Event logs [online]. [cit. 2024-01-22]. Dostupné na: https://www.solarwinds.com/resources/it-glossary/windows-event-log
26. MFT [online]. [cit. 2024-01-22]. Dostupné na: https://www.asdfed.com/Master-File-Table-and-Computer-Forensics

27. Prefetch [online]. [cit. 2024-01-22]. Dostupné na: https://www.makeuseof.com/windows-prefetch-files-explanation/

28. LNK [online]. [cit. 2024-01-22]. Dostupné na: https://www.minitool.com/lib/lnk-file.html

29. Jump list [online]. [cit. 2024-01-22]. Dostupné na: https://threat.media/definition/what-is-a-jump-list/

30. Windows register [online]. [cit. 2024-01-22]. Dostupné na: https://medium.com/@jontemwass/windows-registry-forensics-c092c6ebf3c3

Prílohy

Príloha A: CD médium – diplomová práca v elektronickej podobe, prílohy v elektronickej podobe.

Príloha B: Používateľská príručka

Príloha C: Systémová príručka

Táto časť diplomovej práce je povinná a obsahuje zoznam všetkých príloh vrátané elektronických nosičov. Názvy príloh v zozname musia byť zhodné s názvami uvedenými na príslušných prílohách. Tlačené prílohy majú na prvej strane identifikačné údaje – informácie zhodné s titulnou stranou diplomovej práce doplnené o názov príslušnej prílohy (Systémová príručka, Používateľská príručka). Identifikačné údaje sú aj na priložených diskoch alebo disketách. Ak je médií viac, sú označené aj číselne v tvare I/N, kde I je poradové číslo a N je celkový počet daných médií.

Každá príloha začína na novej strane a je označená samostatným písmenom alebo číslom (Príloha A, Príloha B, ... alebo Príloha 1, Príloha 2, ...). Číslovanie strán príloh nadväzuje na číslovanie strán v hlavnom texte.