SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2075

Izrada i ispitivanje kibernetičkog poligona za uvježbavanje tehničkih sposobnosti

Anđelo Kandić

Zagreb, veljača 2020.

Skenirani zadatak

Sadržaj

1. Uvod

Ubrzani razvoj tehnologije doveo je do većeg broja napada i prijetnji na internetu. Kako je broj napada postajao veći tako je rasla i potražnja za stručnjacima za kibernetičku sigurnost što je na kraju imalo za rezultat da u svijetu trenutačno nedostaje otprilike 2 milijuna stručnjaka za kibernetičku sigurnost. Predviđanja su da će do 2021. godine nedostajati otprilike 3.5 milijuna stručnjaka za informacijsku sigurnost [1].

Razlog zašto napadi postaju sve učestaliji i opasniji je i nedostatak stručnjaka, a kako kibernetička sigurnost tek dobiva na važnosti upitna je i kvaliteta i sposobnost stručnjaka koji tek ulaze u to područje.

Jedan od problema pri uvježbavanju vještina u području kibernetičke sigurnosti je i platforma na kojoj bi se ti stručnjaci uvježbavali. Produkcijski sustavi nisu pogodni za uvježbavanje jer im se može naštetiti dok se provodi uvježbavanje, a uz to oni ne mogu biti potpuno kontrolirani, tako da se testiranja ne mogu ponavljati ako je to potrebno. Zbog toga se javila potreba za simuliranim okolinama u kojima bi ljudi mogli uvježbavati svoje vještine.

Ovaj rad se bavi istraživanjem simuliranih okolina za uvježbavanje tehničkih sposobnosti iz područja kibernetičke sigurnosti, posebice kibernetičkih poligona i načina kako se organiziraju vježbe na kibernetičkim poligonima.

Ovaj rad je strukturiran na sljedeći način. U drugom poglavlju je objašnjeno što je kibernetički poligon, njegova arhitektura i detaljnije je opisana svaka komponenta koja čini njegovu strukturu. U nastavku drugog poglavlja su nabrojani komercijalni kibernetički poligoni i opisane su mogućnosti nekih od njih. Treće poglavlje opisuje kako se organiziraju vježbe na kibernetičkim poligonima. U četvrtom poglavlju su predloženi alati otvorenog koda pomoću kojih se mogu implementirati pojedine komponente koje čine kibernetički poligon. U petom poglavlju su opisana dva kibernetička poligona otvorenog koda te kako se izvodi kibernetička vježba na jednom od njih. Na kraju rada je dan zaključak i pregled korištene literature.

2. Kibernetički poligoni

Riječ poligon često implicira okruženje u kojem se uvježbavaju napadačke taktike, veoma nalik poligonima kakve koristi vojska. Tako kibernetički poligon (*engl. Cyber range*) možemo definirati kao simuliranu okolinu koja omogućava uvježbavanje ljudi u području informacijske sigurnosti. Glavna ideja je da se ljudima omogući uvježbavanje u kontroliranim, sigurnim i legalnim okolinama u kojima mogu vježbati nove tehnike i tako razvijati i testirati svoje vještine te se pripremati za rješavanje problema iz područja kibernetičke sigurnosti koji ih očekuju u stvarnom životu. Kibernetički poligon treba stvoriti realne situacije u kojima se ljudi mogu naći te treba omogućiti uvježbavanje pojedinaca ili cijelih timova u vještinama kao što su penetracijska testiranja, obrana mreže, održavanje kritične infrastrukture ili odgovor na napade.

Kibernetički napad često sadrži neki oblik zlućudnog koda koji se čak i u strogo kontroliranim vježbama mora negdje izvršiti te eventualno načiniti nekakvu štetu. Taj problem nije jedinstven samo u području kibernetičke sigurnosti nego u svakom području gdje se koriste sustavi koji potencijalno mogu prouzročiti štetu, primjerice sustavi koji se koriste za nadzor industrijskih postrojenja. U nekim slučajevima kibernetički napad može prouzrokovati i fizičku štetu. a ne samo štetu na programskim sustavima. Jedan od takvih napada je prouzročio zlućudni kod Stuxnet kojem su meta bili SCADA sustavi zaduženi za upravljanje Iranskim nuklearnim postrojenjima [2].

Zbog toga bi kibernetički poligon trebao simulirati stvarne, kompleksne mreže i napraviti izolaciju od vanjske mreže izložene internetu, tako da se unutar poligona mogu izvršiti i zlućudni programi bez straha da će se korisnička računala izvan poligona zaraziti.

Produkcijska infrastruktura nije predviđena za vježbanje kibernetičke sigurnosti zato što postoji veliki rizik da se dogode neželjene posljedice koje bi se odrazile na stabilnost i dostupnost rada. Drugi razlog je taj što produkcijska infrastruktura nije strogo kontrolirana, tako da se vježbe ne mogu tako lako pratiti i ponoviti po potrebi. Stvaranje duplicirane, ali odvojene infrastrukture koja bi se koristila samo za svrhu eksperimentiranja i vježbanja je idealno, ali nažalost i jako nerealistično očekivanje sa strane kibernetičke sigurnosti. Da bi se riješili ti problemi javila se potreba za stvaranjem kontroliranih okolina za vježbanje poput kibernetičkih poligona.

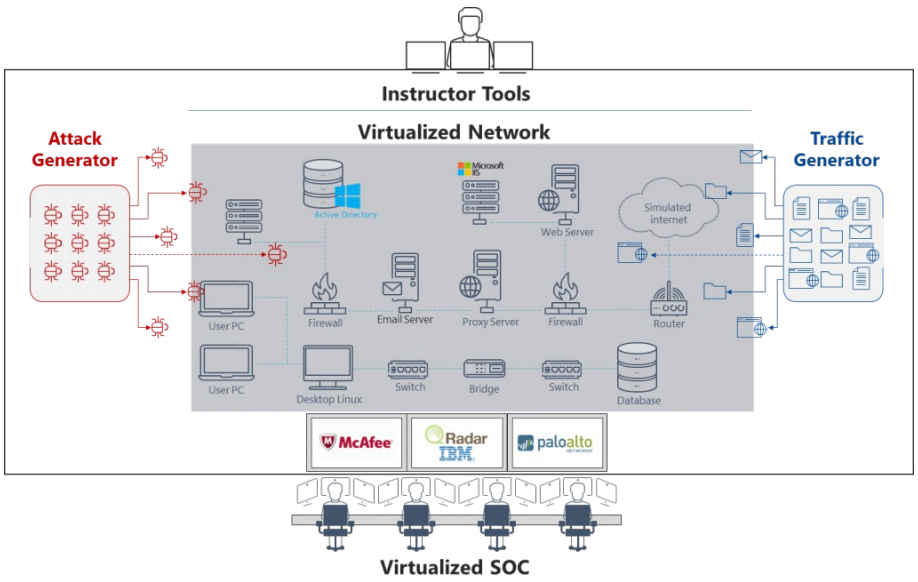
U ovom poglavlju je opisana arhitektura kibernetičkog poligona odnosno svaka pojedina komponenta koja je dio njegove arhitekture. Na kraju su nabrojani neki kibernetički poligoni i opisane su njihove funkcionalnosti.

2.1. Arhitektura kibernetičkih poligona

Kibernetički poligon sastoji se od više komponenti koje su ključne kako bi pružio sve potrebne funkcionalnosti. Tako možemo uočiti sljedeće komponente:

1. simulirana mreža,
2. program za izvršavanje scenarija,
3. generator prometa,
4. virtualni sigurnosni operativni centar,
5. alati za instruktora.

