Masarykova univerzita
Fakulta informatiky

Porovnanie kyberbezpečnostných hier

Bakalárska práca

Rudolf Lukáč

Brno, jar 2021

Masarykova univerzita
Fakulta informatiky

Porovnanie kyberbezpečnostných hier

Bakalárska práca

Rudolf Lukáč

Brno, jar 2021

Na tomto mieste sa v tlačenej práci nachádza oficiálne podpísané zadanie práce a vyhlásenie autora školského diela.

Vyhlásenie

Vyhlasujem, že táto bakalárska práca je mojím pôvodným autorským dielom, ktoré som vypracoval samostatne. Všetky zdroje, pramene a literatúru, ktoré som pri vypracovaní používal alebo z nich čerpal, v práci riadne citujem s uvedením úplného odkazu na príslušný zdroj.

Rudolf Lukáč

Vedúci práce: RNDr. Jan Vykopal, Ph.D.

Poďakovanie

Chcel by som sa poďakovať vedúcemu tejto práce, RNDr. Janovi Vykopalovi, Ph.D., za odborné rady a možnosť pravidelne konzultovať prácu aj v týchto nezvyčajných časoch. Rád by som sa poďakoval aj rodičom a priateľom za podporu počas štúdia a písania tejto práce.

Zhrnutie

Táto práca sa zaoberá porovnávaním kyberbezpečnostných hier vyvíjaných v Kybernetickom polygóne. Keďže súčasné definície hier postrádajú precíznu špecifikáciu obsahu, na základe ktorej by sa dali porovnať, rozhodli sme sa navrhnúť rozšírenie definícií o takúto špecifikáciu. Pre tento účel sme zvolili referenčný rámec *MITRE ATT&CK for Enterprise*, ktorý pokrýva techniky kyberútokov. V rámci práce sme vyvinuli programy, pomocou ktorých je možné porovnať hry na základe doplnenej špecifikácie alebo aj prehľadne zobraziť obsah viacerých hier. Vďaka programom je možné identifikovať podobné i rozdielne hry a ponúknuť hráčom hry spôsobom, ktorý zabráni tomu, že sa v ich hrách budú často opakovať rovnaké časti, čo by mohlo hráčov demotivovať. Vývojári hier pomocou nich môžu odhaliť oblasti kyberbezpečnosti, ktoré nie sú pokryté súčasnými hrami, a zamerať sa na ne pri tvorbe ďalších hier. Nová časť definícií pozostávajúca z techník útokov hry predstavuje užitočné rozšírenie, ktoré môže byť v budúcnosti použité aj na iné účely ako porovnávanie hier.

Kľúčové slová

KYPO, kyberbezpečnosť, kyberbezpečnostné hry, referenčné rámce kyberbezpečnosti, MITRE ATT&CK, porovnanie množín, Python, analýza príkazov

Obsah

1 Úvod **1**
1.1 Obsah práce 2

2 Prehľad **3**
2.1 Kybernetický polygón 3
2.2 Kyberbezpečnostné hry 3
2.3 Referenčné rámce znalostí v kyberbezpečnosti 5
2.4 Referenčné rámce techník útokov v kyberbezpečnosti . 10

3 Návrh **15**
3.1 Výber vhodného referenčného rámca 15
3.2 Modifikácia definícií kyberbezpečnostných hier 20
3.3 Vizualizácia porovnaní techník viacerých hier 22
3.4 Extrakcia a spracovanie dát z definícií hier 29

4 Implementácia **31**
4.1 Programy na vizualizáciu porovnaní hier 31
4.2 Programy na extrakciu a spracovanie dát z definícií hier 37
4.3 Pomocné knižnice 40
4.4 Konfigurácia prostredia 42

5 Záver **43**

A Klasifikácia hier pomocou predstavených rámcov **45**

B Elektronická príloha práce **53**

Bibliografia **55**

Zoznam tabuliek

2.1 Oblasti poznania kategórie Attacks and Defences [7] 9
2.2 Učebné výstupy oblasti poznania Software security [8] 9

3.1 Návrh tabuľky Jaccardových koeficientov 29

A.1 Klasifikácia prvej hry pomocou schopností NICE 45
A.2 Klasifikácia druhej hry pomocou schopností NICE 45
A.3 Klasifikácia tretej hry pomocou schopností NICE 46
A.4 Klasifikácia prvej hry pomocou znalostí NICE 46
A.5 Klasifikácia druhej hry pomocou znalostí NICE 47
A.6 Klasifikácia tretej hry pomocou znalostí NICE 47
A.7 Klasifikácia prvej hry pomocou CyBOK 48
A.8 Klasifikácia druhej hry pomocou CyBOK 48
A.9 Klasifikácia tretej hry pomocou CyBOK 48
A.10 Klasifikácia prvej hry pomocou CCF 49
A.11 Klasifikácia druhej hry pomocou CCF 49
A.12 Klasifikácia tretej hry pomocou CCF 50
A.13 Klasifikácia prvej hry pomocou Cyber Kill Chain 50
A.14 Klasifikácia druhej hry pomocou Cyber Kill Chain 50
A.15 Klasifikácia tretej hry pomocou Cyber Kill Chain 51
A.16 Klasifikácia prvej hry pomocou MITRE ATT&CK 51
A.17 Klasifikácia druhej hry pomocou MITRE ATT&CK 51
A.18 Klasifikácia tretej hry pomocou MITRE ATT&CK 52

Zoznam obrázkov

2.1 Štruktúra tréningu [5] 4
2.2 Časť definície tréningu 6
2.3 Stavebné bloky rámca NICE a vzťahy medzi nimi [6] 8
2.4 Časť referenčného rámca CCF [8] 10
2.5 Popis fázy útoku Exploitation [9] 11
2.6 Časť matice MITRE ATT&CK Matrix for Enterprise [10] 13

3.1 Rozšírená definícia jednej hernej úrovne 21
3.2 Definícia s riešením v nami navrhnutom formáte 23
3.3 Vizualizácia porovnania hier (MITRE ATT&CK Matrix) 25
3.4 Vizualizácia porovnania hier (Vennov diagram) 26
3.5 Vizualizácia porovnania hier (zobrazenie UpSet) 27
3.6 Vizualizácia porovnania hier (zobrazenie Supervenn) 28

4.1 Časť Jinja2 šablóny použitej na generovanie výstupu 32
4.2 Výstup programu technique_visualizer_matrix.py 33
4.3 Výstup programu technique_visualizer_matrix.py (2) 34
4.4 Výstup programu technique_visualizer_supervenn.py 36
4.5 Výstup programu technique_visualizer_jaccard.py 37
4.6 Výstup programu technique_visualizer_jaccard.py (2) 38
4.7 Výstup programu technique_extractor.py 38
4.8 Výstup programu technique_comparator.py 39
4.9 Výstup programu command_extractor.py 40
4.10 Výstup programu command_comparator.py 41

1 Úvod

Nedostatok pracovníkov v kyberbezpečnosti patrí medzi významné problémy súčasnosti. Podľa štúdie z roku 2020 [1] je v tomto odvetví na celom svete viac ako 3 milióny neobsadených pracovných pozícií v dôsledku absencie kvalifikovaných uchádzačov. Navyše až 56 percent respondentov tejto štúdie sa obáva, že tento nedostatok ohrozuje ich spoločnosť. Medzi možné riešenia tohto problému patria zvýšenie záujmu o kyberbezpečnosť medzi študentmi a zvýšenie dostupnosti praktickej výuky kyberbezpečnosti. To sú aj niektoré z cieľov vývoja kyberbezpečnostných hier v Kybernetickom polygóne Masarykovej univerzity, ktoré sa snažia nevšednou a interaktívnou formou priblížiť rôzne koncepty kyberbezpečnosti študentom a motivovať ich, aby sa v tejto oblasti ďalej vzdelávali. V týchto hrách sa hráč postupne v rámci niekoľkých úrovní snaží previesť úspešný útok na systém identifikáciou a zneužitím zraniteľnosti, ktorú tento systém obsahuje.

Jeden z problémov hier vyvíjaných v Kybernetickom polygóne predstavuje absencia prostriedkov umožňujúcich porovnávanie hier. Hry totiž často môžu byť postavené na rovnakej zraniteľnosti či type útoku, čo môže viesť pri hraní hier k nežiaducemu opakovaniu obsahu a znížiť ich učebný prínos pre hráčov. Hlavným cieľom našej práce teda bolo vyvinúť nástroje, pomocou ktorých bude možné hry porovnať a toto porovnanie rôznymi spôsobmi vizualizovať. Súčasné definície hier ale neobsahujú systematickú a strojovo čitateľnú špecifikáciu ich obsahu (napr. aký útok hra precvičuje), na základe ktorej by sa dali porovnať. V rámci práce sme preto navrhli ich rozšírenie o časť popisujúcu ich obsah. Pre tento účel sme zvážili použitie rôznych referenčných rámcov pokrývajúcich znalosti alebo techniky útokov v kyberbezpečnosti. Keďže hry sú zamerané na precvičenie počítačových útokov, rozhodli sme sa uprednostniť jeden z rámcov zameraných na techniky útokov. Následne sme vyvinuli sadu programov na porovnávanie hier na základe pridaných informácií. Ďalším cieľom práce bolo vyvinúť nástroj, ktorý používateľovi zobrazí nástroje využité pri hraní danej hry. Na extrakciu tejto informácie z definícií hier sme sa rozhodli využiť príkazy uvedené v riešení daných hier.

Vyvinuté programy umožňujú vývojárom hier či vyučujúcim získať lepší prehľad o väzbách medzi hrami a využiť ho na výber správ-

nych hier pre hráčov na základe nimi už odohraných hier. Aj samotné rozšírenie definícií hier o techniky útokov ale predstavuje užitočnú informáciu, ktorá môže napr. pomôcť hráčom vybrať nimi preferované hry. Naša práca zároveň poskytuje vývojárom hier nástroje, pomocou ktorých môžu prehľadne zobraziť techniky útokov viacerých hier. Techniky útokov hier môžu v tomto zobrazení aj hromadne agregovať a odhaliť tak nepokryté oblasti kyberbezpečnosti, na ktoré by sa mali zamerať pri tvorbe ďalších hier.

1.1 Obsah práce

Táto bakalárska práca je rozčlenená do piatich kapitol, počínajúc týmto úvodom. V druhej kapitole práce približujeme kyberbezpečnostné hry vyvíjané v Kybernetickom polygóne. Ďalej predstavujeme rôzne referenčné rámce pre popis znalostí, respektíve techník útokov v kyberbezpečnosti. V tretej kapitole porovnávame jednotlivé referenčné rámce z hľadiska potrieb našej práce. Následne prezentujeme náš návrh modifikácie definícií hier spolu so sadou programov pre extrakciu pridaných informácií a nástrojov z príkazov riešení daných hier. Súčasťou tejto kapitoly je ešte návrh programov na porovnanie hier a vizualizáciu tohto porovnania. V štvrtej kapitole práce približujeme implementáciu, použitie a vzorový výstup jednotlivých programov. V poslednej kapitole práce prezentujeme výsledky našej práce a jej ďalšie možné rozšírenia.

2 Prehľad

V tejto kapitole najskôr priblížime Kybernetický polygón a kyberbezpečnostné hry, ktoré sa v ňom vyvíjajú. Na to, aby sme mohli rozšíriť hry o špecifikáciu ich obsahu, sme potrebovali referenčný rámec, ktorým by sme obsah hier mohli popísať. Zvážili sme použitie rôznych referenčných rámcov zameraných na znalosti či techniky útokov v kyberbezpečnosti, ktoré taktiež predstavíme v tejto kapitole.

2.1 Kybernetický polygón

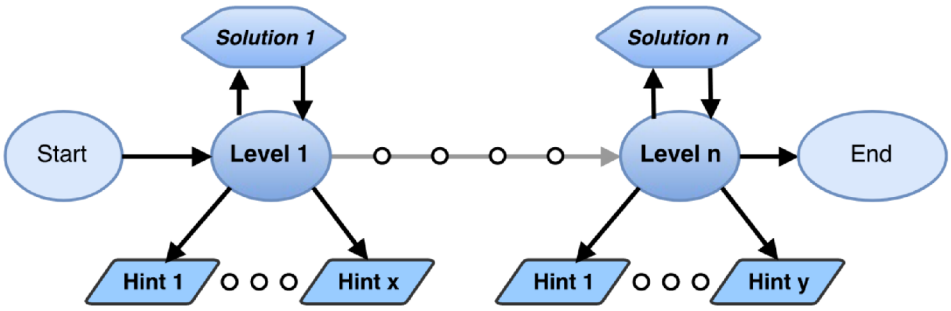
V rámci našej práce sa zaoberáme kyberbezpečnostnými hrami vyvíjanými študentmi v Kybernetickom polygóne. *Kybernetický polygón* (skrátene KYPO) predstavuje virtuálnu platformu pre organizáciu kyberbezpečnostných cvičení, výskum a výuku počítačovej bezpečnosti, najväčšiu svojho druhu v Českej republike [2, 3]. Mimo kyberbezpečnostných cvičení sa využíva aj v praktických predmetoch pre študentov Fakulty informatiky Masarykovej univerzity. Jedným z nich je aj Seminár simulácie kyberútokov [4] (s kódom predmetu PV276). Tento predmet pozostáva z dvoch častí. Prvá časť je teoretická, študentom sú priblížené bežné typy útokov na počítače, základy forenznej analýzy, detekcia a mitigácia útokov. V rámci druhej časti študenti v tímoch vytvárajú vlastné kyberbezpečnostné hry, ktorými sa v tejto práci zaoberáme.

2.2 Kyberbezpečnostné hry

Kyberbezpečnostné hry vyvíjané v KYPO sa podľa oficiálnej dokumentácie [3] označujú ako tréningy. V rámci tréningov sa hráči snažia riešiť úlohy zamerané na kyberbezpečnosť v izolovanom testovacom prostredí obsahujúcom virtuálne siete s virtuálnymi strojmi, ku ktorým sa môže hráč pripojiť a pomocou nich komunikovať s ďalšími strojmi v sieti. Toto prostredie sa zvykne označovať ako sandbox, v tejto práci budeme ďalej používať označenie virtuálne prostredie tréningov.

Štruktúra tréningu

Tréning pozostáva z niekoľkých typov za sebou idúcich úrovní. Základnou časťou hry sú herné úrovne, v ktorých je hráčovi zadaná úloha, riešením ktorej získa vlajku, reťazec. Zadaním tejto vlajky môže hráč pokročiť do ďalšej úrovne. Za úspešné zvládnutie hernej úrovne získava hráč body. Súčasťou herných úrovní sú aj nápovedy, ktoré môže hráč získať výmenou za bodovú penalizáciu, či odporúčané riešenie úrovne, ktoré môže hráč odhaliť vzdaním sa všetkých bodov za úroveň [5]. Okrem herných úrovní ešte tréning môže obsahovať informačné úrovne obsahujúce dodatočné informácie o hre a hodnotiace úrovne, ktoré môžu byť použité na ohodnotenie hráčových znalostí testom alebo získanie spätnej väzby od hráča pomocou dotazníka. Obrázok 2.1 znázorňuje lineárnu štruktúru hry, pre zjednodušenie neberie ohľad na úrovne, ktoré nie sú herné.



Obr. 2.1: Štruktúra tréningu [5]

Definícia tréningu

Obsah tréningu je popísaný definíciou tréningu, ktorá obsahuje základné informácie o hre ako názov hry alebo poznámky k hre pre inštruktorov, ľudí, ktorí usmerňujú hráčov pri hraní tréningov, a predovšetkým úrovne tréningu. Definície tréningov sú dostupné v JSON formáte [3]. Na obrázku 2.2 môžeme vidieť obecné informácie o hre ako jej názov a jednu hernú úroveň tréningovej definície, spočívajúcu v skenovaní zraniteľností serveru. Na základe definície tréningu

KYPO vytvára grafické rozhranie tréningov, cez ktoré sú hráčom napr. zadávané úlohy alebo poskytované nápovedy.

Definícia virtuálneho prostredia

Za súčasť tréningov považujeme aj definíciu virtuálneho prostredia tréningov. Tá pozostáva z definície jeho topológie, ktorá popisuje stroje spolu so sieťou, ktorou sú prepojené, a súborov na dodatočnú konfiguráciu strojov (vytvorenie používateľských účtov, inštalácia balíkov) [3]. Na základe definície sa konfiguruje virtuálne prostredie tréningov, v ktorom hráči riešia zadané úlohy.

2.3 Referenčné rámce znalostí v kyberbezpečnosti

V tejto sekcii priblížime rôzne referenčné rámce pokrývajúce znalosti v kyberbezpečnosti.

Referenčný rámec NICE

Workforce Framework for Cybersecurity [6] (skrátene NICE[1] Framework) je podkladom pre popis informácií o práci v kyberbezpečnosti od inštitútu National Institute of Standards and Technology. Medzi jeho ciele patrí pomôcť organizáciám špecifikovať kompetencie pracovných pozícií v kyberbezpečnosti či priradiť zamestnancom pracovné roly pri tvorení tímov v tomto odvetví.

Základnými stavebnými blokmi referenčného rámca NICE sú úlohy, znalosti a schopnosti.

- *Úloha* (anglicky Task) popisuje aktivitu, ktorá má byť vykonaná a jej zámerom je dosiahnutie nejakého cieľu organizácie. Popis úlohy by mal byť jednoduchý a začínať vykonávanou aktivitou. Príkladom úlohy je napr. „Perform security reviews, identify gaps in security architecture, and develop a security risk management plan.“ s identifikátorom *T0177*.

- *Znalosť* (anglicky Knowledge) je súbor poznaných konceptov. V referenčnom rámci NICE predstavuje predpoklad. Ku každej

1. z National Initiative for Cybersecurity Education

```
{
  "title" : "[Delta] COVID-19 Truth seeker",
  "description" : "Your company has been recently involved in few COVID-
19 conspiracies. Your colleagues don't really seem to care, but it has
caught your attention and you are struggling to work without being sure
that they are just made-up nonsense.  Can you find out the truth?",
  "prerequisities" : [ "Basic shell commands and constructs", "Basics of
computer networking", "Using SSH with public key authentication" ],
  "outcomes" : [ "Level 1: Scanning hosts for vulnerabilities using
Nmap, basics of CVE system", "Level 2: Using Metasploit to exploit
discovered vulnerabilities of host", "Level 3: Downloading files from
remote host using Netcat shell", "Level 4: Cracking encrypted SSH keys
using 'John the Ripper'" ],
  "state" : "UNRELEASED",
  "show_stepper_bar" : true,
  "levels" : [ {
    "type" : "GameLevelImportDTO",
    "title" : "Finding the path",
    "max_score" : 100,
    "level_type" : "GAME_LEVEL",
    "order" : 1,
    "estimated_duration" : 6,
    "flag" : "CVE-2019-15107",
    "content" : "As you have heard, there might be some outdated
software installed on the company's server. You should try to scan it
and see if you cand find some vulnerabilities. Fortunately, you work in
this company long enough to remember server's ip address - **10.1.26.9**
which will definitely help you. Flag is **CVE number** of the
vulnerability of that software discovered in **2019** with the **highest
CVSS Score** in this format: CVE-XXXX-XXXXX.\n\n,
    "solution" : "* Get list of server vulnerabilities: `nmap 10.1.26.9
--script=vuln` and analyze it.\n* Search \"webmin\" vulnerabilities on
www.cvedetails.com and choose the one with highest (10.0) CVSS Score.",
    "solution_penalized" : true,
    "hints" : [ {
      "title" : "Where to find the CVE number?",
      "content" : "Use www.cvedetails.com to find the right
vulnerability.",
      "hint_penalty" : 20,
      "order" : 2
    }, {
      "title" : "What tool can I use?",
      "content" : "You look to your left and you find word \"**nmap**\"
wrote on tile near the toilet paper. What can it be?",
      "hint_penalty" : 20,
      "order" : 0
    } ],
    "incorrect_flag_limit" : 100
  }, {
```


Obr. 2.2: Časť definície tréningu

úlohe sú priradené znalosti, ktoré sú nevyhnutné pre úspešné zvládnutie danej úlohy. Príkladom znalosti je napr. „Knowledge of cyber defense and vulnerability assessment tools and their capabilities.“ s identifikátorom *K0013*. Táto znalosť je priradená napr. k úlohe zmienenej ako príklad vyššie.

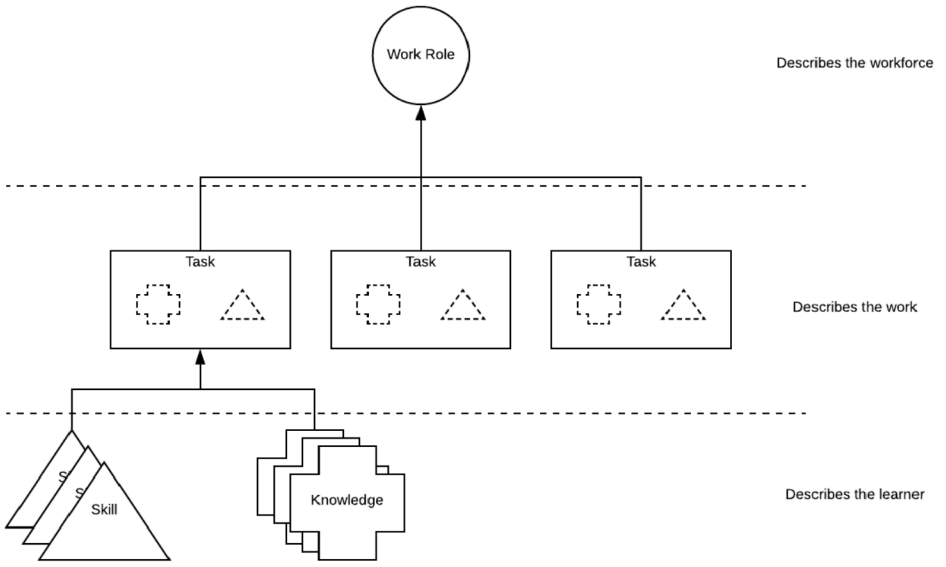
- *Schopnosť* (anglicky Skill) predstavuje spôsobilosť vykonať nejakú badateľnú činnosť. Rovnako ako znalosti sú aj schopnosti priradené k úlohám, pričom zvládanie danej schopnosti je nevyhnutné pre úspešné dokončenie danej úlohy. Príkladom schopnosti je napr. „Skill in conducting vulnerability scans and recognizing vulnerabilities in security systems.“ s identifikátorom *S0001*. Aj táto schopnosť je priradená k úlohe, ktorá bola uvedená ako príklad vyššie.

Tieto stavebné bloky môžu byť organizáciami využité na vytváranie pracovných rolí či kompetencií. Obe tieto konštrukcie sú podobné, pozostávajú predovšetkým z úloh, ktoré sú im priradené. Stavebné bloky rámca NICE sú teda usporiadané v hierarchiách, akú môžeme vidieť napr. na obrázku 2.3. Pracovné roly sa používajú agregovane na budovanie tímov zložených z rôznych takýchto rolí. Pomocou kompetencií sa napr. vytvára popis pracovnej pozície, do ktorej hľadá organizácia uchádzačov.

Referenčný rámec CyBOK

Referenčný rámec The Cyber Security Body of Knowledge [7] (skrátene CyBOK) sa snaží zmapovať znalosti v kyberbezpečnosti. Kyberbezpečnosť tento rámec rozdeľuje do 19 oblastí poznania, ako je napr. kryptografia. Navzájom súvisiace oblasti poznania sú spojené do piatich kategórií. V tabuľke 2.1 môžeme vidieť oblasti poznania kategórie *Attacks and Defences* spolu s ich krátkym popisom.

Každá oblasť poznania je rozdelená na sekcie a podsekcie a jej obsah nájdeme v samostatnom dokumente. CyBOK sa snaží poskytnúť úvod do všetkých relevantných znalostí každej oblasti. Ako uvádza dokumentácia tohto referenčného rámca, CyBOK sa nesnaží obsiahnuť všetko, čo bolo o daných oblastiach poznania publikované, ale poslúžiť ako sprievodca súborom znalostí.



Obr. 2.3: Stavebné bloky rámca NICE a vzťahy medzi nimi [6]

Referenčný rámec Cybersecurity Curricular Framework

Referenčný rámec Cybersecurity Curricular Framework [8] (skrátene CCF) je výsledkom činnosti spojenej pracovnej skupiny CSEC2017, ktorá vznikla v roku 2015 ako kolaborácia viacerých medzinárodných profesných organizácii vrátane Association for Computing Machinery či IEEE Computer Society. Jedným z hlavných cieľov CSEC2017 bolo vyvinúť podrobnú a flexibilnú osnovu pre výuku kyberbezpečnosti, ktorá by slúžila ako pomôcka pri vývoji učebných programov.

Podobne ako referenčný rámec CyBOK aj CCF združuje znalosti do oblastí poznania. Tie sú ďalej členené na jednotky poznania pozostávajúce z tém. K jednotkám poznania a témam je v CCF priradený krátky popis. Každá oblasť poznania obsahuje základné pojmy (anglicky essentials), ku ktorým tento rámec poskytuje aj učebné výstupy. Na obrázku 2.4 môžeme vidieť dve témy jednotky poznania *Fundamental principles* oblasti poznania *Software security* a v tabuľke 2.2 niektoré učebné výstupy tejto oblasti poznania.

Tabuľka 2.1: Oblasti poznania kategórie *Attacks and Defences* rámca CyBOK [7]

Attacks and Defences	
Malware & Attack Technologies	Technical details of exploits and distributed malicious systems, together with associated discovery and analysis approaches.
Adversarial Behaviours	The motivations, behaviours & methods used by attackers, including malware supply chains, attack vectors, and money transfers.
Security Operations	The configuration, operation and maintenance of secure systems including the detection of and response to security incidents and the collection and use of threat intelligence.
Forensics	The collection, analysis & reporting of digital evidence in support of incidents or criminal events.

Tabuľka 2.2: Niektoré učebné výstupy oblasti poznania *Software security* rámca CCF [8]

Essentials	**Learning outcomes**
Fundamental Design Principles; Least Privilege, Open Design and Abstraction	Discuss the implications of relying on open design or the secrecy of design for security. Describe why each principle is important to security.
Security requirements and the roles they play in design	Explain why security requirements are important. Identify common attack vectors.

SOFTWARE SECURITY		
Essentials - Fundamental design principles including least privilege, open design, and abstraction, - Security requirements and their role in design, - Implementation issues, - Static and dynamic testing, - Configuring and patching, and - Ethics, especially in development, testing and vulnerability disclosure.		
Knowledge Units	**Topics**	**Description/Curricular Guidance**
Fundamental Principles [*See also Component Security KA for related content, p. 29.*]		This knowledge unit introduces the principles that underlie both design and implementation. The first five are restrictiveness principles, the next three are simplicity principles, and the rest are methodology principles.
	Least privilege	Software should be given only those privileges that it needs to complete its task.
	Fail-safe defaults	The initial state should be to deny access unless access is explicitly required. Then, unless software is given explicit access to an object, it should be denied access to that object and the protection state of the system should remain unchanged.

Obr. 2.4: Časť referenčného rámca CCF [8]

2.4 Referenčné rámce techník útokov v kyberbezpečnosti

Mimo rámcov zameraných na znalosti sú verejne dostupné aj rôzne referenčné rámce orientované na techniky útokov v kyberbezpečnosti, ktoré predstavujeme v rámci tejto sekcie.

Referenčný rámec Cyber Kill Chain

Referenčný rámec Cyber Kill Chain [9] je súčasťou modelu Intelligence Driven Defense od spoločnosti Lockheed Martin pre identifikáciu a prevenciu vniknutí. Tento model identifikuje, čo musí útočník spraviť pre dosiahnutie cieľu útoku. Samotný rámec popisuje jednotlivé fázy

útoku. Jeho základnou myšlienkou je to, že útok pozostáva z viacerých fáz a zastavenie útočníka v hociktorej z nich potlačí celý útok. Popis každej fázy zahŕňa cieľ útočníka spolu s možnými metódami jeho dosiahnutia. Taktiež ale obsahuje možné spôsoby obrany v danej fáze útoku. Na obrázku 2.5 môžeme vidieť popis fázy útoku *Zneužitie zraniteľnosti*.

EXPLOITATION *Gain Access to Victim*

ADVERSARY

The adversaries must exploit a vulnerability to gain access. The phrase "zero day" refers to the exploit code used in just this step.

- Software, hardware, or human vulnerability
- Acquire or develop zero day exploit
- Adversary triggered exploits for server-based vulnerabilities
- Victim triggered exploits
  - Opening attachment of malicious email
  - Clicking malicious link

DEFENDER

Here traditional hardening measures add resiliency, but custom capabilities are necessary to stop zero-day exploits at this stage.

- User awareness training and email testing for employees.
- Secure coding training for web developers.
- Regular vulnerability scanning and penetration testing.
- Endpoint hardening measures:
  - Restrict admin privileges
  - Use Microsoft EMET
  - Custom endpoint rules to block shellcode execution
- Endpoint process auditing to forensically determine origin of exploit.

Obr. 2.5: Popis fázy útoku *Exploitation* rámca Cyber Kill Chain [9]

Referenčné rámce bázy znalostí MITRE ATT&CK

MITRE ATT&CK [10] je báza znalostí taktík a techník kyberbezpečnostných útokov založená na pozorovaniach skutočných útokov. Táto báza špecifikuje útoky v troch úrovniach:

- *Taktika* reprezentuje cieľ útoku, dôvod, prečo útočník danú akciu vykonáva. Taktiky pokrývajú zvyčajné časti útokov ako sú napr. iniciálny prístup, získanie prístupových údajov atď., podľa ktorých sú aj nazvané.
- *Technika* reprezentuje spôsob prevedenia útoku ako je napr. slovníkový útok. Môže obsahovať aj výsledok útoku, ako je to napr. u taktiky *Discovery*, ktorej techniky upresňujú, čo útok odhalil (súbor na počítači, počítač v sieti atď.).
- *Podtechnika* bližšie upresňuje techniku, napr. podtechniky techniky *Account Discovery* špecifikujú, aký druh účtu bol objavený (napr. emailový účet). Cieľom podtechník je zvýšiť úroveň abstrakcie techník a zredukovať počet techník na zvládnuteľné množstvo.