Arhitektura jednog kibernetičkog poligona koji ima sve navedene funkcionalnosti se može vidjeti na slici 2.1.



Slika 2.1. Arhitektura kibernetičkog poligona [1]

Na slici su prikazane komponente kibernetičkog poligona. U središtu je simulirana mreža koju čine razni umreženi uređaji poput korisničkih računala, vatrozida, usmjeritelja i poslužitelja. Na simuliranu mrežu je spojen generator mrežnog prometa i program za izvršavanje scenarija koji je između ostalog zadužen i za izvođenje napada. Virtualizirani sigurnosni operativni centar čine stručnjaci koji uvježbavaju obrambene vještine na kibernetičkom poligonu. Oni imaju instalirane razne alate koji im služe za praćenje mrežnog prometa i stanja sustava. Instruktor uz pomoć alata nadzire rad kibernetičkog poligona, upravlja generatorom mrežnog prometa i programom za izvršavanje scenarija.

2.1.1. Simulirana mreža

Kako bi se kibernetička vježba mogla organizirati te da bi ljudi imali na čemu vježbati potrebno je izgraditi mrežu računala koja su povezana te umrežena jedna s drugima. Ta simulirana mreža ne smije imati pristup internetu, odnosno vanjskom svijetu, ali računala unutar simulirane mreže moraju biti umrežena kako bi unutar nje svi sustavi mogli raditi i međusobno komunicirati, a ljudi koji sudjeluju u vježbi pristupiti tim sustavima.

Stvarni korisnici tijekom svog rada čitaju i odgovaraju na elektroničke poruke, razmjenjuju datoteke, otvaraju internet stranice svoje tvrtke te koriste razne druge sustave koji su im potrebni u svakodnevnom poslu. Sve navedene funkcionalnosti trebaju biti moguće i unutar simulirane mreže jednog kibernetičkog poligona.

Simulirana mreža oponaša svu kompleksnost prave mreže, uključujući mrežne čvorove, preklopnike, baze podataka, poslužitelje, uređaje "stvarnih" korisnika, a neki čak i simulaciju javnog interneta. Kako bi simulirali javni internet, istraživači s Massachusetts instituta za tehnologiju su koristili različite tehnike. Uzimali su uzorke od 10 do 1000 različitih stranica čiji su sadržaj spremali te onda te uzorke premještali na svoju infrastrukturu i uz pomoć ručno napisanih programa podešavali da rade na internet poslužiteljima koji su dio kibernetičkog poligona [3]. Kako su svi ti uzorci zapravo statički, odnosno ne možemo raditi nikakvu interakciju s njima nego ih samo pregledavati, periodički su dohvaćani novi sadržaji te su zamijenjeni za stari sadržaj kako bi se barem dijelom simulirale stvarne internet stranice.

Simuliranje dinamičkog sadržaja na internet stranicama, odnosno sadržaja kakav danas nalazimo na gotovo svim internet stranicama i koji predstavlja većinu današnjeg interneta prometa nije tako jednostavno kao simuliranje sadržaja internet stranica, a u većini slučajeva i ne znamo koji su sve sustavi i servisi pokrenuti na internet poslužiteljima. Zbog toga se u kibernetičkim poligonima većinom nalazi samo statički sadržaj, iznimka su samo neke interne internet stranice koje su od koristi ljudima koji sudjeluju u vježbi.

2.1.2. Program za izvršavanje scenarija

Program za izvršavanje scenarija je najvažniji dio u kibernetičkom poligonu. On je zadužen za izvršavanje određenih događaja koji su definirani u scenariju. Uz automatizirane događaje koje će izvršiti program postoje i događaji koji se iz nekog razloga ne mogu automatizirati nego se moraju ručno izvršiti. Oni su također dio scenarija i izvršavaju ih stručnjaci zaduženi samo za to. Redoslijed izvršavanja događaja i njihovo točno vrijeme izvršavanja nazivamo scenarij.

Scenarij može simulirati događaje koje bi izvršili zlonamjerni napadači, tehničari zaduženi za obranu od napada ili čak neki legitimni korisnici. Ako je potrebno uvježbavati tehničare koji su zaduženi za obranu od napada onda će se koristiti scenarij koji ima zapis nekog napada na internet poslužitelje, primjerice napad grubom silom (*engl. bruteforce*) ili napad uskraćivanja usluge (*engl. distributed denial of service*).

2.1.3. Generator prometa

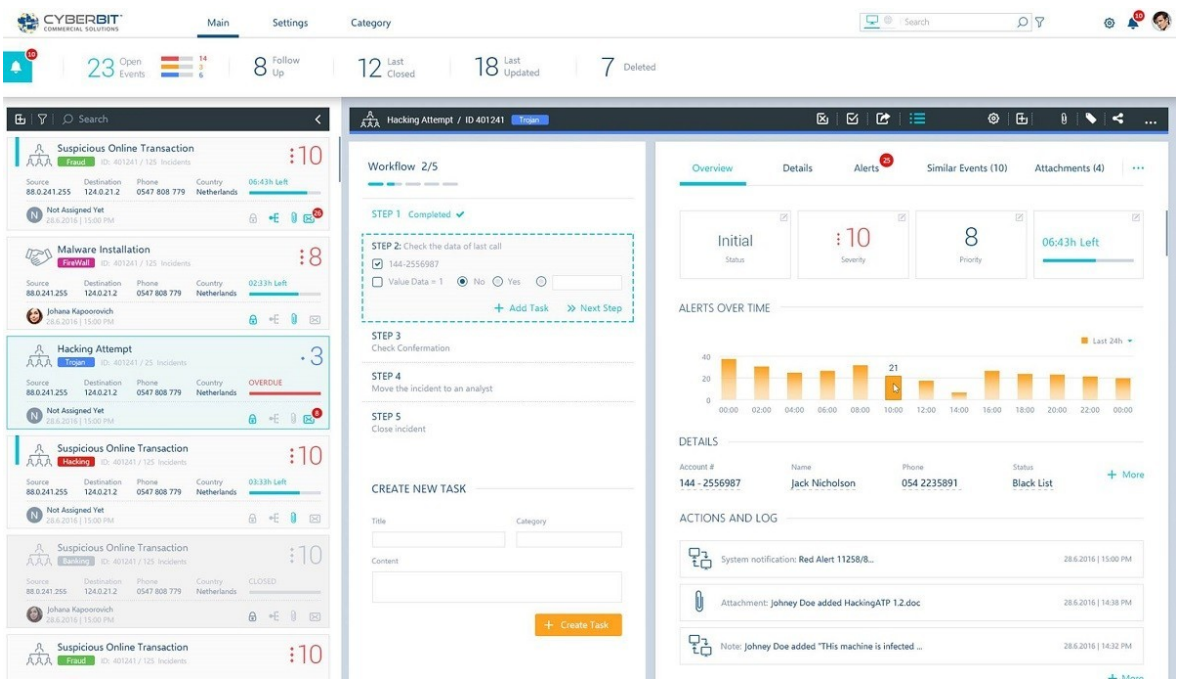
Generator prometa je komponenta koja služi da stvori promet unutar simulirane mreže kako bi ona što više nalikovala na javnu internet mrežu. Promet je potrebno stvoriti kako bi se unio šum i otežao posao timu koji je zadužen za obranu od napada. Kada u simuliranoj

mreži ne bi bilo šuma onda bi na svaku pojavu mrežnog prometa branitelji znali da je to promet koji dolazi od zlonamjernih napadača i tako bi se mogli lakše obraniti.

Kibernetički poligoni najčešće imaju mogućnost da se odabere točno određena vrsta mrežnog prometa te da se odredi između kojih čvorova će se taj promet stvoriti i u kojem vremenu. Generator prometa kojeg koriste u Ixia kibernetičkom poligonu može simulirati promet više od 300 stvarnih programa, a također može simulirati i više od 37000 različitih napada i zlonamjernih programa [4].

2.1.4. Virtualni sigurnosni operativni centar

Sigurnosni operativni centar (*engl. security operations center*) ili SOC najčešće označava fizičko mjesto ili prostor unutar nekog ureda gdje stručnjaci za kibernetičku sigurnost rade i nadziru kompletnu mrežu koju trebaju štititi. Virtualni sigurnosni operativni centar ili VSOC u kibernetičkom poligonu označava centraliziranu platformu kojoj imaju pristup članovi VSOC tima. Platforma im omogućava lakši nadzor kompletne sigurnosti svih sustava koje trebaju zaštititi. Uz pomoć platforme dobivaju uzbune ako se otkrije sigurnosni incident. Na slici 2.2. se može vidjeti VSOC kojeg je Cyberbit razvio i kojeg upotrebljavaju na svom kibernetičkom poligonu.



Slika 2.2. Cyberbit virtualni sigurnosni operativni centar [7]

Cyberbit VSOC na jednom mjestu skuplja sve bitne informacije koje jednom članu VSOC tima mogu zatrebati. VSOC s lijeve strane ispisuje sve potencijalne sigurnosne incidente, nakon što se odabere sigurnosni incident na kojeg se želi reagirati na desnoj se strani

pojavljuju koraci koje član VSOC tima treba poduzeti kako bi se riješio taj incident. Na ostatku VSOC slike se nalaze ostali bitni podaci o odabranom sigurnosnom incidentu.

Cilj VSOC tima je otkriti, analizirati i odgovoriti na kibernetičke incidente koristeći kombinaciju tehnoloških rješenja i strogo zadanih pravila.

Da bi VSOC bio uspješan nije dovoljno samo koristiti najnovija tehnološka rješenja, bitnije je da ima kvalitetno posložen tim stručnjaka koji pokrivaju različita područja. Tako jedan SOC tim mora minimalno biti podijeljen prema sljedećim ulogama [5]:

1. soc upravitelj,
2. sigurnosni analitičar,
3. forenzički istražitelj,
4. revizor usklađenosti.

SOC upravitelj je jedna osoba koja vodi timske operacije. On je zadužen da rasporedi zadatke i osigura dobru komunikaciju unutar tima.

Sigurnosni analitičar je zadužen da aktivno nadzire sustave u potrazi za sumnjivim aktivnostima i prijetnjama. On je također zadužen za procjenu rizika i procjenu ranjivosti.

Forenzički istražitelj provodi istragu o sigurnosnom incidentu nakon što sigurnosni analitičar otkrije neku sumnjivu aktivnost ili zlonamjernu aktivnost.

Iako uloga revizora usklađenosti nije tehnička uloga ipak je bitna u kibernetičkoj sigurnosti. Dolaskom na snagu opće uredbe o zaštiti podataka (*engl. General data protection regulation*) ili skraćeno GDPR zahtjeva se obavještavanje o sigurnosnom incidentu unutar 72 sata od saznanja za njega [6]. Danas svaka tvrtka koja posluje u Europi mora biti usklađena barem s GDPR-om, a ovisno o zahtjevima i području poslovanja i s drugim regulativama. Zadaća revizora usklađenosti je da se brine da radnje koje SOC provodi budu usklađene s potrebnim regulativama.

2.1.5. Alati za instruktora

Alati za instruktora obuhvaćaju sve alate koji pomažu instruktorima u organiziranju i provođenju kibernetičke vježbe. Instruktori obično imaju pristup na kontrolnu ploču koja pruža dvije glavne funkcionalnosti, organiziranje i nadzor kibernetičke vježbe.

Za organizaciju kibernetičke vježbe instruktor mora:

1. dodijeliti vježbenike određenim timovima kojima će pripadati dok traje kibernetička vježba,
2. stvoriti mrežnu topologiju na kojoj će se održati kibernetička vježba,
3. stvoriti scenarij koji će biti pokrenut na kibernetičkom poligonu.

Nakon što se kibernetička vježba organizira i započne instruktor nadzire odvijanje vježbe te na kontrolnoj ploči može imati sljedeće funkcionalnosti:

1. praćenje tijeka izvođenja scenarija u kibernetičkoj vježbi,
2. dodjeljivanje bodova i komuniciranje sa sudionicima u kibernetičkoj vježbi,
3. upravljanje s generatorom lažnog prometa.

2.2. Implementacije kibernetičkih poligona

Kibernetički poligoni zbog svoje kompleksnosti i cijene nisu namijenjeni da bi ih pojedinci privatno koristili. Trenutno postoji jako malo besplatnih rješenja koja ne pružaju ni blizu toliko funkcionalnosti kao što su komercijalna rješenja. Zbog toga kibernetičke poligone najčešće koriste:

- vojske,
- vlade i vladine agencije,
- fakulteti,
- korporacije,
- pružatelji mrežne opreme.

Podjelu na kibernetičke poligone možemo napraviti gledajući ih prema tome tko ih je napravio. Prema tome se mogu podijeliti na:

- komercijalne
  - Cyberbit [10],
  - Paloaltonetworks [11],
  - Ixia [12],
  - Cisco Cyber Range [21],
  - IBM X-Force [25],
  - Airbus [22],
- akademske
  - AIT Cyber Range [9],
  - Kypo [13],
  - Virginia Cyber Range [23],
  - Michigan Cyber Range [24],
- vojne ili državne
  - CRATE [14],

- National Cyber Range [15],
- CR14 [26],
- CSDCP [27],
- National Cyber Testbed [30].

Kibernetički poligoni se također mogu podijeliti i prema tome da li se nalaze u fizičkom okruženju ili su smješteni u oblaku. Poligoni smješteni u oblaku su često jeftiniji i lakši za održavanje zbog toga što se infrastruktura ne nalazi smještena fizički kod vlasnika nego je kod treće strane, te se mogu lakše nadograđivati, te paliti i gasiti po potrebi bez nepotrebnih troškova.

Zbog kompleksnosti i cijene izrade kibernetičkih poligona podaci o njima su često javno nedostupni. U sljedećim poglavljima će biti opisani kibernetički poligoni za koje se skupilo dovoljno informacija da ih se može opisati.

2.2.1. Cyberbit

Cyberbit je stvorio trenutno najkorišteniju platformu za izradu kibernetičkih poligona na svijetu. Njihova platforma je namijenjena svima zainteresiranima koji žele stvoriti svoj kibernetički poligon. Tako velike korporacije mogu uvježbavati svoje SOC timove u obrani od napada, akademske ustanove mogu uvježbavati svoje studente na karijeru u kibernetičkoj sigurnosti, a tvrtke koje pružaju informatičke usluge mogu stvoriti svoj kibernetički poligon te na njemu uvježbavati stručnjake.