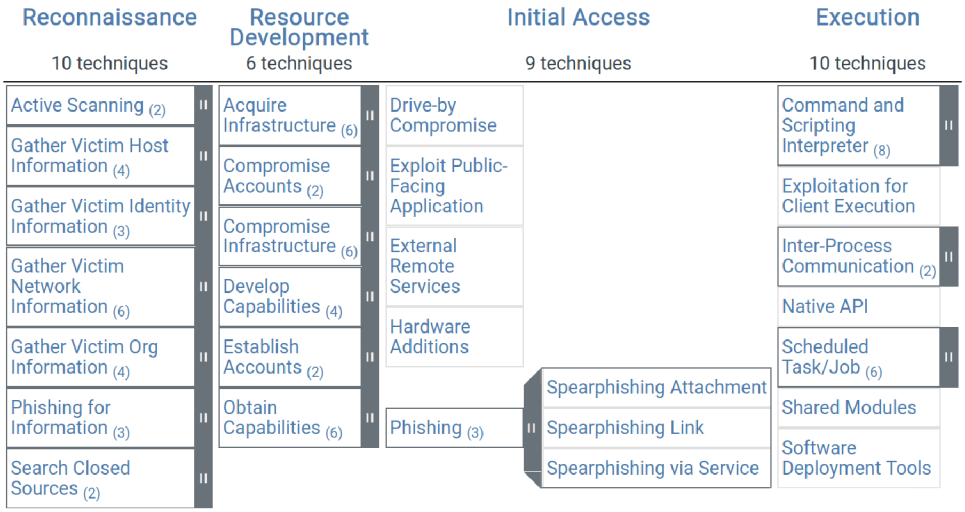
Referenčné rámce bázy znalostí MITRE ATT&CK určujú, na akú technologickú doménu sa jednotlivé taktiky so svojimi technikami vzťahujú. Báza obsahuje tieto tri rámce:

- *Enterprise*, ktorý pokrýva tradičné podnikové siete a cloud technológie. Svojim zameraním je tento rámec najbližšie systémom objavujúcim sa v hrách v Kybernetickom polygóne.
- *Mobile*, v rámci ktorého sú obsiahnuté mobilné telefóny.
- *ICS*, ktorý pokrýva priemyselné kontrolné systémy.

Zmienené rámce sa ďalej môžu deliť podľa presnej platformy (operačného systému) daných zariadení, napr. Mobile na Android a iOS.

Na vizualizáciu jednotlivých referenčných rámcov sa využíva zobrazenie *MITRE ATT&CK Matrix* [11], ktoré ďalej budeme označovať ako maticu (na základe prekladu). V rámci tohto zobrazenia sú taktiky usporiadane v stĺpcoch vedľa seba, pričom obsahom týchto stĺpcov sú techniky jednotlivých taktík. Podtechniky techník sú prístupné po

kliknutí na tlačidlo vedľa techniky. Obrázok 2.6 obsahuje časť matice referenčného rámca Enterprise, nazývanú MITRE ATT&CK Matrix for Enterprise. Zobrazená je len časť tejto matice (niektoré techniky 4 zo 14 taktík), keďže celá matica je príliš veľká pre dostatočne viditeľné zobrazenie v rámci tejto práce.



Obr. 2.6: Časť matice *MITRE ATT&CK Matrix for Enterprise* [10]

3 Návrh

V tejto kapitole porovnáme jednotlivé referenčné rámce z hľadiska požiadaviek našej práce, predstavíme naše návrhy modifikácií herných definícií a štruktúru programovej časti práce.

3.1 Výber vhodného referenčného rámca

Rôzne rámce znalostí či techník útokov v kyberbezpečnosti sme bližšie predstavili v predchádzajúcej kapitole tejto práce. V tejto sekcii zhodnotíme adekvátnosť jednotlivých rámcov vzhľadom na požiadavky našej práce a zvolíme ten, ktorý sa ukáže ako aj najvyhovujúcejší.

Pri posudzovaní vhodnosti ich použitia sme pracovali s tromi hrami: dve z nich boli založené na zneužití rovnakej zraniteľnosti a boli do veľkej miery podobné (mierne sa líšili v cieli útoku), konkrétne sa jednalo o hry s názvom *COVID-19 Truth Seeker* a *Uploaded Kobylka 3302*. Tretia hra mala názov *Captain Slovakistan* a narozdiel od prvých dvoch hier v nej bola využitá iná zraniteľnosť s iným spôsobom útoku. Medzi požiadavky kladené na hodnotené rámce teda patrilo aj to, že pri porovnaní hier pomocou nich by mali vyjsť prvé dve hry ako veľmi podobne a značne odlišné od tretej hry. Ďalšími boli vysoká úroveň detailu rámca a dostupnosť dostatočne kvalitných materiálov umožňujúcich efektívnu prácu s daným rámcom (napr. dokumentácie alebo nástrojov na vyhľadávanie v rámci). Tieto vlastnosti boli dôležité preto, aby vývojári hier dokázali klasifikovať svoje hry pomocou zvoleného rámca bez nutnosti vynaložiť enormné úsilie na pochopenie štruktúry a obsahu tohto rámca. Klasifikácie každej hry podľa jednotlivých rámcov sú dostupné v prílohe práce A.

NICE

Na prácu s rámcom NICE sú dostupné až tri portály [12]. V dvoch z nich sa ale dá v rámci orientovať len cez pracovné role (a im priradené úlohy, znalosti a schopnosti), čo bolo v kontexte našej práce veľmi neefektívne. Zamerali sme sa teda na zvyšný portál [13], ktorý umožňoval vyhľadávanie medzi schopnosťami a znalosťami na základe slov, ktoré obsahovali. Hoci tento prístup bol značným zlepšením oproti

predchádzajúcemu, aj tak sa nedal označiť za ideálny. Pokúsili sme sa teda opísať hry pomocou úloh, schopností aj znalostí rámca NICE. V prípade úloh sme ale dospeli k záveru, že to je prakticky nemožné, pretože pracovné role v NICE (a aj ich úlohy) sú prevažne defenzívneho charakteru, čo predstavuje značný kontrast s ofenzívne ladenými kyberbezpečnostnými hrami. So znalosťami aj schopnosťami sa nám to podarilo, nami priradené schopnosti k jednotlivým hrám môžeme vidieť v tabuľkách A.1, A.2 a A.3, znalosti v tabuľkách A.4, A.5 a A.6. Aj v tomto prípade sme ale narazili na niekoľko problémov. V prípade schopností sa nám nepodarilo nájsť vhodnú schopnosť pre aspoň jednu úroveň v každej hre, napr. sme nenašli v rámci vhodnú znalosť pre opis útoku SQL injection, ktorý sa využíva v druhej úrovni hry Captain Slovakistan na obídenie prihlásenia. Taktiež sme mali problém klasifikovať hlavný cieľ útokov v jednotlivých hrách (prístup k tajným informáciám), museli sme sa uspokojiť so schopnosťou „Skill to identify sources, characteristics, and uses of the organization's data assets.“, ktorá ale nie je presným opisom daných úrovní. Aj v prípade znalostí sme narazili na problematickú úroveň, konkrétne sa jedná o tretiu úroveň v hre Captain Slovakistan, kde je úlohou hráča nahrať na webový server web shell [1] a pomocou neho získať obsah súboru. Jedná sa o dôležitú časť útoku, ktorá je ale pomerne jednoduchá (stačí využiť používateľské rozhranie serveru). Pre jej opis sme nenašli vhodnú znalosť.

CyBOK

S referenčným rámcom CyBOK je možné pracovať dvoma rôznymi spôsobmi: buď je možné využiť samostatné dokumenty ku každej z 19 oblastí poznania alebo agregovaný dokument všetkých týchto oblastí. Ani jeden z týchto prístupov sa ale neukázal ako efektívny. V prvom prípade môže byť zložité určiť, kde hľadať vhodnú sekciu či podsekciu pre opis úrovne. U agregovaného dokumentu predstavuje problém jeho veľkosť, viac ako 850 strán, čo značne komplikuje orientáciu v tomto dokumentu. Napriek tomu sme sa ho rozhodli uprednostniť, pretože práca s jednotlivými dokumentmi vyžadovala pokročilú znalosť tohto rámca. Pri klasifikácii hier (tabuľky A.7, A.8

1. skript, ktorý po nahratí na server umožňuje útočníkovi získať nad ním kontrolu [14]

a A.9) sme narazili na niekoľko problémov. Najväčším z nich bol ten, že CyBOK neobsahuje dostatočne presnú špecifikáciu možných cieľov útoku. Táto téma sa čiastočne rozoberá v sekcii *A Characterisation of Adversaries* oblasti *Adversial Behaviour*, lenže táto sekcia postráda členenie na subsekcie, pomocou ktorých by sa dali hry presnejšie opísať. Taktiež sme mali problém klasifikovať činnosti ako sken otvorených portov na serveri či slovníkový útok na zašifrovaný SSH kľúč, preto niektoré úrovne hier COVID-19 Truth Seeker a Uploaded Kobylka 3302 sú len čiastočne alebo nie sú vôbec klasifikované.

CCF

Jediný dostupný zdroj k referenčnému rámcu CCF, ktorí sme našli, je publikácia *Cybersecurity Curricula 2017* [8], pri klasifikácii hier sme si teda museli vystačiť s ňou. Keďže rámec CCF je stručnejší a má precíznejšiu štruktúru ako CyBOK, nepredstavovalo to až takú veľkú limitáciu ako u druhého zmieneného rámca. Pri snahe klasifikovať úrovne troch nám dostupných hier (tabuľky A.10, A.11 a A.12) sme boli úspešní pri všetkých úrovniach až na jednu, jednalo sa o tretiu úroveň hry Captain Slovakistan, v rámci ktorej má hráč nahrať web shell na webový server pre prístup k súboru s vlajkou. Nepodarilo sa nám nájsť vhodnú jednotku znalostí pre klasifikáciu tohto kroku útoku. Okrem tejto úrovne sme ale ešte narazili na niekoľko iných úrovní, ktoré sme klasifikovali, ale nie uspokojivo presne. Jednalo sa napr. o prvú úroveň v hrách COVID-19 Truth Seeker a Uploaded Kobylka 3302 (skenovanie otvorených portov, klasifikované ako *Network access control*) alebo tretiu úroveň prvej zmienenej hry a štvrtú úroveň druhej (exfiltrácia dát z napadnutého serveru, klasifikovaná ako *Insider threat*).

Cyber Kill Chain

Cyber Kill Chain bol zo všetkých porovnávaných rámcov najmenej obsiahly. Pokúsili sme sa pomocou neho klasifikovať jednotlivé hry (tabuľky A.13, A.14 a A.15), narazili sme ale na viacero problémov. Možné akcie fázy *Reconnaissance* nedokázali dostatočne presne opísať útok v príslušných úrovniach klasifikovaných hier, a tak sme sa museli uspokojiť s akciou *Discover internet-facing server*, ktorá ale nie vždy

dokázala presne zachytiť obsah danej úrovne. Ešte väčší problém ale predstavovalo to, že niektoré úrovne hier sme vôbec nedokázali popísať. Vo štvrtej úrovni hry COVID-19 Truth seeker musí hráč pomocou slovníkového útoku prelomiť ochranu SSH kľúča, ktorý získal v predchádzajúcej úrovni. Najbližším popisom tohto útoku je akcia *Collect user credentials*, čo sa ale zjavne nedá označiť za precízne vystihnutie situácie. Podobne je to pri druhej úrovni hry Uploaded Kobylka 3302. V nej hráč musí nájsť zneužiteľnú zraniteľnosť systému na základe skenu otvorených portov, ktorý vykonal v predchádzajúcej úrovni. Cyber Kill Chain neposkytuje na opis tejto úrovne žiadnu vhodnú akciu.

MITRE ATT&CK for Enterprise

Matica MITRE ATT&CK Matrix for Enterprise je nielen prehľadnou reprezentáciou rámca MITRE ATT&CK for Enterprise, ale aj užitočným nástrojom pri práci s týmto rámcom. Logicky sme sa ju teda rozhodli využiť pri klasifikácii našich hier. Naštudovanie obsahu matice je podstatne menej náročné ako u materiálov iných rámcov, ktoré sme predstavili v tejto práci. Samotnú klasifikáciu hier sme u tohto rámca zvládli bez väčších problémov (tabuľky A.16, A.17 a A.18). Hry dokázal popísať pomerne presne, ako izolovanú výnimku by sme mohli zmieniť techniku *Exploit Public-Facing Application* taktiky *Initial Access*, u ktorej by sme ocenili dodatočné podtechniky na upresnenie zneužitej zraniteľnosti.

Zhodnotenie

V tejto sekcii sme celkovo klasifikovali obsah úrovní troch hier vyvinutých pre KYPO šiestimi rôznymi spôsobmi. Tieto spôsoby môžeme rozdeliť do dvoch kategórií:

- klasifikácia opisujúca znalosti (rámce CyBOK, CCF a znalosti rámca NICE),
- klasifikácia opisujúca techniky útoku, ktorého úspešné prevedenie je cieľom hry (rámce MITRE ATT&CK for Enterprise, Cyber Kill Chain a do istej miery schopnosti rámca NICE).

Začnime rámcami zameranými na znalosti v kyberbezpečnosti. U týchto rámcov (predovšetkým CyBOK a CCF) sme bohužiaľ zaznamenali absenciu nástrojov na efektívnu prácu s rámcom (napr. vyhľadávanie v rámci či prehľadné zobrazenie rámca). Pri priraďovaní znalostí k jednotlivým úrovniam bolo potrebné určiť, čo presne budú znalosti u úrovní predstavovať. Mohlo sa jednať napr. o predpoklady alebo učebné výstupy daných úrovní. Oba tieto prístupy ale trpeli tým, že znalosti rámcov často predstavovali pomerne široké zoskupenie konceptov, kým hry sa dotýkali iba úzkej časti priradených znalostí. To by teda viedlo k tomu, že predpoklad úrovne hry by mohol byť zbytočne silne reštriktívny, nadhodnocoval by znalosti skutočne potrebné na zvládnutie úrovne. Oproti tomu učebný výstup úrovne by často bol precenením reálneho učebného prínosu úrovne. Rozhodli sme sa teda nepoužiť takéto konkrétne označenia a miesto toho k hrám priraďovať znalosti, ktoré nejakým spôsobom precvičujú. Napriek tomu ale predstavovala identifikácia znalostí jednotlivých úrovní pomerne komplexnú činnosť vyžadujúcu pokročilú znalosť použitého rámca. Obávali sme sa, že ak od vývojárov hier budeme vyžadovať navrhnutú klasifikáciu hier, značne to predĺži čas vývoja hier. Celkovo sme nadobudli dojem, že rámce znalostí sa zväčša orientovali na teoretické koncepty a defenzívne činnosti, čo bolo v kontraste s našimi hrami, ktoré sa snažia precvičovať schopnosti potrebné k prevedeniu počítačových útokov. Dospeli sme teda k záveru, že klasifikácia hier pomocou znalostí nie je vhodnou voľbou pre našu prácu.