Uz pomoć Cyberbit platforme su svoje kibernetičke poligone stvorile mnoge vladine agencije, velike tvrtke i akademske ustanove. Najznačajnije od njih su [16]:

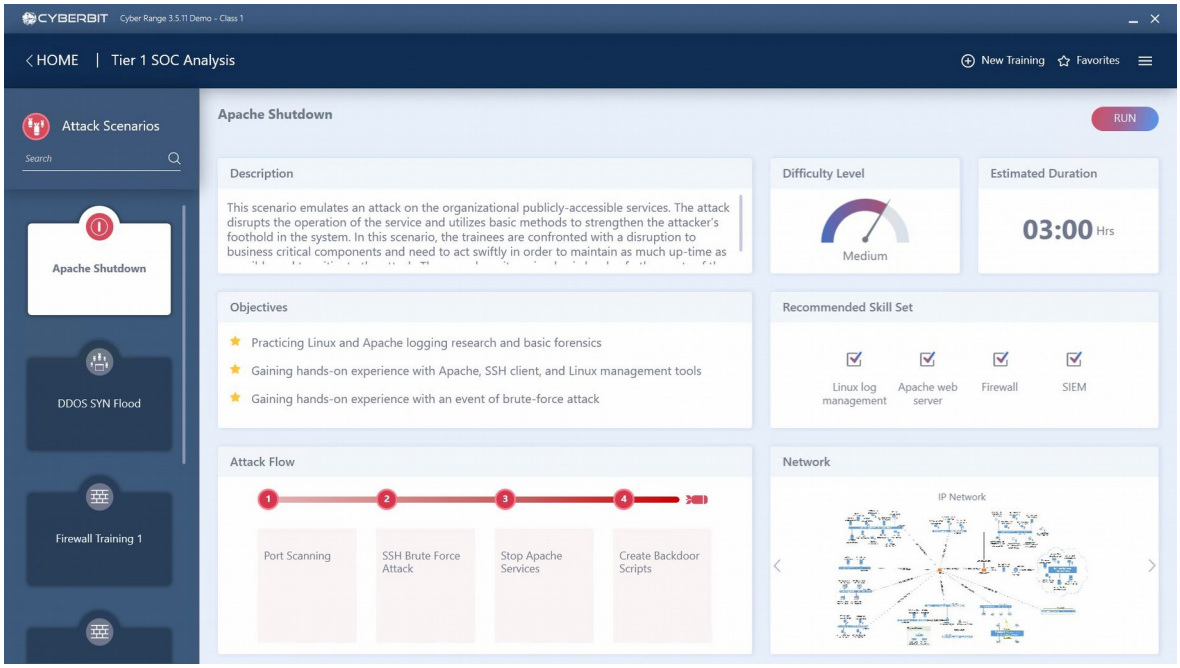
- Izraelske obrambene snage,
- A1 telekom, Austrija,
- Australski odjel za obranu,
- Sveučilište Regent,
- Sveučilište Miami Dade.

Cyberbit platforma ima sljedeće funkcionalnosti [1]:

- simulirana mreža,
- virtualni sigurnosni operativni centar,
- program za izvršavanje scenarija,
- generator prometa,

- okruženje za nadzor izvođenja kibernetičke vježbe – moguće je snimati kibernetičku vježbu, ponovno ju pogledati, ocjenjivati sudionike i davati im upute dok traje vježba,
- podršku za različite timove – mogu se vježbati napadači i branitelji,
- podršku za stvaranje svojih scenarija i mrežne topologije,
- podršku za industrijske kontrolne sustave (*engl. Industrial control systems*) i SCADA sustave.

Na slici 2.3. se može vidjeti kontrolna ploča uz pomoć koje se organizira kibernetička vježba.



Slika 2.3. Cyberbit kontrolna ploča za organiziranje kibernetičke vježbe [17]

Cyberbit kontrolna ploča omogućava jednostavno organiziranje kibernetičke vježbe i upravljanje s kibernetičkim poligonom. Organizator može stvoriti novu vježbu, vizualno mu je prikazana mrežna topologija koju može mijenjati, može podesiti vrijeme trajanja vježbe te pokrenuti razne napade koji su predefinirani.

2.2.2. Palo Alto Networks

Palo Alto Networks je multinacionalna tvrtka koja se bavi izradom raznih rješenja za poboljšavanje kibernetičke sigurnosti. Glavni proizvod im je platforma koja uključuje napredni vatrozid za okruženja u oblaku (*engl. cloud environments*).

Tvrtka je 2018. godine pokrenula Globalnu inicijativu za kibernetičke poligone (*engl. Global Cyber Range Initiative*) [18]. Cilj inicijative je da izgradi kibernetičke poligone za svoje korisnike u Europi, Srednjem Istoku, Africi, Amerikama i Azijsko-Pacifičkoj regiji.

Na tim kibernetičkim poligonima njihovi korisnici se mogu uvježbavati i tako unaprjeđivati svoje vještine.

Uz pomoć Paloaltonetworks platforme su održane neke od najvećih kibernetičkih vježbi, poput [19]:

- *Combined endeavor's Cyber Endeavor,*
- *Pacific endeavorrs Cyber Endeavor,*
- *Singapore Critical national infrastructure exercises,*

Promatrajući Paloaltonetworks kibernetički poligon prema njegovim funkcionalnostima može se zaključiti da ima sve funkcionalnosti koje jedan kompletan kibernetički poligon treba posjedovati. Tako Paloaltonetworks ima grafičko sučelje za izgradnju mrežne topologije i scenarija koji će se odviti u kibernetičkoj vježbi, podržava uvježbavanje napadača i branitelja, a instruktori na svojem sučelju mogu nadzirati izvođenje kibernetičke vježbe i komunicirati sa sudionicima. Za generator prometa koriste fizički uređaj koji kontinuirano dohvaća zlonamjerni promet s cijelog interneta i ponovno ga rekonstruira za potrebe vježbe [20].

2.2.3. Kypo

Kypo je najveći akademski kibernetički poligon u Češkoj Republici. Od 2010. godine ga sponzorira Ministarstvo unutarnjih poslova Češke Republike. Platforma je u potpunosti napravljena u oblaku i može se koristiti za istraživanje, edukaciju i održavanje kibernetičkih vježbi. Na njoj se održavaju državne kibernetičke vježbe poput *Cyber Czech* [13].

Uspoređujući Kypo s Cyberbit ili Paloaltonetworks platformama može se uočiti nedostatak određenih funkcionalnosti od kojih su najbitniji:

- nedostatak naprednih funkcionalnosti u kontrolnoj ploči za organiziranje kibernetičkog poligona,
- nedostatak automatskog izvršavanja napada,
- nedostatak generatora prometa.

Iako Kypo platforma posjeduje kontrolnu ploču za organiziranje kibernetičke vježbe ona je jako limitirana s funkcionalnostima. Ne postoji mogućnost dinamičkog mijenjanja mrežne topologije i infrastrukture ili timova koji sudjeluju u kibernetičkoj vježbi. Ti podaci se moraju unaprijed pripremiti u obliku datoteke s ekstenzijom JSON (*engl. JavaScript Object Notation*) koja se nakon toga učitava u kontrolnoj ploči [28]. Nakon što se datoteka učita nema više mogućnosti mijenjanja podataka te se može samo pokrenuti vježba.