Na druhej strane máme rámce techník útokov. V tomto prípade bola klasifikácia jednotlivých úrovní pomerne priamočiara, na rozdiel od použitia znalostí. Pomocou techník sme sa snažili charakterizovať útok danej úrovne. Rámec Cyber Kill Chain bol z pohľadu našej práce príliš jednoduchý, často sme ním nedokázali vôbec popísať úrovne alebo len nedostatočne presne. Podobne sme nie vždy dokázali kvalitne klasifikovať hry pri použití schopností rámca NICE kvôli jeho orientácii na zväčša defenzívne ladené pracovné pozície. Naproti tomu sme pri použití rámca MITRE ATT&CK for Enterprise nenarazili na žiadne väčšie problémy. Tento rámec má precíznu štruktúru, pri práci s ním je možné použiť jeho prehľadnú reprezentáciu, maticu MITRE ATT&CK Matrix for Enterprise, a výsledné klasifikácie hier pri jeho

použití patrili k najlepším výsledkom v našom porovnaní rámcov. Rozhodli sme sa ho teda v našej práci použiť.

3.2 Modifikácia definícií kyberbezpečnostných hier

Rozšírenie definícií o techniky útokov

Na opis obsahu hier sme sa rozhodli použiť referenčný rámec MITRE ATT&CK for Enterprise. Definície hier sme chceli rozšíriť o časť, ktorá by využitím taktík a (pod)techník tohto rámca opisovala útok, ktorého úspešné prevedenie je cieľom tejto hry. Túto časť sme sa rozhodli zakomponovať v rámci každej hernej úrovne, aby sa dala použiť aj napr. na ujasnenie priebehu útoku z hľadiska jednotlivých úrovní. Do JSON objektu reprezentujúceho úroveň sme u herných úrovní pridali kľúč *MITRE_techniques*, ku ktorému je ako hodnota priradený JSON objekt obsahujúci taktiky a (pod)techniky ako reťazce. Presný formát tohto objektu je nasledovný:

- Jeho kľúče majú formát:

  ```
  <identifikátor taktiky>.<identifikátor techniky>
  (.<identifikátor podtechniky>)
  ```

- Jeho hodnoty majú formát:

  ```
  <názov taktiky>.<názov techniky>
  (.<názov podtechniky>)
  ```

Časť s podtechnikou označená okrúhlou zátvorkou nie je povinná, pretože nie vždy existuje vhodná podtechnika vzhľadom na útok vyskytujúci sa v hre a niektoré techniky rámca MITRE ATT&CK for Enterprise dokonca ani žiadnu podtechniku nemajú. Do objektu boli zaradené aj názvy taktík a (pod)techník, aby používatelia pochopili obsah úrovne bez nutnosti vyhľadávania v matici. Na obrázku 3.1 môžeme vidieť príklad jednej úrovne modifikovanej definície s vyznačenou novou časťou.

Špecifikácia formátu riešení úrovní

Jeden z cieľov našej práce bolo extrahovať z definícií prác nástroje použité v jednotlivých hrách. Keďže techniky útokov, o ktoré sme definície hier rozšírili, presný nástroj použitý pri útoku nešpecifikujú,

```
  }, {
    "type" : "GameLevelImportDTO",
    "title" : "Exploiting the vulnerability",
    "max_score" : 100,
    "level_type" : "GAME_LEVEL",
    "order" : 2,
    "estimated_duration" : 10,
    "flag" : "always-verify-the-source",
    "content" : "You scanned the server and found out that it has **Webmin**
vulnerability **CVE-2019-15107**. Now you need to exploit it and access the
server. Flag is located as string in file **hoax.txt**.",
    "solution" : "* Exploit the server vurnerability with this commands:
\n`msfconsole`  \n`use exploit/unix/webapp/webmin_backdoor`  \n`set RHOSTS
10.1.26.9`  \n`set LHOST 10.1.135.83`  \n`set SSL true`  \n`exploit`  \n`cat
hoax.txt`\n",
    "solution_penalized" : true,
    "hints" : [ {
      "title" : "Which module should I use?",
      "content" : "You google more and find out that module **webmin_backdoor**
should help you.",
      "hint_penalty" : 20,
      "order" : 1
    }, {
      "title" : "Which options should I set?",
      "content" : "There is LHOST - that's you (but don't use 'localhost'!).
RHOST - that's server. Do not forget to allow negotiate SSL/TLS for outgoing
connections. See **show options** for more help.",
      "hint_penalty" : 20,
      "order" : 2
    }, {
      "title" : "What tool can I use?",
      "content" : "You google \"vulnerability exploitation tool\" and find out
that **Metasploit** exists.",
      "hint_penalty" : 20,
      "order" : 0
    } ],
    "incorrect_flag_limit" : 100,
    "MITRE_techniques" : {
      "TA0001.T1190" : "Initial Access.Exploit Public-Facing Application",
      "TA0008.T1021.004" : "Lateral Movement.Remote Services.SSH"
    }
  }, {
```

Obr. 3.1: Rozšírená definícia jednej hernej úrovne

neboli na tento účel vhodné. Po zvážení rôznych už existujúcich častí tréningových definícií sme sa rozhodli využiť časť obsahujúcu vzorové riešenie úrovne. Táto časť je hodnotou priradenou ku kľúču `solution` a jej súčasťou sú často aj príkazy, ktorými sa hráč dopracuje k vlajke danej úrovne. Navrhli sme nasledujúci formát riešení úrovní: jednotlivé príkazy v nich majú byť uzatvorené pomocou dvoch znakov `'`, vďaka čomu sa dajú z riešenia pomerne jednoducho automatizovane extrahovať. Na obrázku 3.2 môžeme vidieť príklad jednej hernej úrovne, ktorej riešenie je navrhovanom formáte.

3.3 Vizualizácia porovnaní techník viacerých hier

Dôležitou súčasťou našej práce bola vizualizácia porovnaní techník útokov jednotlivých hier. Vývojárom hier (a iným záujemcom) sme chceli ponúknuť viacero rôznych spôsobov vizualizácie porovnaní hier, aby sme nimi dokázali uspokojiť množstvo rôznorodých nárokov na tieto vizualizácie. Keďže techniky útokov hry dokopy tvoria množinu, hľadali sme spôsoby, ako znázorniť porovnanie množín.

O funkcionalite programu (respektíve programov) pre vizualizáciu porovnania hier sme mali pomerne jasnú predstavu. Program mal byť spustiteľný cez príkazovú riadku a parametrizovateľný pomocou vstupných argumentov a na vstup by bral definície dvoch či viacerých hier na porovnanie (cestu k súborom). V závislosti od spôsobu vizualizácie by program buď vygeneroval obrázok s požadovanou vizualizáciou alebo ju vykreslil na štandardný výstup (u niektorých spôsobov by mohol byť možný výber medzi oboma spôsobmi). Našou preferenciou bolo u výstupu pomocou generovaných súborov umožniť výstup vo formáte SVG pre vyššiu kvalitu výsledných obrázkov. Taktiež sme chceli poskytnúť možnosť špecifikovať cestu pre uloženie výsledného súboru. Dôležitou požiadavkou pre nás bola aj robustnosť, naším cieľom bolo, aby skripty korektne fungovali aj v budúcnosti bez ohľadu na zmenu externých faktorov (ako napr. knižnice).

Pred priblížením jednotlivých možných spôsobov vizualizácie porovnania hier by sme chceli zmieniť, že pre programovú časť práce sme si ako programovací jazyk zvolili Python. Našu voľbu ovplyvnila dobrá prenositeľnosť jazyka (jedná sa o interpretovaný jazyk), množstvo dostupných knižníc vďaka jeho vysokej popularite ako aj nízka

```
  }, {
    "type" : "GameLevelImportDTO",
    "title" : "Small inconvenience",
    "max_score" : 100,
    "level_type" : "GAME_LEVEL",
    "order" : 4,
    "estimated_duration" : 9,
    "flag" : "bobrules",
    "content" : "You are almost there! However, it turns out that key you have found
is encrypted, so you will first need to **crack** it before you can access computer
of your boss. Flag is the **cracked passphrase**.\n",
    "solution" : "* Prepare the files for 'John the Ripper' and then use him to crack
the key file:   \n`cp /usr/share/wordlists/rockyou.txt.gz .`        \n`gunzip
rockyou.txt.gz`     \n`cp /usr/share/john/ssh2john.py .`  \n`python ssh2john.py
id_rsa > hash`  \n`john hash -wordlist=rockyou.txt`   \n",
    "solution_penalized" : true,
    "hints" : [ {
      "title" : "Which auxiliary files should I use (and how)?",
      "content" : "You want to use wordlist **'/usr/share/wordlists/rockyou.txt.gz'**
that you first need to unpack with **'gunzip'** command and python script
**'/usr/share/john/ssh2john.py'** to **convert file** to suitable format for
John.\n",
      "hint_penalty" : 30,
      "order" : 2
    }, {
      "title" : "Where can I find auxiliary files for the cracker?",
      "content" : "Suitable wordlists are contained in directory
**/usr/share/wordlists** and you can find auxiliary tools for the cracker in the
directory **/usr/share/john**.\n",
      "hint_penalty" : 10,
      "order" : 1
    }, {
      "title" : "Which tool can I use to crack the key?",
      "content" : "There are multiple tools designed for this purpose, I would
recommend one, which name starts with **\"John\"**.\n",
      "hint_penalty" : 10,
      "order" : 0
    } ],
    "incorrect_flag_limit" : 100
  }, {
```

Obr. 3.2: Definícia jednej hernej úrovne s riešením v nami navrhnutom formáte

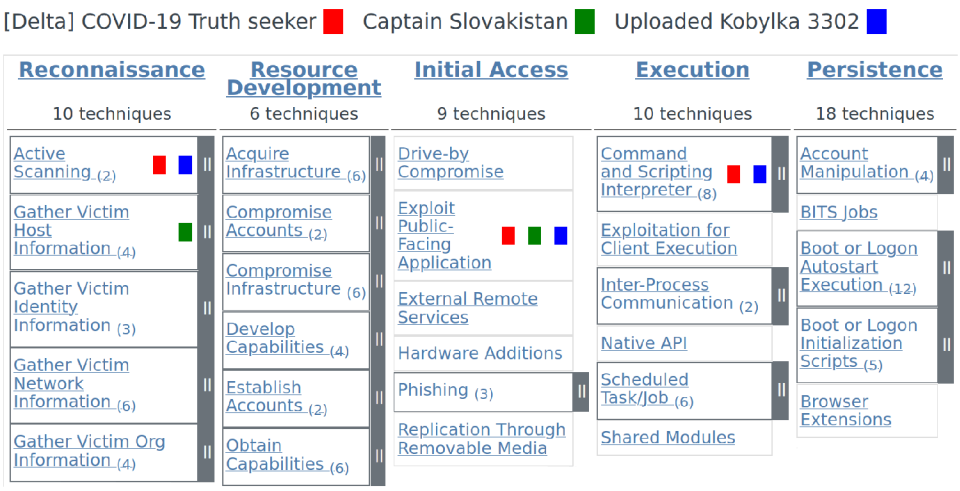
požiadavka našej práce na rýchlosť jazyka. Táto poznámka je dôležitá z toho dôvodu, že pri predstavení možných spôsobov vizualizácie porovnaní hier sme sa zamerali aj na možnú podporu tohto spôsobu zobrazenia z hľadiska nami vybraného jazyka a jeho knižníc.

Vizualizácia pomocou matice MITRE ATT&CK Matrix for Enterprise

Keďže k referenčnému rámcu MITRE ATT&CK for Enterprise je voľne dostupné zobrazenie vo forme matice MITRE ATT&CK Matrix for Enterprise, ktoré prehľadným spôsobom prezentuje obsah tohto rámca, rozhodli sme sa ho využiť ako jeden spôsob vizualizácie porovnania hier. Táto matica je definovaná ako HTML element, ktorý môžeme vidieť na stránke projektu MITRE ATT&CK. Naša predstava o vizualizácii bola nasledovná: každá hra, ktorá by bola súčasťou porovnania, by bola reprezentovaná značkou s vlastnou unikátnou farbou. Bunky matice reprezentujúce techniky útokov obsiahnuté v danej hre by boli označené značkou danej hry. Keďže technika môže byť zdieľaná viacerými hrami, mohla by byť označená aj viac ako jednou značkou. Program určený na realizáciu tejto vizualizácie by vygeneroval HTML súbor obsahujúci maticu MITRE ATT&CK Matrix for Enterprise rozšírenú o značky techník hier zahrnutých do porovnania. Vizuálnu reprezentáciu nášho návrhu môžeme vidieť na obrázku 3.3. Pretože referenčné rámce bázy znalostí MITRE ATT&CK v priebehu času prechádzajú úpravami, rôzne techniky sú do nich dodatočne pridávané alebo už existujúce techniky môžu byť odobrané, bolo pre nás podstatné túto vizualizáciu generovať takým spôsobom, aby nezávisel na konkrétnej verzii referenčného rámca.

Vizualizácia pomocou Vennových diagramov

Agregované techniky útokov priradené k úrovniam jednej hry predstavujú množinu. Jednou z možností, ako vizuálne porovnať viacero takýchto množín, predstavujú Vennove diagramy. Každá množina je v ňom reprezentovaná priehľadným útvarom. Prieniky útvarov znázorňujú prvky zdieľané množinami reprezentovanými danými útvarmi [15]. Zvykne sa k nim uvádzať číslo označujúce kardinalitu (veľkosť) daného prieniku. Príklad Vennovho diagramu troch mno-

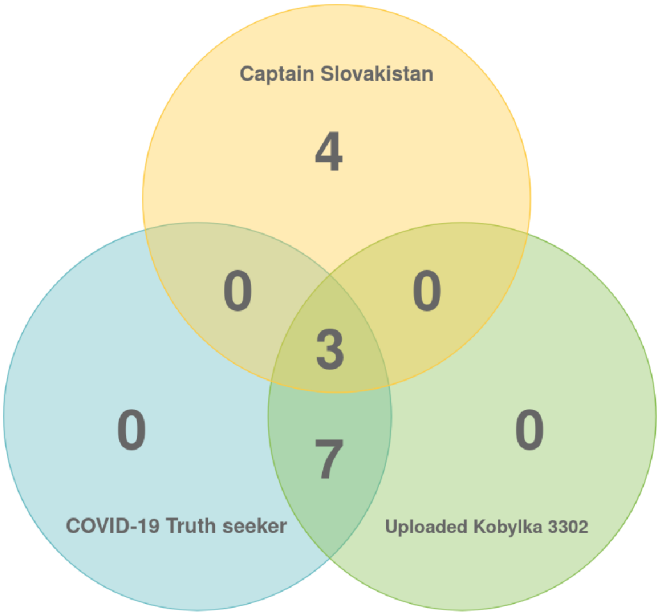


Obr. 3.3: Návrh vizualizácie porovnania hier využitím matice MITRE ATT&CK Matrix for Enterprise

žín môžeme vidieť na obrázku 3.4. Vennove diagramy predstavujú zrozumiteľný spôsob zobrazenia prieniku viacerých množín, ale len pre obmedzený počet množín. Pri piatich alebo viacerých množinách prestáva byť diagram prehľadný [15]. Existujú však rôzne zobrazenia vychádzajúce z Vennových diagramov, ktoré sa snažia tento problém vyriešiť. Predstavíme dve z nich: zobrazenie UpSet a Supervenn.

Vizualizácia pomocou zobrazenia UpSet

Základným stavebným blokom zobrazenia UpSet [16] sú prieniky množín. Každý stĺpec v hlavnej časti zobrazenia je ekvivalentný jednej oblasti Vennovho diagramu (prázdne oblasti sú zo zobrazenia väčšinou vynechané), riadky reprezentujú porovnávané množiny. Príslušnosť množiny v danom stĺpci (prieniku) je reprezentovaná krúžkom nachádzajúcim sa v tomto stĺpci a v riadku prislúchajúcom tejto množine: ak je označený, prvky prieniku patria do množiny, inak nie. Nad krúžkami je zobrazená kardinalita jednotlivých prienikov. Súčasťou zobrazenia môže byť aj informácia o kardinalite jednotlivých množín, zvykne sa uvádzať vedľa názvu objektov, ktoré sú reprezentované ako množiny. Prípadné prepojenie označených krúžkov nemá z hľadiska



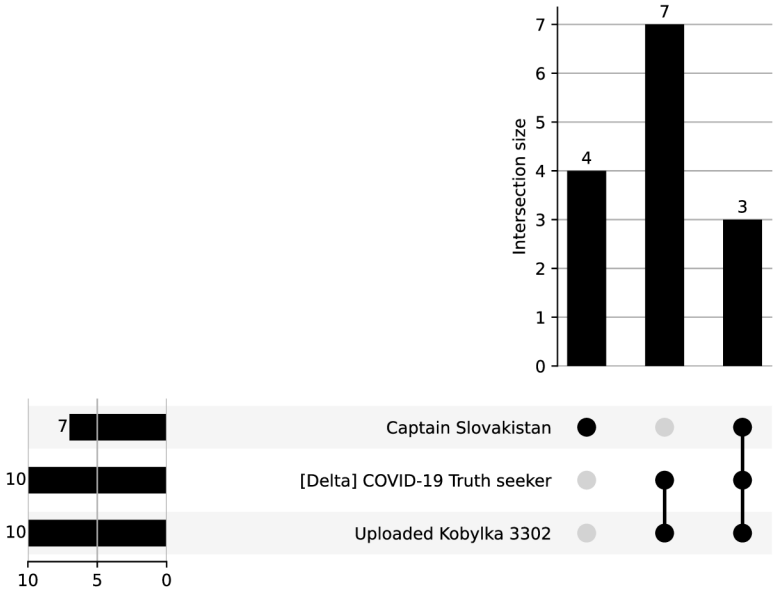
Obr. 3.4: Vizualizácia porovnania troch množín (hier) pomocou Vennovho diagramu

zobrazenia žiadny špeciálny význam. Na obrázku 3.5 môžeme vidieť príklad porovnania troch hier pomocou zobrazenia UpSet.

Existujú dve knižnice umožňujúce vizualizáciu porovnania množín pomocou zobrazenia UpSet v jazyku Python: *pyUpSet* [17] a *UpSetPlot* [18]. Obe knižnice sú nadstavbou široko rozšírenej knižnice *matplotlib* [19] a aj ich použitie je podobné. Podporujú výstup aj vo formáte SVG. Knižnica pyUpSet je dostupná pod MIT licenciou, kým UpSetPlot pod licenciou New BSD. Podstatným rozdielom ale je, že knižnica UpSetPlot je aj v súčasnosti udržiavaná, kým posledná verzia pyUpSet bola zverejnená v roku 2016.

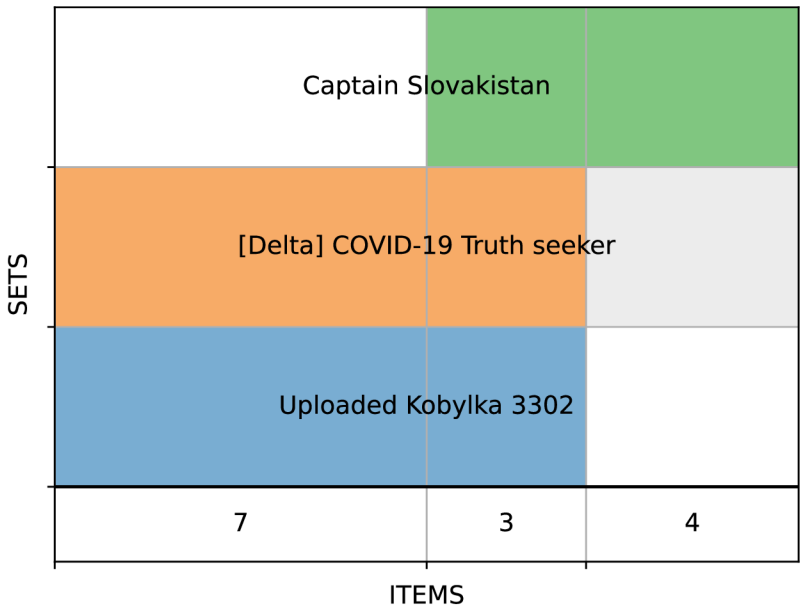
Vizualizácia pomocou zobrazenia Supervenn

Vizualizovať porovnanie viacerých množín umožňuje aj knižnica jazyka Python *supervenn* [20], ktorá je taktiež nadstavbou knižnice matplotlib. Nepodarilo sa nám nájsť žiadne iné materiály k tomuto zobrazeniu, rozhodli sme sa ho teda označovať ako Supervenn. Zobrazenie



Obr. 3.5: Vizualizácia porovnania troch hier pomocou zobrazenia UpSet

Supervenn vizualizuje porovnanie množín pomocou tabuľky. Riadky tejto tabuľky predstavujú množiny, stĺpce prieniky množín. Množina pozostáva z farebne vyznačených častí riadku, ktoré zároveň určujú, prienik ktorých množín daný stĺpec reprezentuje. Na spodku zobrazenia sú uvedené veľkosti príslušných prienikov. Knižnica Supervenn je dostupná pod licenciou MIT a je udržiavaná. Za zmienku taktiež stojí, že používa kombinatorický optimalizačný algoritmus pre usporiadanie stĺpcov (prienikov množín), ktoré minimalizuje fragmentáciu riadkov na oddelené vyfarbené bloky. Príklad porovnania troch množín (hier) pomocou zobrazenia Supervenn môžeme vidieť na obrázku 3.6. Knižnica supervenn podporuje výstup vo formáte SVG.



Obr. 3.6: Vizualizácia porovnania troch hier pomocou zobrazenia Supervenn

Vizualizácia pomocou tabuľky Jaccardových koeficientov

Ako bolo spomenuté už skôr, agregované techniky útokov jednej hry predstavujú množinu. Medzi štatistické údaje, pomocou ktorých môžeme porovnať (konečné) množiny, patrí aj Jaccardov koeficient [21]. Jaccardov koeficient dvoch množín S a T je definovaný ako podiel veľkosti prieniku a veľkosti zjednotenia týchto množín, formálne zapísané: $J(S,T) = \frac{S \cap T}{S \cup T}$. V prípade, že množiny S aj T sú prázdne, kladieme $J(S,T) = 1$. Platí teda $0 \leq J(S,T) \leq 1$. Hoci pomocou Jaccardovho koeficientu je možné porovnať iba dve množiny, pre viacero množín je možné zostrojiť tabuľku Jaccardových koeficientov. Pre k množín sa jedná o tabuľku, ktorá má k stĺpcov a k riadkov, kde hodnota bunky v n. riadku a m. stĺpci, $1 \leq n, m \leq k$, je rovná Jaccardovmu koeficientu n. a m. množiny. Tabuľka 3.1 predstavuje návrh tabuľky

Tabuľka 3.1: Návrh tabuľky Jaccardových koeficientov

	Hra 1	Hra 2	Hra 3
Hra 1	1	0.25	0.75
Hra 2	0.25	1	0
Hra 3	0.75	0	1

Jaccardových koeficientov pre tri hry. Zjavne je každá hra sama so sebou totožná, preto tabuľka na diagonále z ľavého horného rohu do pravého dolného rohu obsahuje samé jednotky. V uvedenom návrhu sú do veľkej miery podobné hry 1 a 3, kým hry 2 a 3 sú úplne odlišné. Natívne knižnice jazyka Python poskytujú dostatočné prostriedky pre implementáciu programu generujúceho tabuľku Jaccardových koeficientov s výstupom na štandardný výstup bez potreby použitia knižníc tretích strán.

3.4 Extrakcia a spracovanie dát z definícií hier

Cieľom našej práce bolo porovnanie kyberbezpečnostných hier na základe ich definícií. Samotné porovnanie v sebe zahŕňa niekoľko krokov. Prvým krokom je extrakcia informácií, na ktorých je porovnanie založené, z definícií hier. Nasleduje spracovanie extrahovaných informácií do vhodnej dátovej štruktúry. Na koniec je potrebné prezentovať porovnanie hier zrozumiteľným spôsobom pre používateľa, ktorý chce dané hry porovnať. V tejto sekcii sa zameriame na prvé dva kroky tohto procesu.

Programy na prácu s technikami útokov hier

Základ programovej časti našej práce sú programy na extrakciu a porovnanie techník útokov z definícií hier. Náš návrh počítal s dvoma programami. Prvý by slúžil na extrakciu techník z jednej alebo viacerých hier, pomocou druhého by sa dali porovnať dve a viac hier. Oba programy by boli spustiteľné cez príkazovú riadku a parametrizovateľné pomocou vstupných argumentov programu. Na vstupe by brali cesty k súborom predstavujúcim definície hier. Pre korektnú funkčnosť programov by bolo nutné, aby tieto definície hier obsahovali nami navrhnuté rozšírenie herných úrovní o techniky útokov. Výstup prog-

ramov by bol realizovaný pomocou štandardného výstupu. Program na extrakciu by vypisoval extrahované techniky útokov, kým program na porovnanie hier by vypisoval textovú reprezentáciu dátovej štruktúry určenej na porovnanie hier. Súčasne by ale oba programy poskytovali svoju funkcionalitu aj iným programom cez funkčné rozhranie, napr. samotný program na porovnanie hier pomocou techník by využíval funkcionalitu programu na extrahovanie techník. Programy na vizualizáciu porovnaní hier by zas využívali program na porovnanie hier.

Programy na prácu s príkazmi hier

Ako sme zmienili v sekcii 3.2, medzi ciele našej práce patrí aj extrahovanie a prezentovanie nástrojov obsiahnutých v hrách. Na tento účel sme navrhli formát riešení herných úrovní definícií hier, ktorý je možné spracovávať automatizovane.

Náš návrh bol v podstate totožný ako v predchádzajúcej podsekcii. Znova sme navrhli rozdelenie na dva programy, prvý na extrakciu príkazov z hier a druhý na ich porovnanie. Naše funkčné požiadavky na programy boli rovnaké ako na programy pracujúce s technikami hier, akurát namiesto požiadavky na obsah novej časti s technikami útokov bola na definície kladená požiadavka, aby formát riešení herných úrovní zodpovedal skôr spomenutej špecifikácii.

V našej práci sme zvážili aj vizualizáciu porovnania nástrojov (príkazov) hier ako doplnok vizualizácie porovnania hier na základe ich techník útokov. Zásadným problémom bolo ale v tomto ohľade to, že v niektorých hrách riešenia úrovní neobsahovali takmer žiadne príkazy, pretože hra nebola orientovaná na príkazovú riadku, ale napr. na internetový prehliadač a web shell. Tieto hry by teda neboli porovnateľné na základe ich príkazov, preto sme sa rozhodli od tohto nápadu upustiť.

4 Implementácia

V tejto kapitole predstavíme programovú časť práce, teda programy a knižnice, ktoré sme na základe nášho návrhu implementovali, spolu s programom umožňujúcim jednoduchú konfiguráciu prostredia spĺňajúceho predpoklady pre korektnú funkcionalitu týchto programov.

4.1 Programy na vizualizáciu porovnaní hier

Ako prvú predstavíme implementáciu možností vizualizácie porovnania techník útokov viacerých hier predstavených v predchádzajúcej kapitole.

Vizualizácia pomocou matice MITRE ATT&CK Matrix for Enterprise

Vizualizáciu porovnania hier pomocou matice MITRE ATT&CK Matrix for Enterprise sprostredkúva program `technique_visualizer_matrix.py`. Narozdiel od iných programov v našej práci nebola implementácia tohto programu priamočiara, pretože v rámci procesu jeho vývoja sme zvážili rôzne spôsoby jeho implementácie. Prvá verzia programu využívala knižnicu *Beatiful Soup* [22], ktorá umožňuje efektívnu prácu s HTML dokumentami, a pomocou nej sme modifikovali lokálne uloženú kópiu matice MITRE ATT&CK Matrix for Enterprise. Hoci tento prístup produkoval požadovaný výsledok, rozhodli sme sa od neho upustiť, pretože nebol robustný. Referenčné rámce bázy znalostí MITRE ATT&CK sú priebežne upravované, pridávajú sa do nich nové techniky a rôzne staré techniky môžu byť označené za zastarané a vylúčené z príslušných rámcov. Lokálna kópia matice vo formáte HTML by teda pomerne rýchlo mohla byť neaktuálna a nezodpovedať súčasnej podobe matice. Tento problém by mohol byť riešený pravidelnou aktualizáciou kópie, to sme však považovali za zbytočné zaťažovanie používateľov týchto programov, ktoré by bolo náchylné k používaniu neaktuálnej verzie uvedenej matice.

Finálna verzia programu získava techniky útokov priamo z oficiálneho serveru projektu MITRE ATT&CK a následne prezentuje

výslednú maticu na základe pripravenej šablóny. Pre komunikáciu so serverom sa využíva knižnica *taxii2-client* [23]. Na reprezentáciu taktík a (pod)techník rámca bázy MITRE ATT&CK sa používa špeciálny formát STIX 2 (Structured Threat Information Expression 2), s ktorým v jazyku Python pracujeme pomocou knižnice *stix2* [24]. Využitím týchto knižníc teda zo serveru získame súčasné taktiky a im priradené techniky. Na vytvorenie HTML reprezentácie matice náš program využíva šablónovací jazyk *Jinja2*, ktorý umožňuje rozšíriť HTML dokument o príkazy jazyka Python a pomocou nich vygenerovať časti dokumentu. Časť šablóny, ktorú sme na tento účel použili, môžeme vidieť na obrázku 4.1. Na prácu s pripravenou šablónou jazyka Jinja2 používame knižnicu *Jinja2* [25], pomocou ktorej vygenerujeme HTML reprezentáciu matice, kde sú časti šablóny s príkazmi nahradené ich výstupom. Táto implementácia programu sa teda dokáže prispôsobiť zmenám obsahu rámca MITRE ATT&CK for Enterprise.

```
                <thead>
                  <tr>
                  {% for tactic in tactics %}
                    <td class="tactic name">
                      <a data-toggle="tooltip" data-placement="top" title="{{
tactic["external_references"][0]["external_id"] }}">{{ tactic["name"] }}</a>
                    </td>
                  {% endfor %}
                  </tr>
                  <tr>
                  {% for tactic in tactics %}
                    <td class="tactic count">
                      {{ techniques[loop.index0]|length }} techniques
                    </td>
                  {% endfor %}
                  </tr>
                </thead>
```

Obr. 4.1: Časť Jinja2 šablóny použitej na generovanie výstupu programu *technique_visualizer_matrix.py*, tabuľky reprezentujúcej maticu MITRE ATT&CK Matrix for Enterprise.