Kypo platforma također nema mogućnost automatskog izvršavanja napada. U svakoj vježbi u kojoj je cilj uvježbavanje obrane od napadača mora postojati i tim napadača koji

će sve napade zadane scenarijem izvršavati ručno što povećava broj ljudi koji moraju sudjelovati u vježbi.

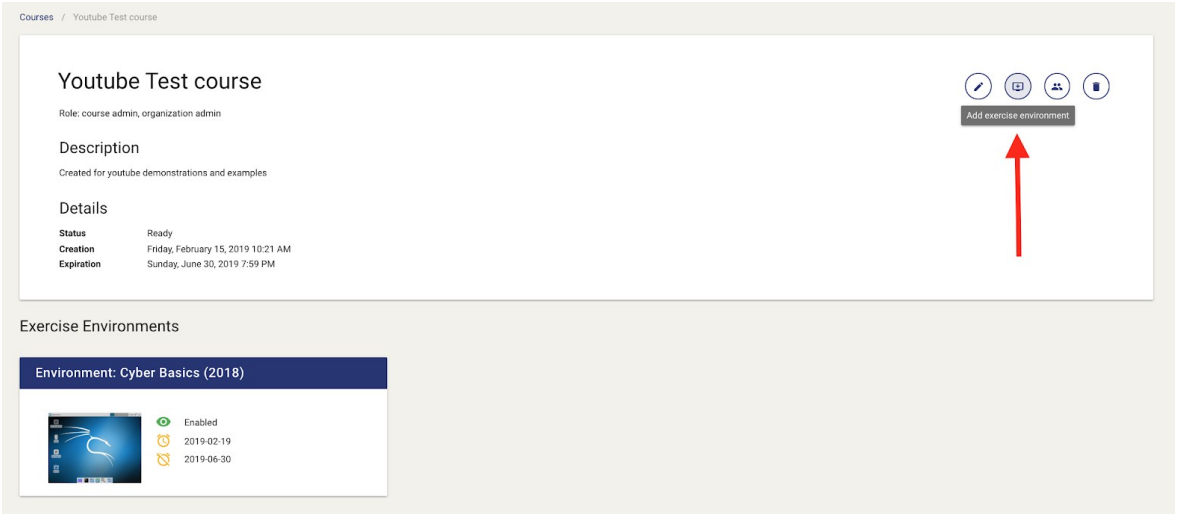
Treći bitni nedostatak se nalazi u tome što Kypo platforma nema nikakav generator prometa. To bitno olakšava posao obrambenom timu zato što čim se pojavi mrežni promet znaju da su to napadači.

2.2.4. Virginia Cyber Range

Virginia kibernetički poligon je nastao 2016. godine s ciljem da olakša pristup resursima za učenje računalne sigurnosti u srednjim školama i fakultetima. Razlog stvaranja vlastitog kibernetičkog poligona je nedostatak stručnjaka za računalnu sigurnost diljem države. Od tada je kibernetički poligon koristilo preko 5000 učenika, 200 srednjih škola i fakulteta te se stvorilo preko 50000 virtualnih okruženja [23].

Korištenje kibernetičkog poligona i svih materijala za učenje je potpuno besplatno za sve javno financirane srednje škole i fakultete na području Virginie. Većina nastavnika koristi poligon za podučavanje predmeta iz računalne sigurnosti, ali također se može koristiti i za organiziranje natjecanja u hvatanju zastava (*engl. Capture the flag ili CTF*).

Postoji kontrolna ploča gdje se mogu registrirati nastavnici i učenici. Nastavnik može na jednostavan način dodati novi predmet koji predaje te pozvati učenike kao sudionike u tom predmetu. Nastavnik nakon toga može dodati razna virtualna okruženja koja služe učenicima za vježbanje. Svaki student može pokrenuti svoje okruženje koje je odvojeno od okruženja drugih učenika. Na slici 2.4. se može vidjeti dio kontrolne ploče preko koje nastavnik upravlja pojedinim predmetom.



Slika 2.4. Kontrolna ploča za profesora na kibernetičkom poligonu Virginia [29]

Slika prikazuje kontrolnu ploču preko koje nastavnik upravlja Virgina kibernetičkim poligonom. Nastavnik može dodati novo okruženje za vježbanje, ali nema mogućnost

mijenjanja mrežne topologije ili pokretanja predefiniranih napada. Svaki učenik pokreće svoju instancu kibernetičkog poligona preko koje može vježbati.

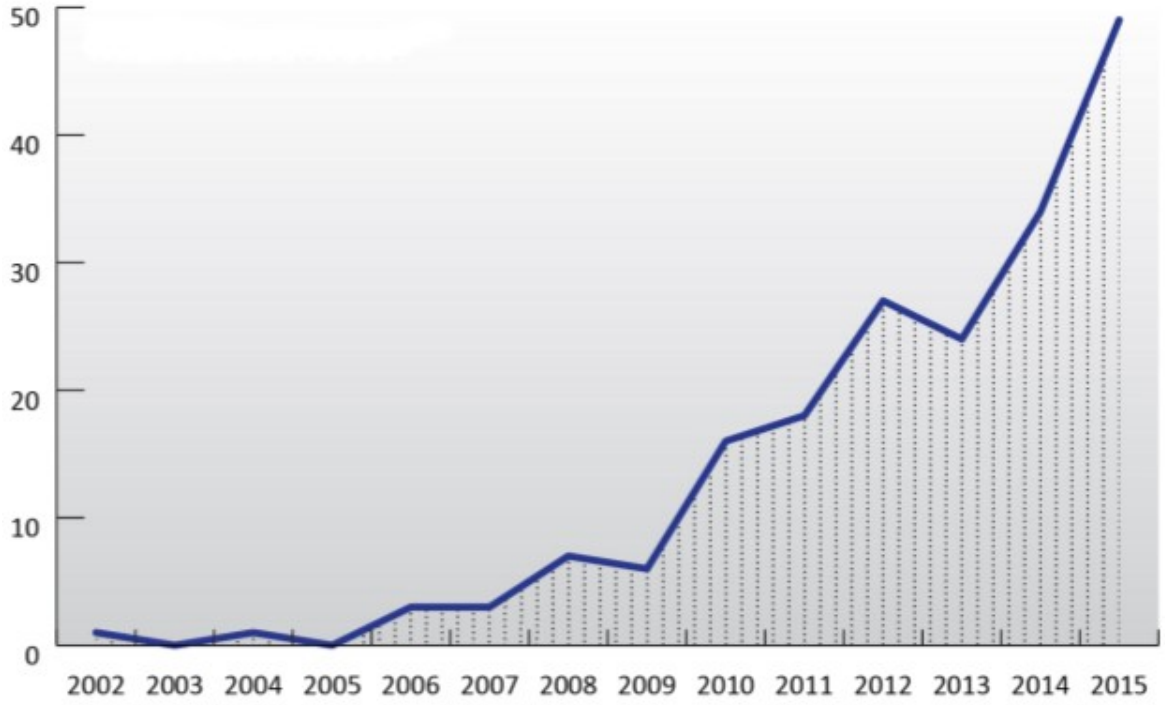
Kibernetički poligon Virginia jednako kao i Kypo zaostaje za komercijalnim rješenjima. Vidljivo je da nema nikakve mogućnosti izgradnje scenarija po želji, ne mogu se automatski izvršavati napadi te ne postoji neki generator prometa.