Pri implementácii vizualizácie matice sme sa rozhodli do matice nezaradiť podtechniky pre vyššiu prehľadnosť výsledného zobrazenia. Príklad časti vygenerovanej matice môžeme vidieť na obrázku 4.2.

Okrem špecifikovania súborových ciest k definíciám porovnávaných hier ešte program `technique_visualizer_matrix.py` pomocou

Games: Captain Slovakistan [Delta] COVID-19 Truth seeker

Reconnaissance	Resource Development	Initial Access	Execution
10 techniques	7 techniques	9 techniques	12 techniques
Active Scanning	Acquire Infrastructure	Drive-by Compromise	Command and Scripting Interpreter
Gather Victim Host Information	Compromise Accounts	Exploit Public-Facing Application	Container Administration Command
Gather Victim Identity Information	Compromise Infrastructure	External Remote Services	Deploy Container
Gather Victim Network Information	Develop Capabilities	Hardware Additions	Exploitation for Client Execution
Gather Victim Org Information	Establish Accounts	Phishing	
Phishing for Information	Obtain Capabilities		

Obr. 4.2: Časť výstupu programu `technique_visualizer_matrix.py` pri použití štandardného režimu zobrazenia.

argumentov umožňuje špecifikovať názov súboru s výstupom a možnosť jednofarebného vykreslenia, pri ktorom sú techniky útokov všetkých hier agregované dokopy a miesto označovania pomocou bodov sú bunky techník zahrnutých v niektorých zo špecifikovaných hier vyfarbené žltou farbou. Toto zobrazenie zrejme nie je vhodné pre porovnanie hier, ale umožňuje používateľovi získať vizuálnu reprezentáciu pokrytia techník matice technikami špecifikovaných hier. Príklad výstupu programu pri použití tejto možnosti môžeme vidieť na obrázku 4.3.

Pomocou jednofarebného zobrazenia je možné zobraziť ľubovoľný počet hier. Štandardný spôsob zobrazenia je obmedzený na 15 hier, už pri tomto počte hier môžu byť bunky matice značne široké a výsledné zobrazenie pomerne neprehľadné. Pri vývoji tohto spôsobu vizualizácie sme tiež zvážili rôzne spôsoby špecifikácie podoby pridaných prvkov vygenerovanej matice. Pôvodná verzia programu používala jednoduchší spôsob štylizácie využitím HTML atribútu `style`

Games: Captain Slovakistan, [Delta] COVID-19 Truth seeker

Reconnaissance	Resource Development	Initial Access	Execution
10 techniques	7 techniques	9 techniques	12 techniques
Active Scanning	Acquire Infrastructure	Drive-by Compromise	Command and Scripting Interpreter
Gather Victim Host Information	Compromise Accounts	Exploit Public-Facing Application	Container Administration Command
Gather Victim Identity Information	Compromise Infrastructure	External Remote Services	Deploy Container
Gather Victim Network Information	Develop Capabilities	Hardware Additions	Exploitation for Client Execution
Gather Victim Org Information	Establish Accounts	Phishing	Inter-Process Communication
Phishing for Information	Obtain Capabilities		

Obr. 4.3: Časť výstupu programu `technique_visualizer_matrix.py` pri použití jednofarebného režimu zobrazenia.

priamo vo vygenerovanom HTML súbore. Tento prístup ale značne komplikoval možnosť dodatočnej modifikácie štýlu týchto prvkov (napr. ich farby), pretože štýl je duplikovaný u každého prvku, preto sme od neho upustili. Farebné schémy oboch spôsobov zobrazenia je teda možné upraviť modifikáciou priloženého CSS súboru `style.css`, ktorý je mierne modifikovaný CSS súbor originálnej matice. Na začiatku súboru sa pod triedou `root` nachádzajú premenné špecifikujúce farbu značiek jednotlivých hier (`--game*`), farbu buniek v jednofarebnom režime špecifikuje premenná `--single-color`.

Vizualizácia pomocou zobrazenia UpSet

Pri implementácii vizualizácie pomocou zobrazenia UpSet sme sa rozhodli použiť knižnicu *UpSetPlot*, keďže je narozdiel od knižnice *pyUpSet* udržiavaná. Pomocou vstupných argumentov programu

`technique_visualizer_upset.py`, ktorý toto zobrazenie implementuje, je okrem ciest k súborom s hernými definíciami možné špecifikovať cestu pre vygenerovaný súbor. Ďalej je možné špecifikovať to, či sa pri porovnaní jednotlivých hier majú ignorovať podtechniky techník útokov, ktoré sú k nim priradené. Podporované formáty pre vygenerované súbory sú SVG a PNG. Výsledná vizualizácia je totožná s návrhom uvedeným v predchádzajúcej kapitole tejto práce (obrázok 3.5).

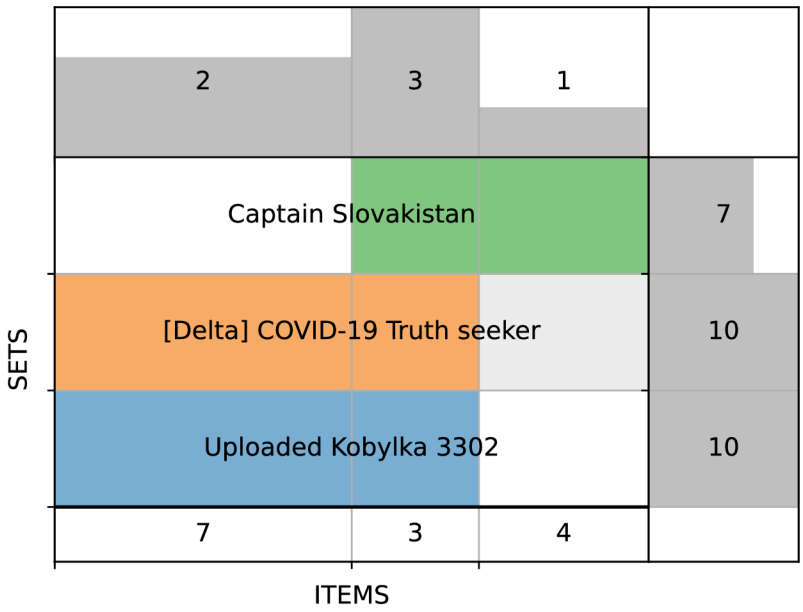
Vizualizácia pomocou zobrazenia Supervenn

Vizualizáciu pomocou zobrazenia Supervenn v našej práci poskytuje program `technique_visualizer_supervenn.py`. Pri jeho implementácii sme použili v súčasnosti jedinú dostupnú knižnicu špecializovanú na toto zobrazenie, knižnicu *supervenn*. Pomocou vstupných argumentov je u tohto programu okrem ciest k súborom s hernými definíciami možné špecifikovať cestu pre vygenerovaný súbor, možnosť pri porovnaní ignorovať podtechniky špecifikované pri technikách útokov jednotlivých hier a možnosť rozšíriť výsledne zobrazenie o dodatočné informácie. Takéto rozšírené zobrazenie môžeme vidieť na obrázku 4.4. Horný riadok obsahuje informáciu o počte množín (hier) obsiahnutých v prieniku reprezentovanom daným stĺpcom a prvý stĺpec sprava špecifikuje počet techník hry reprezentovanej daným riadkom. Z obrázku podľa očakávaní vidno, že hry COVID-19 Truth seeker a Uploaded Kobylka 3302 sú z hľadiska obsiahnutých techník útokov úplne totožné, kým hra Captain Slovakistan je od nich pomerne odlišná. Rovnako ako u programu `technique_visualizer_upset.py` sú podporované formáty pre generované súbory SVG alebo PNG.

Vizualizácia pomocou tabuľky Jaccardových koeficientov

Vizualizáciu pomocou tabuľky Jaccardových koeficientov implementuje program `technique_visualizer_jaccard.py`. Tento program poskytuje dve možnosti pre formát výstupu:

- Výpis na štandardný výstup. Tento výpis pozostáva zo zoznamu názvov porovnávaných hier, textovej reprezentácie tabuľky Jaccardových koeficientov a na koniec obsahu tejto tabuľky v CSV



Obr. 4.4: Výstup programu `technique_visualizer_supervenn.py` pri špecifikovaní dodatočných informácii v zobrazení a ignorovaní podtechník.

formáte. Príklad výstupu programu pri použití tohto formátu môžeme vidieť na obrázku 4.5. Táto možnosť sa použije implicitne (keď nie je špecifikovaná cesta pre uloženie vygenerovaného súboru).

- Vygenerovaný obrázok obsahujúci tabuľku Jaccardových koeficientov spolu so zoznamom porovnávaných hier. Táto možnosť sa použije pri špecifikácii cesty pre uloženie vygenerovaného obrázku. Pri vytvorení obrázku sa využívajú knižnice *matplotlib* a *numpy* [26]. Príklad vygenerovaného výstupu môžeme vidieť na obrázku 4.6. Opäť je podporovaným formátom pre vygenerovaný obrázok SVG alebo PNG.

```
Compared games:
1 - Captain Slovakistan
2 - [Delta] COVID-19 Truth seeker
3 - Uploaded Kobylka 3302

+------+------+------+------+
|      |Game 1|Game 2|Game 3|
+------+------+------+------+
|Game 1| 1.000| 0.214| 0.214|
+------+------+------+------+
|Game 2| 0.214| 1.000| 1.000|
+------+------+------+------+
|Game 3| 0.214| 1.000| 1.000|
+------+------+------+------+

,Game 1,Game 2,Game 3
Game 1,1.000,0.214,0.214
Game 2,0.214,1.000,1.000
Game 3,0.214,1.000,1.000
```

Obr. 4.5: Výstup programu `technique_visualizer_jaccard.py` na štandardný výstup

Pomocou vstupných argumentov sa u tohto programu špecifikujú cesty k herným definíciám, ďalej sa môže špecifikovať súborová cesta pre výstup (pričom tento argument rozhoduje o formáte výstupu) a možnosť ignorovať pri porovnaní hier podtechniky.

4.2 Programy na extrakciu a spracovanie dát z definícií hier

Programy na prácu s technikami útokov hier

Na základe nášho návrhu sme pre prácu s technikami útokov hier implementovali dva programy, ktoré okrem vlastnej funkcionality poskytujú funkčné rozhranie ďalším programom našej práce: `technique_extractor.py` a `technique_comparator.py`. Oba programy ako vstupné argumenty berú cesty k súborom s definíciami porovnaných hier, pričom je ale ešte možné pomocou dodatočného argumentu špecifikovať, či pri spracovaní ignorovať podtechniky

	Game 1	Game 2	Game 3
Game 1	1.000	0.214	0.214
Game 2	0.214	1.000	1.000
Game 3	0.214	1.000	1.000

Game 1 - Captain Slovakistan
Game 2 - Uploaded Kobylka 3302
Game 3 - [Delta] COVID-19 Truth seeker

Obr. 4.6: Výstup programu `technique_visualizer_jaccard.py` ako vygenerovaný obrázok

techník útokov danej hry (implicitne sa berú do úvahy). Program `technique_extractor.py` pri spustení na štandardný výstup vypíše techniky útokov extrahované z herných definícií zadaných ako vstupné argumenty vo formáte, ktorý môžeme vidieť na obrázku 4.7. Výstup je členený podľa úrovní jednotlivých hier, pre každú techniku útoku sa vypíše oddelene identifikátor, taktika, technika (a prípadne podtechnika, ak nie sú ignorované) rámca MITRE ATT&CK for Enterprise. Iným programom umožňuje extrahovať techniky útokov z definícií hier ako množinu identifikátorov.

```
Processed game: Captain Slovakistan

Level 0: Introduction
Not a game level, skipping...

Level 1: Find the insecure internal web page of the army
Techniques of this levels:
TA0043.T1594       Reconnaissance      Search Victim-Owned Websites
```

Obr. 4.7: Časť výstupu programu `technique_extractor.py`

Formát výstupu programu `technique_comparator.py` môžeme vidieť na obrázku 4.8. Pri porovnaní techník útokov hier tento program vytvára vnorený slovník: kľúčom na prvej úrovni je identifikátor taktiky, na druhej úrovni identifikátor techniky a na tretej úrovni identifikátor podtechniky. Hodnotou na poslednej úrovni je množina čísel, ktoré predstavuje indexy hier, ktorým bola daná technika útoku priradená. V prípade, že u danej techniky nie je špecifikovaná podtechnika, sa ako kľúč sa použije prázdny reťazec. Ak je pomocou argumentu nastavené ignorovanie podtechník, vytvorený slovník neobsahuje tretiu úroveň zanorenia a množiny indexov sú hodnotou už na druhej úrovni. Na výstup teda program `technique_comparator.py` vracia textovú reprezentáciu tejto dátovej štruktúry. Na jej vytvorenie sa využíva natívna knižnica jazyka Python *json* [27]. Vo výpise programu je možné usporiadať poradie položiek podľa kľúčov slovníka použitím dodatočného argumentu pri volaní programu.

```
Compared games:
1 - [Delta] COVID-19 Truth seeker
2 - Uploaded Kobylka 3302

Techniques found in games:
{
    "TA0042": {
        "T1588": {
            "006": [
                1,
                2
            ]
        }
    },
```

Obr. 4.8: Časť výstupu programu `technique_comparator.py` na vstupe s dvomi hrami, ktoré obsahujú rovnakú taktiku, techniku a podtechniku.

Programy na prácu s príkazmi hier

Programy `command_extractor.py` a `command_comparator.py` sú do veľkej miery podobné ich ekvivalentom pre prácu s technikami útokov, či už použitím alebo formátom výstupu. Mimo ciest k súborom

s definíciami je u prvého zmieneného programu možné ešte špecifikovať dodatočný argument, ktorý zo získaných príkazov extrahuje názov príkazu. Príkaz uvedený v riešení zbaví argumentov a prípadne odstráni prompt na jeho začiatku. Program `command_extractor.py` na výstup postupne po jednotlivých úrovniach vypíše príkazy zadaných hier, ako môžeme vidieť na obrázku 4.9. Na extrakciu príkazov z riešenia úrovni hier využíva regulárne výrazy za použitia natívnej knižnice jazyka Python *re* [28].

```
Processed game: [Delta] COVID-19 Truth seeker

Level 0: Introduction
Not a game level, skipping...

Level 1: Finding the path
Used commands:
nmap 10.1.26.9 --script=vuln
```

Obr. 4.9: Časť výstupu programu `command_extractor.py`

Podobne ako `technique_comparator.py` aj `command_comparator.py` pri porovnaní hier zadaných ako vstupné argumenty vytvára slovník, ktorý ale má jednoduchší formát. Jeho kľúčmi sú názvy príkazov (argumenty príkazov nám neprišlo vhodné pri porovnaní brať do úvahy) a hodnotou množiny indexov hier, u ktorých sa daný príkaz objavil pri riešení niektorej úrovne. Príklad výstupu tohto programu môžeme vidieť na obrázku 4.10. Rovnako sa aj v tomto prípade na vytvorenie textovej reprezentácie výsledného objektu porovnania používa knižnica *json*. Vo výpise programu je opäť možné usporiadať poradie položiek podľa kľúčov slovníka použitím dodatočného argumentu pri volaní programu.

4.3 Pomocné knižnice

Súčasťou programovej časti našej práce sú aj pomocné knižnice, ktoré sprostredkúvajú určitú funkcionalitu ostatným programom. Keďže každý program v našej práci berie na vstup ako argumenty cesty k definíciám hier, musí zo súborov s definíciami získať časti, ktoré sú podstatné pre jeho funkcionalitu. V snahe zabrániť duplikácii kódu sme

```
Compared games:
1 - [Delta] COVID-19 Truth seeker
2 - Uploaded Kobylka 3302

Commands found in games:
{
    "nmap": [
        1,
        2
    ],
    "msfconsole": [
        1,
        2
    ],
```

Obr. 4.10: Časť výstupu programu `command_comparator.py` na vstupe s dvoma hrami, pri riešení ktorých sa používajú dva rovnaké nástroje.

sa preto rozhodli do práce zakomponovať knižnicu `game_loader.py`, ktorá obsahuje definície dvoch funkcií určených na načítavanie hier. Základnou funkciou umožňujúcou načítanie jednej hernej definície je funkcia `load_game`. Táto funkcia na vstup berie cestu k súboru s definíciou hry a vracia dvojicu: názov hry a pole obsahujúce úrovne danej hry (jedná sa o JSON objekty prevedené na slovníky). Zároveň je táto funkcia zodpovedná aj za ošetrenie chýb ako je napr. nesprávna cesta k súboru alebo nesprávny formát definície, pričom pri týchto chybách dôjde k vypísaniu chybovej správy a ukončeniu programu. Predstavená knižnica obsahuje ešte rozšírenie tejto funkcie pre načítavanie viacerých hier naraz, funkciu `load_games`.

Rovnako je potrebné pri všetkých programoch spracovať dodatočné vstupné argumenty od používateľa, napr. názov súboru pre výstup programu. Pre tento účel sme použili natívnu knižnicu jazyka Python *argparse* [29]. U niektorých programov bolo možné špecifikovať úplne rovnaké dodatočné argumenty, preto sme sa rozhodli vyčleniť spracovanie týchto argumentov do samostatnej knižnice `argument_parser.py`, aby sme zabránili prípadnej duplikácii kódu.

4.4 Konfigurácia prostredia

Pre prácu s programami našej práce je požiadavkou inštalácia interpreteru jazyka Python, verzie 3.7 alebo vyššej. Súčasťou implementačnej časti práce je konfiguračný súbor `Pipfile` pre nástroj *Pipenv* [30]. Pipenv predstavuje veľmi užitočný nástroj, pomocou ktorého je možné vytvárať nezávisle virtuálne prostredia rôznych verzií jazyka Python so špecifikovanými knižnicami tretích strán bez ovplyvnenia systémovej inštalácie jazyka Python. Priložený konfiguračný súbor umožňuje jednoducho vytvoriť a nakonfigurovať virtuálne prostredie, ktoré spĺňa predpoklady programov našej práce. Pipenv je možné nainštalovať pomocou správcu balíkov jazyka Python *Pip*, ktorý je súčasťou inštalácie tohto jazyka od verzie 3.4. Programy v našej práci využívajú celkovo sedem nenatívnych knižníc, ktoré Pipenv pri použití nami dodaného konfiguračného súboru v rámci prostredia nainštaluje: numpy, stix2, taxii2-client, Jinja2, supervenn, UpSetPlot a matplotlib (táto knižnica je nevyhnutná pre korektnú funkcionalitu nielen viacerých programov ale aj knižníc supervenn a UpSetPlot, takže by bola nainštalovaná automaticky pri inštalácii týchto knižníc aj bez explicitnej zmienky, ale považovali sme za vhodné ju do `Pipfile` zaradiť pre prípad, keby ju novšie verzie týchto knižníc nevyužívali).

5 Záver

Hlavným cieľom našej práce bolo implementovať sadu programov, pomocou ktorých sa budú dať porovnať kyberbezpečnostné hry na základe ich definícií. Výsledok tohto porovnania by mohol byť užitočnou pomôckou pri výbere vhodných hier pre hráčov. Dali by sa pomocou neho odhaliť hry založené na rovnakej zraniteľnosti či type útoku a obmedziť tak opakovanie obsahu v hrách ponúknutých hráčovi. Programy sa taktiež mali dať použiť na prehľadné zobrazenie obsahu či nástrojov použitých vo viacerých hrách.

V práci sme najskôr bližšie predstavili kyberbezpečnostné hry vyvíjané v Kybernetickom polygóne, ktorými sme sa v práci zaoberali. Keďže definície súčasných hier neobsahujú precíznu špecifikáciu obsahu, na základe ktorej by sa dali porovnať, rozhodli sme sa v práci navrhnúť, ako by sa o ňu dali rozšíriť. Priblížili sme rôzne referenčné rámce zamerané na znalosti alebo techniky útokov v kyberbezpečnosti, ktoré by sa na tento účel dali použiť. Pomocou každého predstaveného rámca sme sa pokúsili klasifikovať obsah troch hier, pričom ako najvhodnejší rámec nám vyšiel *MITRE ATT&CK for Enterprise*. Navrhli sme teda rozšírenie definícií hier o časť obsahujúcu techniky útokov tohto rámca pre popis obsahu hier.

Ďalej sme predstavili návrh praktického výstupu práce. Prezentovali sme rôzne možnosti vizualizácie porovnania hier. Taktiež sme priblížili náš návrh extrakcie nástrojov použitých v daných hrách, ktorý ich získaval z príkazov uvedených v riešeniach daných hier. Praktickým výsledkom našej práce je zbierka programov spolu s popisom ich funkcionality a vzorovou ukážkou ich výstupu.

Ciele našej práce sa nám podarilo splniť. Nami vyvinuté programy umožňujú porovnať hry a odhaliť, ktoré hry sú do veľkej miery podobné alebo odlišné, zamerané na iné spôsoby útoku. Vizualizácia za použitia matice *MITRE ATT&CK Matrix for Enterprise* predstavuje prehľadný spôsob zobrazenia techník útokov obsiahnutých vo viacerých hrách. V rámci práce sme tiež vyvinuli program, ktorý umožňuje z príkazov riešení hier získať nástroje, ktoré hráči v rámci hier používajú.

Za podstatný výsledok práce považujeme aj rozšírenie definícií hier o techniky útokov. Táto nová časť definícií môže predstavovať uži-

točnú informáciu pre hráčov, ktorí chcú poznať obsah hry. Zaujímavá môže byť aj pre vývojárov, ktorým môže pomôcť získať lepší rozhľad v pokrytí rôznych oblastí kyberbezpečnosti dosiaľ vyvinutými hrami a pomôcť im zamerať sa pri vývoji nových hier na dosiaľ nepokryté oblasti kyberbezpečnosti.

Navrhnutý spôsob špecifikácie obsahu hier ale stále môže byť ešte vylepšený. Za zváženie napr. stojí rozšírenie rámca MITRE ATT&CK for Enterprise o dodatočné vlastné podtechniky, ktoré by presnejšie klasifikovali obsah niektorých úrovní. Nespokojný sme boli s absenciou podtechník napr. u techniky *Exploit Public-Facing Application*, pod ktorú v našich klasifikáciách spadá obídenie prihlásenia využitím útoku SQL injection, ale aj zneužitie zraniteľného softvéru servera pomocou útoku command injection. Do budúcna by sme tieto útoky chceli rozlíšiť aspoň na úrovni podtechník.

Pridané techniky útokov do hier ale považujeme za mocný nástroj, ktorý môže nájsť v budúcnosti aj ďalšie uplatnenia. Zaujímavým pokračovaním našej práce by mohla byť snaha prepojiť techniky s učebnými predpokladmi a snažiť sa tak u hier špecifikovať potrebné znalosti na ich zvládnutie. Hoci by to pravdepodobne predstavovalo značne náročnú prácu, pre hráčov by špecifikácia predpokladov u hier mohla byť veľmi hodnotnou pomôckou. Ďalšou potencionálnou výzvou pre budúcu prácu by mohlo logické usporiadanie techník útokov, napr. z hľadiska obtiažnosti ich prevedenia. Následne by sa mohli na základe ich usporiadania zoradiť aj hry pre čo najlepšiu skúsenosť pri ich hraní.

Z hľadiska výuky kyberbezpečnosti predstavuje dôležitú oblasť samozrejme aj obrana proti útokom. V súčasnosti sa už začali v Kybernetickom polygóne vyvíjať hry, ktoré obsahujú i úrovne zamerané na obranu ako napr. ošetrenie chýb v zraniteľnom systéme. Náš súčasný návrh zjavne nedokáže klasifikovať takéto herné úrovne. Súčasťou bázy MITRE ATT&CK, do ktorej patrí nami použitý rámec MITRE ATT&CK for Enterprise, je aj referenčný rámec *Enterprise Mitigations*, pokrývajúci techniky obrany v kyberbezpečnosti. Rozšírením našej práce by teda mohla byť aj klasifikácia defenzívnych úrovní kyberbezpečnostných hier pomocou tohto rámca.

A Klasifikácia hier pomocou predstavených rámcov

Prvý stĺpec tabuliek (Ú) označuje poradové číslo (hernej) úrovne hry.

Tabuľka A.1: Klasifikácia hry COVID-19 Truth seeker pomocou schopností rámca NICE

Ú	Schopnosť
1	Skill in using network analysis tools to identify vulnerabilities. (e.g., `fuzzing`, `nmap`, etc.). Skill in interpreting vulnerability scanner results to identify vulnerabilities.
2	Skill in using tools, techniques, and procedures to remotely exploit and establish persistence on a target.
3	*Neklasifikovateľné.*
4	*Neklasifikovateľné.*
5	Skill to identify sources, characteristics, and uses of the organization's data assets. Skill in reading, interpreting, writing, modifying, and executing simple scripts (e.g., PERL, VBS) on Windows and Unix systems (e.g., those that perform tasks like parsing large data files, automating manual tasks, and fetching/processing remote data). Skill in verifying the integrity of all files. (e.g., checksums, Exclusive OR, secure hashes, check constraints, etc.)

Tabuľka A.2: Klasifikácia hry Uploaded Kobylka 3302 pomocou schopností rámca NICE

Ú	Schopnosť
1	Skill in using network analysis tools to identify vulnerabilities. (e.g., `fuzzing`, `nmap`, etc.).
2	Skill in interpreting vulnerability scanner results to identify vulnerabilities.
3	Skill in using tools, techniques, and procedures to remotely exploit and establish persistence on a target.
4	Skill to identify sources, characteristics, and uses of the organization's data assets.
5	*Neklasifikovateľné.*
6	Skill to identify sources, characteristics, and uses of the organization's data assets.

Tabuľka A.3: Klasifikácia hry Captain Slovakistan pomocou schopností rámca NICE

Ú	Schopnosť
1	Skill in recognizing technical information that may be used for leads to enable remote operations (data includes users, passwords, email addresses, IP ranges of the target, frequency in DNI behavior, mail servers, domain servers, SMTP header information).
2	*Neklasifikovateľné.*
3	Skill in using tools, techniques, and procedures to remotely exploit and establish persistence on a target.
4	Skill in using databases to identify target-relevant information.
5	Skill to identify sources, characteristics, and uses of the organization's data assets.

Tabuľka A.4: Klasifikácia hry COVID-19 Truth seeker pomocou znalostí rámca NICE

Ú	Znalosť
1	Knowledge of networking and Internet communications fundamentals (i.e. devices, device configuration, hardware, software, applications, ports/protocols, addressing, network architecture and infrastructure, routing, operating systems, etc.). Knowledge of how to use network analysis tools to identify vulnerabilities. Knowledge of vulnerability information dissemination sources (e.g., alerts, advisories, errata, and bulletins).
2	Knowledge of structure, approach, and strategy of exploitation tools (e.g., sniffers, keyloggers) and techniques (e.g., gaining backdoor access, collecting/exfiltrating data, conducting vulnerability analysis of other systems in the network).
3	Knowledge of command-line tools (e.g., `mkdir`, `mv`, `ls`, `passwd`, `grep`). Knowledge of controls related to the use, processing, storage, and transmission of data.
4	Knowledge of attack methods and techniques (DDoS, brute force, spoofing, etc.).
5	Knowledge of remote access technology concepts. Knowledge of concepts for operating systems (e.g., Linux, Unix). Knowledge of command-line tools (e.g., `mkdir`, `mv`, `ls`, `passwd`, `grep`). Knowledge of authentication, authorization, and access control methods.

Tabuľka A.5: Klasifikácia hry Uploaded Kobylka 3302 pomocou znalostí rámca NICE

Ú	Znalosť
1	Knowledge of networking and Internet communications fundamentals (i.e. devices, device configuration, hardware, software, applications, ports/protocols, addressing, network architecture and infrastructure, routing, operating systems, etc.). Knowledge of network tools (e.g., `ping`, `traceroute`, `nslookup`).
2	Knowledge of how to use network analysis tools to identify vulnerabilities. Knowledge of vulnerability information dissemination sources (e.g., alerts, advisories, errata, and bulletins).
3	Knowledge of structure, approach, and strategy of exploitation tools (e.g., sniffers, keyloggers) and techniques (e.g., gaining backdoor access, collecting/exfiltrating data, conducting vulnerability analysis of other systems in the network).
4	Knowledge of command-line tools (e.g., `mkdir`, `mv`, `ls`, `passwd`, `grep`).
5	Knowledge of controls related to the use, processing, storage, and transmission of data. Knowledge of attack methods and techniques (DDoS, brute force, spoofing, etc.).
6	Knowledge of remote access technology concepts. Knowledge of concepts for operating systems (e.g., Linux, Unix). Knowledge of authentication, authorization, and access control methods.

Tabuľka A.6: Klasifikácia hry Captain Slovakistan pomocou znalostí rámca NICE

Ú	Znalosť
1	Knowledge of concepts related to websites (e.g., web servers/pages, hosting, DNS, registration, web languages such as HTML).
2	Knowledge of system and application security threats and vulnerabilities (e.g., buffer overflow, mobile code, cross-site scripting, Procedural Language/Structured Query Language [PL/SQL] and injections, race conditions, covert channel, replay, return-oriented attacks, malicious code).
3	*Neklasifikovateľné.*
4	Knowledge of query languages such as SQL (structured query language). Knowledge of database systems.
5	Knowledge of query languages such as SQL (structured query language). Knowledge of database systems.