3. Vježbe na kibernetičkim poligonima

Vježbe su važan alat u procjeni spremnosti računalnih stručnjaka u borbi protiv kibernetičkih kriza, tehnoloških neuspjeha ili incidenata koji se tiču kritične informacijske infrastrukture. One omogućuju organizacijama da se usmjere na specifične slabosti, poboljšaju suradnju između timova stručnjaka koji rade na različitim područjima, identificiraju međuovisnosti sustava i potaknu poboljšanja u kontinuiranom planiranju.

Agencija Europske Unije za kibernetičku sigurnost (*engl. European Union Agency for Cybersecurity ili ENISA*) je 2015. godine provela istraživanje u koje su uključili podatke od preko 200 održanih vježbi [31]. U istraživanju je zaključeno da se od 2012. godine kontinuirano i ubrzano povećava broj vježbi koje se održavaju. Također je uočen i porast broja vježbi na kojima zajedno sudjeluju javni i privatni sektor. Iz tih podataka se može zaključiti da je svima u interesu da se na tim vježbama okupi što veći broj sudionika kako bi svi dobili veću korist od vježbe.

Istraživanje koje je provela ENISA je otkrilo da se vježbe više fokusiraju na istraživanje novih procesa rada i međusobnu suradnju, a manje na učvršćivanje i poboljšavanje postojećih. Na slici 3.1. se mogu vidjeti podaci o broju održanih vježbi po godinama od 2002. do 2015. godine.

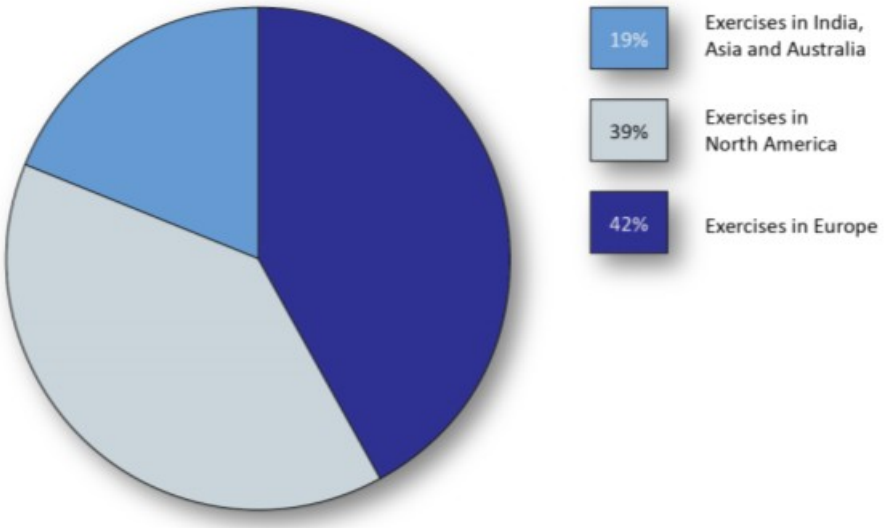


Slika 3.1. Broj održanih vježbi od 2002. do 2015. godine [31]

Graf prikazuje broj održanih vježbi u posljednja dva desetljeća. Početkom prvog desetljeća su se vježbe jako rijetko održavale, ali je postojao blagi porast u broju održavanja. Već na

početku drugog desetljeća dolazi do eksponencijalnog rasta u broju održanih vježbi. Predviđanja su takva da će se rast još više ubrzati u idućim godinama.

Na slici 3.2. možemo vidjeti graf raspodjele organizacije vježbi prema kontinentima.



Slika 3.2. Raspodjela kibernetičkih vježbi prema kontinentu gdje su se organizirale [31]

Gledajući statistiku prema kontinentu gdje se organiziraju vježbe može se zaključiti da Europa predvodi u vježbama. Od 2012. do 2015. godine se na području Europe organiziralo 42% svih svjetskih vježbi, 39% vježbi se organiziralo u Sjevernoj Americi, a 19% je podijeljen između Indije, Azije i Australije.

Neke od najpoznatijih vježbi koje se organiziraju na međunarodnoj razini su:

- *Locked Shields* [56],
- *Crossed Swords* [53],
- *Cyber Coalition* [54],
- *Cyber Europe* [55].

Locked Shields je godišnja vježba koju organizira NATO-ov Udruženi centar za kibernetičku obranu (engl. NATO Cooperative Cyber Defence Center of Excellence ili NATO CCDCOE) u Tallinn-u u Estoniji, a vježba se održava uz pomoć CR14 kibernetičkog poligona kojim upravljaju Estonske Defenzivne Snage. Smatra se da su *Locked Shields* najveća, najkompleksnija i tehnološki najnaprednija vježba na svijetu za uvježbavanje kibernetičke obrane. Preko 1200 stručnjaka iz različitih 30 država je sudjelovalo na vježbi 2019. godine [56]. Timovi koji su zaduženi za napade su izvršili preko 2500 napada na 4000 sustava koje timovi zaduženi za obranu moraju obraniti. U vježbu se svake godine dodaju novi izazovi i tehnologije za koje se misli da su dio kritične infrastrukture svake države, tako je 2019. dodan sustav za raspodjelu energije koji je imao i komponentu zaduženu za stvaranje energije koja se u vježbi morala obraniti.

Crossed Swords je također NATO-va godišnja vježba. Organizira se s ciljem uvježbavanja napadača u penetracijskim testiranjima i digitalnoj forenzici. Iako je *Crossed Swords* ispočetka zamišljena kao radionica, vježba je vrlo brzo proširena u kompleksnosti. Vježba također pruža priliku za uvježbavanje onih stručnjaka koji u vježbi *Locked Shields* sudjeluju u timu napadača.

Cyber Coalition je godišnja vježba koju organizira NATO. Vježba traje 5 dana, a okuplja oko 1000 stručnjaka iz zemalja članica NATO-a i država koje su saveznici NATO-a. U vježbi sudjeluju eksperti za kibernetičku sigurnost, pravni eksperti, državni i vojni službenici te predstavnici iz akademskog i industrijskog sektora.

Cyber Europe je vježba koju agencija ENISA organizira svake 2 godine za zemlje članice Europske Unije. *Cyber Europe* je civilno organizirana vježba za razliku od *Locked Shields* i *Cyber Coalition* koje su vojno bazirane vježbe. U zadnjoj vježbi održanoj 2018. godine je sudjelovalo 900 stručnjaka iz 28 članica Europske Unije. Ciljana publika vježbe su bile sve organizacije i profesionalci koji su uključeni u zračni promet.

U ovom poglavlju su opisane vrste kibernetičkih vježbi, objašnjeno je kako se kibernetičke vježbe organiziraju te koji su sudionici u tim vježbama. Nakon toga je objašnjeno što je scenarij u kibernetičkoj vježbi. Na kraju je opisano kako se boduju timovi te kako se nadzire odvijanje kibernetičke vježbe.

3.1. Vrste vježbi

Kibernetičke vježbe se mogu provoditi na različite načine. Iz podataka dobivenih u istraživanju koje je provela ENISA se može zaključiti da postoje i različiti tipovi vježbi. Sve te razlike mogu se svesti na parametre koji su opisani u međunarodnom standardu ISO 22398. Prema ISO 22398 standardu vježbe možemo podijeliti prema ciljevima na koje su usredotočene [32]:

- poboljšanje i testiranje suradnje u odgovaranju na nacionalne ili međunarodne kibernetičke incidente,
- procjena vještina na području kibernetičke sigurnosti te spremnosti pojedinca i organizacija na reakciju prilikom sigurnosnih incidenata,
- procjena spremnosti, sposobnosti, izdržljivosti i tehničkih kapaciteta,
- uvježbavanje sudionika u scenarijima iz stvarnog svijeta pomoću kojih se može proširiti znanje, steći iskustvo te razviti tehničke vještine prije nego se dogode incidenti u stvarnom svijetu.

Iako postoje različiti tipovi vježbi poput hvatanja zastavice, simulacija, radionica, seminara ili vježbi obrane, one se sve mogu podijeliti u tri glavne kategorije [33]:

- *table top,*
- *hybrid,*

- *full live.*

U *table top* vježbama svi scenariji, podscenariji i planirani napadi su unaprijed napisani i pripremljeni. U većini slučajeva organizatori i sudionici skupa sjednu za stol i izvrše vježbu, zbog toga se vježba zove *table top*. Takve vježbe imaju limitiran broj sudionika i jako dobro definirane ciljeve. Namijenjene su organizacijama koje su nove i još nisu sudjelovale u vježbama ili organizacijama koje moraju vježbati svoje procese između održavanja dvije vježbe. Proces prijavljivanja na vježbu je jednostavan zato što se takva vježba može brzo isplanirati i organizirati u jako kratkom vremenu. Vrijeme planiranja je obično jedan do dva mjeseca, a vrijeme trajanja jedan do tri dana.

Kod *hybrid* vježbi su scenariji, podscenariji i napadi također unaprijed isplanirani, ali za razliku od *table top* vježbe napadi se izvršavaju za vrijeme izvođenja vježbe i nisu poznati unaprijed. To se radi kako bi se vježba odvila što realnije. Ova vježba zahtjeva dužu pripremu koja traje obično tri do šest mjeseci dok je vrijeme trajanja vježbe tri do pet dana. Za organizaciju *hybrid* vježbe je potrebno više ljudi i vremena te realnije mete za scenarij. Organizacije kojima je ovakva vrsta vježbe namijenjena moraju jasno znati ciljeve koje žele postići u vježbi.

Full live vježbe također imaju pripremljene scenarije i podscenarije, ali za razliku od ostalih vrsta vježbi scenarij se može prilagođavati trenutnom stanju vježbe. Kod ove vrste vježbe također postoje i događaji koji se zovu ubacivanja (*engl. injects*). To su događaji koji se ne dogode stvarno nego sudionici moraju zamisliti da su se dogodili te moraju adekvatno reagirati na njih. Timovi zaduženi za napade također razvijaju nove strategije napada u ovisnosti o stanju i napretku obrambenog tima. Ova vrsta vježbe zahtjeva puno duže i detaljnije planiranje koje traje od šest do dvanaest mjeseci, kibernetički poligon se većinom priprema od dva do tri mjeseca, a vrijeme trajanja vježbe iznosi od sedam do četrnaest dana. U takvim vježbama sudjeluje veliki broj organizacija i sudionika koji su veoma iskusni u provođenju vježbi.

3.2. Životni ciklus vježbe

Životni ciklus vježbe na kibernetičkom poligonu se može podijeliti na četiri glavne faze [34]:

- identificiranje,
- planiranje,
- provođenje,
- procjena.

U fazi identificiranja vježbe potrebno je poduzeti sljedeće korake:

- definirati jasne ciljeve,

- identificirati ključne sudionike,
- sagledati moguće scenarije koji će se provesti,
- odrediti tip i veličinu vježbe.

Vježbe se mogu organizirati tako da se tim sudionika pojedinačno uvježbava u izoliranom okruženju ili kao dio veće vježbe kojoj svi sudionici imaju pristup. Proces planiranja vježbe je u oba pristupa sličan. U fazi planiranja se prvo mora odrediti koja je svrha vježbe i ciljevi koji se planiraju postići izvođenjem vježbe. Ciljevi se prezentiraju sudionicima vježbe tako da se na kraju može procijeniti da li su zadani ciljevi ispunjeni ili ne, odnosno da li se organizacije mogu uspješno braniti od napada u stvarnom svijetu.

Nakon što se provede faza identificiranja slijedi faza planiranja koja traje najduže i najzahtjevnija je od svih faza. U fazi planiranja mora se učiniti sljedeće:

- informirati i pripremiti sudionike na vježbu,
- odrediti politiku dijeljenja informacija s medijima,
- pozvati promatrače i članove medija,
- raspodijeliti potrebna financijska sredstva,
- dogovoriti se o vremenu i mjestu izvođenja vježbe,
- raspodijeliti uloge sudionicima i stvoriti što realniji scenarij vježbe,
- pripremiti materijale koji će se koristiti u vježbi.

Da bi se vježba uspješno organizirala potrebno je održati nekoliko sastanaka gdje se vježba postupno planira, ti sastanci su nazvani:

- Razrada koncepta (*engl. Concept development meeting*),
- Inicijalni sastanak (*engl. Initial planning meeting*),
- Planiranje glavnog scenarija (*engl. Master scenario event list planning meeting*),
- Središnji sastanak planiranja (*engl. Mid-term planning meeting*),
- Završni sastanak planiranja (*engl. Final planning meeting*).

U planiranje vježbe je uključeno više ljudi koji su podijeljeni prema ulogama koje obavljaju unutar organizacije. Prilikom razrade koncepta bi trebali biti prisutni samo glavni odgovorni ljudi iz organizacije koja vodi vježbu, oni moraju jasno objasniti ciljeve vježbe te to prenijeti voditeljima planiranja. Razrada koncepta je interni sastanak u kojem se raspravlja o idejama, određuju ciljevi i organizacije koje će biti uključene u Inicijalni sastanak. Također se odlučuje kojeg će tipa biti vježba, raspravlja se o mogućem scenariju, mjestu održavanja i resursima potrebnim za održavanje vježbe. Ovaj sastanak je zamišljen kao otvorena rasprava u kojoj se osvrće na lekcije naučene iz prijašnjih vježbi s ciljem da se što bolje organizira trenutna vježba i da se ne ponove iste greške.

Na inicijalnom sastanku trebaju biti prisutni svi odgovorni za planiranje iz organizacije koja vodi vježbu kao i svi vanjski sudionici koji su uključeni u planiranje vježbe. Jako je bitno da se što ranije u fazi planiranja uključe i vanjski sudionici kako bi se osiguralo da svi shvate što se planira i koje su njihove uloge u tom procesu.

Inicijalni sastanak započinje osvrtom na odluke koje su se donijele prilikom razrade koncepta. Tako svi vanjski sudionici dobivaju informacije potrebne za daljnju raspravu. Svi sudionici ovog sastanka moraju se složiti oko scenarija vježbe i svih ciljeva koji se žele postići vježbom. Sastanak se završava s popisivanjem zadataka svim prisutnim sudionicima tako da svatko zna za koji dio planiranja je odgovoran.

Sastanak za planiranje glavnog scenarija je potrebno održati kako bi se jasno definirao scenarij i sva ubacivanja tijekom vježbe. Fokus ovog sastanka je da se prođu svi tehnički i netehnički detalji scenarija.

Na središnjem sastanku bi se trebao napraviti konačni nacrt scenarija kako bi se mogla identificirati sva dodatna logistika i zahtjevi potrebni za organizaciju vježbe. Također je potrebno pregledati i potvrditi svu dokumentaciju koja će se koristiti u vježbi.