Tabuľka A.7: Klasifikácia hry COVID-19 Truth seeker pomocou rámca CyBOK

Ú	Oblasť poznania	Dodatočná klasifikácia
1	Network Security	Transport-Layer Security
2	Web & Mobile Security	Server Side Vulnerabilities and Mitigations (Injection Vulnerabilities: Command Injections)
3	Adversial Behaviour	A Characterisation of Adversaries
4	*Neklasifikovateľné*	
5	Adversial Behaviour	A Characterisation of Adversaries

Tabuľka A.8: Klasifikácia hry Uploaded Kobylka 3302 pomocou rámca CyBOK

Ú	Oblasť poznania	Dodatočná klasifikácia
1	Network Security	Transport-Layer Security
2	Security Operations & Incident Management	Knowledge: Intelligence and analytics (Cybersecurity knowledge managment)
3	Web & Mobile Security	Server Side Vulnerabilities and Mitigations (Injection Vulnerabilities: Command Injections)
4	Adversial Behaviour	A Characterisation of Adversaries
5	*Neklasifikovateľné*	
6	Adversial Behaviour	A Characterisation of Adversaries

Tabuľka A.9: Klasifikácia hry Captain Slovakistan pomocou rámca CyBOK

Ú	Oblasť poznania	Dodatočná klasifikácia
1	Web & Mobile Security	Fundamental Concepts and Approaches (Webification: Hypertext Markup Language)
2	Web & Mobile Security	Server Side Vulnerabilities and Mitigations (Injection Vulnerabilities: SQL-injection)
3	Web & Mobile Security	Server Side Vulnerabilities and Mitigations (Injection Vulnerabilities: User Uploaded Files)
4	Adversial Behaviour	A Characterisation of Adversaries
5	Adversial Behaviour	A Characterisation of Adversaries

Tabuľka A.10: Klasifikácia hry COVID-19 Truth seeker pomocou rámca CCF

Ú	Oblasť poznania	Jednotka poznania	Téma (Kľúčové slovo)
1	Connection Security	Network Defense	Network access control
2	Connection Security	Network Services	Vulnerabilities and example exploits
3	System Security	System Management	Insider threat
4	Data Security	Data Integrity and Authentication	Password attack techniques (Dictionary attack)
5	Data Security	Secure Communication Protocols	Application and transport layer protocols (SSH)
	Data Security	Access Control	Logical data access control (Discretionary Access Control)
	Data Security	Information Storage Security	Data collection

Tabuľka A.11: Klasifikácia hry Uploaded Kobylka 3302 pomocou rámca CCF

Ú	Oblasť poznania	Jednotka poznania	Téma (Kľúčové slovo)
1	Connection Security	Network Defense	Network access control
2	System Security	System Control	Vulnerabilities models
3	Connection Security	Network Services	Vulnerabilities and example exploits
4	Data Security	Information Storage Security	Data collection
5	System Security	System Management	Insider threat
	Data Security	Data Integrity and Authentication	Password attack techniques (Dictionary attack)
6	Data Security	Secure Communication Protocols	Application and transport layer protocols (SSH)
	Data Security	Access Control	Logical data access control (Discretionary Access Control)
	Data Security	Information Storage Security	Data collection

Tabuľka A.12: Klasifikácia hry Captain Slovakistan pomocou rámca CCF

Ú	Oblasť poznania	Jednotka poznania	Téma (Kľúčové slovo)
1	Software Security	Implementation	Taking environment into account
2	Connection Security Data Security	Network Services Information Storage Security	Vulnerabilities and example exploits Database security
3	*Neklasifikovateľné*		
4	Organizational Security	Systems Administration	Database system administration
5	Organizational Security Data Security	Systems Administration Information Storage Security	Database system administration Data erasure

Tabuľka A.13: Klasifikácia hry COVID-19 Truth seeker pomocou rámca Cyber Kill Chain

Ú	Fáza	Akcia
1	Reconnaissance	Discover internet-facing server
2	Exploitation Command & Control	Sofware vulnerability Open two way communications channel to C2 infrastructure
3	Actions on objectives	Collect user credentials
4	*Neklasifikovateľné*	
5	Actions on objectives Actions on objectives	Lateral Movement through environment Collect and exfiltrate data

Tabuľka A.14: Klasifikácia hry Uploaded Kobylka 3302 pomocou rámca Cyber Kill Chain

Ú	Fáza	Akcia
1	Reconnaissance	Discover internet-facing server
2	*Neklasifikovateľné*	
3	Exploitation Command & Control	Sofware vulnerability Open two way communications channel to C2 infrastructure
4	Actions on objectives	Collect and exfiltrate data
5	Actions on objectives	Collect user credentials
6	Actions on objectives Actions on objectives	Lateral Movement through environment Collect and exfiltrate data

Tabuľka A.15: Klasifikácia hry Captain Slovakistan pomocou rámca Cyber Kill Chain

Ú	Fáza	Akcia
1	Reconnaissance	Discover internet-facing server
2	Exploitation	Sofware vulnerability
3	Installation	Install webshell on web server
4	Actions on objectives Actions on objectives	Collect user credentials Collect and exfiltrate data
5	Actions on objectives	Surreptitiously modify data

Tabuľka A.16: Klasifikácia hry COVID-19 Truth seeker pomocou rámca MITRE ATT&CK for Enterprise

Ú	Taktika	Technika (Podtechnika)
1	Reconnaissance Reconnaissance Resource Development	Active Scanning (Vulnerability Scanning) Gather Victim Host Information (Software) Obtain Capabilities (Vulnerabilities)
2	Initial Access Execution	Exploit Public-Facing Application Command and Scripting Interpreter (Unix Shell)
3	Credential Access Exfiltration	Unsecured Credentials (Private Keys) Exfiltration Over C2 Channel
4	Credential Access	Brute Force (Password Cracking)
5	Lateral Movement Collection	Remote Services (SSH) Data from Local System

Tabuľka A.17: Klasifikácia hry Uploaded Kobylka 3302 pomocou rámca MITRE ATT&CK for Enterprise

Ú	Taktika	Technika (Podtechnika)
1	Reconnaissance	Active Scanning
2	Reconnaissance Resource Development	Gather Victim Host Information (Software) Obtain Capabilities (Vulnerabilities)
3	Initial Access Execution	Exploit Public-Facing Application Command and Scripting Interpreter (Unix Shell)
4	Collection Exfiltration	Data from Local System Exfiltration Over C2 Channel
5	Credential Access Exfiltration Credential Access	Unsecured Credentials (Private Keys) Exfiltration Over C2 Channel Brute Force (Password Cracking)
6	Lateral Movement Collection	Remote Services (SSH) Data from Local System

Tabuľka A.18: Klasifikácia hry Captain Slovakistan pomocou rámca MITRE ATT&CK for Enterprise

Ú	Taktika	Technika (Podtechnika)
1	Reconnaissance	Search Victim-Owned Websites
2	Initial Access	Exploit Public-Facing Application
3	Persistence Collection	Server Software Component (Web Shell) Data from Local System
4	Credential Access Initial Access Collection	Unsecured Credentials (Credentials in Files) Valid Accounts (Local Accounts) Data from Local System
5	Impact	Data Manipulation (Stored Data Manipulation)

B Elektronická príloha práce

Elektronická príloha práce pozostáva z vyvinutých programov, dokumentu `README.md`, ktorý opisuje konfiguráciu prostredia pomocou priloženého súboru `Pipfile` a použitie priložených programov, licencie k programom `LICENSE`, a troch adresárov.

- Adresár `definitions` obsahuje definície troch hier použitých pri vývoji a testovaní vyvinutých programov. Ku každej hre je dostupná pôvodná verzia definície spolu s verziou rozšírenou o techniky útokov (verzie s technikami majú v názve príponu `_with_techniques`).
- Adresár `matrix` obsahuje pomocné súbory pre program `technique_visualizer_matrix.py`: šablónu jazyka Jinja2, CSS súbor pre štylizáciu súborov vygenerovaných týmto programom a vzorové výstupy programu pre obe jeho režimy použitia vo formáte HTML.
- Adresár `images` obsahuje vzorové výstupy ďalších programov na vizualizáciu porovnania hier vo formáte SVG aj PNG, s možnosťou ignorovania aj brania ohľad na podtechniky pri porovnaní hier (špecifikované v názve súborov príponou `_no_sub` ak boli ignorované, `_sub` ak neboli). K programu `technique_visualizer_supervenn.py` sú ešte dostupné ďalšie dva obrázky s príponou `_side_plots`. Pri generovaní týchto obrázkov bola zvolená možnosť zobrazenia dodatočných informácii vo výslednej vizualizácii (a ignorovania podtechník).

Bibliografia

1. (ISC)^2. *(ISC)^2 Cybersecurity Workforce Study, 2020*. 2020. Tech. spr. Dostupné tiež z: https://www.isc2.org/-/media/ISC2/Research/2020/Workforce-Study/ISC2ResearchDrivenWhitepaperFINAL.
2. VYKOPAL, Jan; OŠLEJŠEK, Radek et al. KYPO Cyber Range: Design and Use Cases. In: *Proceedings of the 12th International Conference on Software Technologies - Volume 1: ICSOFT*, SciTePress, 2017, s. 310–321. ISBN 9789897582622. Dostupné z DOI: 10.5220/0006428203100321.
3. TOVARŇÁK, Daniel et al. *KYPO Cyber Range Platform* [online]. 2021 [cit. 2021-05-16]. Dostupné z: https://docs.crp.kypo.muni.cz/.
4. ŠVÁBENSKÝ, Valdemar; VYKOPAL, Jan; CERMAK, Milan; LAŠTOVIČKA, Martin. Enhancing Cybersecurity Skills by Creating Serious Games. In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. Larnaca, Cyprus: Association for Computing Machinery, 2018, s. 194–199. ITiCSE 2018. ISBN 9781450357074. Dostupné z DOI: 10.1145/3197091.3197123.
5. ŠVÁBENSKÝ, Valdemar; VYKOPAL, Jan. Challenges Arising from Prerequisite Testing in Cybersecurity Games. In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, s. 56–61. SIGCSE '18. ISBN 9781450351034. Dostupné z DOI: 10.1145/3159450.3159454.
6. PETERSEN, Rodney; SANTOS, Danielle; SMITH, Matthew; WITTE, Gregory. *Workforce Framework for Cybersecurity (NICE Framework)*. 2020. Tech. spr. National Institute of Standards & Technology.
7. MARTIN, Andrew; RASHID, Awais; CHIVERS, Howard; DANEZIS, George; SCHNEIDER, Steve; LUPU, Emil. The Cyber Security Body of Knowledge. In: University of Bristol, 2019, kap. Intro-

duction to CyBOK. Dostupné tiež z: https://www.cybok.org/. Version 1.0.

8. BURLEY, Diana L. et al. *Cybersecurity Curricula 2017: Curriculum Guidelines for Post-Secondary Degree Programs in Cybersecurity*. New York, NY, USA: Association for Computing Machinery, 2018. ISBN 978-1-4503-8919-8.
9. LOCKHEED MARTIN. *Applying Cyber Kill Chain® Methodology to Network Defense*. 2015. Tech. spr. Dostupné tiež z: https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/Gaining_the_Advantage_Cyber_Kill_Chain.pdf.
10. STROM, Blake E.; APPLEBAUM, Andy; MILLER, Doug P.; NICKELS, Kathryn C.; PENNINGTON, Adam G.; THOMAS, Cody B. *MITRE ATT&CK®: Design and philosophy*. 2020. Tech. spr. Dostupné tiež z: https://attack.mitre.org/docs/ATTACK_Design_and_Philosophy_March_2020.pdf.
11. MITRE. *Enterprise Matrix* [online]. 2021 [cit. 2021-05-16]. Dostupné z: https://attack.mitre.org/matrices/enterprise/.
12. NIST. *NICE Framework Resource Center* [online]. 2021 [cit. 2021-05-16]. Dostupné z: https://www.nist.gov/itl/applied-cybersecurity/nice/nice-framework-resource-center/resources.
13. NICCS. *Workforce Framework for Cybersecurity* (*NICE Framework*) [online]. 2021 [cit. 2021-05-16]. Dostupné z: https://niccs.cisa.gov/workforce-development/cyber-security-workforce-framework.
14. WRENCH, Peter M.; IRWIN, Barry V. W. Towards a PHP webshell taxonomy using deobfuscation-assisted similarity analysis. In: *2015 Information Security for South Africa* (*ISSA*). 2015, s. 1–8. Dostupné z DOI: 10.1109/ISSA.2015.7335066.
15. BARDOU, Philippe; MARIETTE, Jérôme; ESCUDIÉ, Frédéric; DJEMIEL, Christophe et al. jvenn: an interactive Venn diagram viewer. *BMC bioinformatics*. 2014, roč. 15, č. 1, s. 1–7.
16. LEX, A.; GEHLENBORG, N.; STROBELT, H.; VUILLEMOT, R.; PFISTER, H. UpSet: Visualization of Intersecting Sets. *IEEE Transactions on Visualization and Computer Graphics*. 2014, roč. 20, č. 12, s. 1983–1992. Dostupné z DOI: 10.1109/TVCG.2014.2346248.

17. KELLEY, Kyle et al. *pyUpSet* [online]. 2016 [cit. 2021-05-16]. Dostupné z: https://github.com/ImSoErgodic/py-upset.

18. NOTHMAN, Joel. *UpSetPlot documentation* [online]. 2020 [cit. 2021-05-16]. Dostupné z: https://upsetplot.readthedocs.io/en/stable/.

19. HUNTER, John et al. *Matplotlib: Visualization with Python* [online]. 2021 [cit. 2021-05-22]. Dostupné z: https://matplotlib.org/.

20. INDUKAEV, Fedor. *Supervenn Python package*. Zenodo, 2021. Ver. v0.3.2. Dostupné z DOI: 10.5281/zenodo.4424381.

21. FLETCHER, Sam; ISLAM, Md Zahidul. Comparing sets of patterns with the Jaccard index. *Australasian Journal of Information Systems*. 2018, roč. 22. Dostupné z DOI: 10.3127/ajis.v22i0.1538.

22. RICHARDSON, Leonard. *Beatiful Soup* [online]. 2021 [cit. 2021-05-22]. Dostupné z: https://www.crummy.com/software/BeautifulSoup/.

23. OASIS OPEN. *TAXII 2 Python API Documentation* [online]. 2017 [cit. 2021-05-22]. Dostupné z: https://taxii2client.readthedocs.io/en/latest/.

24. OASIS OPEN. *STIX 2 Python API Documentation* [online]. 2017 [cit. 2021-05-22]. Dostupné z: https://stix2.readthedocs.io/en/latest/.

25. PALLETS. *Jinja* [online]. 2021 [cit. 2021-05-16]. Dostupné z: https://jinja.palletsprojects.com/en/3.0.x/.

26. BERG, Sebastian et al. *NumPy* [online]. 2021 [cit. 2021-05-22]. Dostupné z: https://numpy.org/.

27. DRAKE JR., Fred L. et al. *json – JSON encoder and decoder* [online]. 2021 [cit. 2021-05-22]. Dostupné z: https://docs.python.org/3/library/json.html.

28. DRAKE JR., Fred L. et al. *re – Regular expression operations* [online]. 2021 [cit. 2021-05-22]. Dostupné z: https://docs.python.org/3/library/re.html.

29. DRAKE JR., Fred L. et al. *argparse – Parser for command-line options, arguments and sub-commands* [online]. 2021 [cit. 2021-05-22]. Dostupné z: `https://docs.python.org/3/library/argparse.html`.

30. REITZ, Kenneth. *Pipenv: Python Dev Workflow for Humans* [online]. 2021 [cit. 2021-05-16]. Dostupné z: `https://pipenv.pypa.io/en/latest/`.