Završni sastanak se održava mjesec dana prije izvođenja vježbe i služi kako bi se provjerilo da organizacija vježbe ide po planu. Na ovom sastanku se također donosi završna odluka hoće li se vježba održati ili ne.

Nakon faze planiranja slijedi provođenje vježbe gdje je potrebno:

- definirati pravila koja će sudionici morati poštovati,
- izvršiti scenarij i ubacivanja prema zadanom redoslijedu,
- razriješiti moguće probleme i pogreške koje se mogu dogoditi tijekom izvođenja vježbe,
- promatrati sudionike i zapisati bilješke o odlukama i aktivnostima sudionika,
- provesti anketu o vježbi nad sudionicima.

Faza provođenja vježbe je samo praćenje dogovorenog plana i scenarija te nadzor sudionika vježbe i njihovih odgovora na zadane situacije.

Kada se vježba provede do kraja slijedi faza procjene u kojoj se:

- okuplja tim ljudi koji će procijeniti rezultate vježbe,
- prikuplja i procjenjuje anketa koja se provela nad sudionicima,
- pripremaju dokumenti koji će se poslati medijima,
- priprema izvještaj koji se šalje ocjenjivačima.

Životni ciklus vježbe završava s fazom procjene. U ovoj fazi se analiziraju događanja tijekom vježbe uz pomoć zapažanja koja su vodili organizatori. Na kraju vježbe se također provodi *Hot wash*. *Hot wash* je vrsta ankete koja se treba provesti nad svim sudionicima odmah nakon završetka dok su sva događanja još uvijek svježa. Tako organizatori

trenutačno dobivaju povratne informacije o tome što je bilo dobro, što se može unaprijediti te što bi se moglo dodati u idućoj vježbi. Također se sudionicima može podijeliti i upitnik s jednostavnim da ili ne odgovorima.

Nakon što se prikupe sve potrebne informacije sastavlja se Izvještaj nakon akcije (*engl. After action report*). To je izvještaj o provedenoj vježbi u kojoj su objašnjene radnje svih sudionika vježbe. Izvještaj nakon akcije sadrži detaljno objašnjen scenarij, svrhu vježbe, popis sudionika, način bodovanja, popis infrastrukture, izvršene napade, glavne pogreške, zapažanja svih sudionika te preporuku za iduće vježbe.

3.3. Timovi

Sudionici su u vježbama obrane podijeljeni u pet timova prema njihovim vještinama, ulogama i zadaćama koje obavljaju u vježbi. Timovi su imenovani prema bojama i nazivaju se:

- plavi tim,
- crveni tim,
- bijeli tim,
- zeleni tim,
- žuti tim.

Plavi tim je zadužen za sigurnost informacijskih sustava i obranu od napada koje provode napadači. Napadači su izmišljeni i predstavlja ih crveni tim. U vježbama može biti više plavih timova, a ako su vježbe međunarodne onda svaki plavi tim predstavlja jednu državu koja sudjeluje u vježbi.

Zbog zakona i regulativa oko kibernetičke sigurnosti nije dovoljno samo tehnički obraniti svoj kibernetički prostor. Zato su se vježbe počele organizirati tako da plavi tim uz tehničku obranu mora paziti i na sve postojeće zakone, pravne regulative i medijsku politiku. U kibernetičkoj vježbi *Locked Shields* član svakog plavog tima je bio i jedan pravni savjetnik [35]. Njegov zadatak je bio da obavještava ostale članove plavog tima o zakonima kojih se moraju držati i odgovara na pravna pitanja koja je dobivao kao dio scenarija. Također postoje jasna pravila koja plavi tim mora poštivati, ne smije napadati infrastrukturu na kojoj se vježba odvija, kao ni druge plave i crvene timove.

Crveni tim u kibernetičkoj vježbi obrane ima ulogu napadača. Članovi crvenog tima ne izvršavaju napade na sustave koje brani plavi tim nasumično nego po točno zadanom scenariju. To znači da oni iskorištavaju namjerno postavljenje sigurnosne ranjivosti i po pravilima ne bi smjeli koristiti nikakve druge napade osim onih zadanih scenarijem. Uspješno izvršeni napadi imaju negativan utjecan na konačan rezultat plavog tima. Crveni i bijeli tim tijekom vježbe rade skupa, crveni tim treba slušati upute od bijelog tima. Crveni

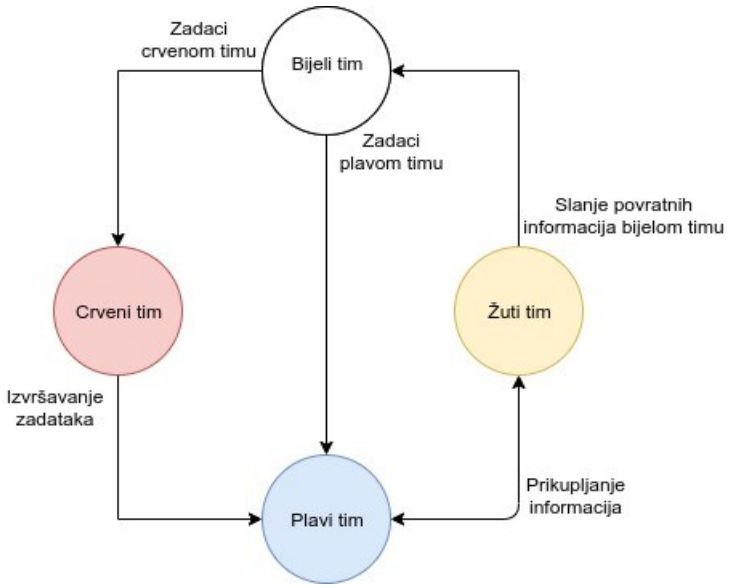
tim također ne smije napadati infrastrukturu koja je dio vježbe, a koja nije pod kontrolom plavog tima, primjerice poslužitelje koje zeleni tim koristi za nadzor vježbe.

Bijeli tim je zadužen za organizaciju i nadzor vježbe. Članovi bijelog tima odlučuju kada će crveni tim napasti i kada će izvršiti ubacivanja, brinu se oko bodovanja timova, paze da svi sudionici poštivaju pravila i komuniciraju s njima. U bijelom timu postoji i član koji se naziva *blonde user*. On glumi nepažljivog člana plavog tima u scenarijima gdje se neke korisničke akcije ne mogu automatizirati, primjerice otvaranje elektroničke pošte zlonamjernog sadržaja ili otvaranje stranica sa zlonamjernim sadržajem. Plavi tim po pravilima vježbe takve korisnike ne smije zaustaviti u njihovim radnjama, ali bi trebao prepoznati i prikladno reagirati na rezultat njihovih postupaka.

Zeleni tim je zadužen za pripremu i održavanje sustava i infrastrukture na kojima se održava vježba. Također je zadužen za pružanje tehničke podrške tijekom izvođenja vježbe u slučaju da je to potrebno.

Žuti tim je zadužen da nadzire napredak crvenog i plavog tima kroz vježbu te o tome obavještava bijeli tim. Glavni izvor informacija su izvještaji plavog tima o obrani, crvenog tima o provođenju napada te automatski i ručni rezultati bodovanja timova.

Na slici 3.3. je prikazan tok kojim informacije putuju između timova za vrijeme provođenja vježbe.



Slika 3.3. Tok informacija u fazi provođenja vježbe

Bijeli tim je zadužen da zadaje zadatke crvenom timu. Crveni tim u vježbama obrane glumi napadače i zadužen je da izvršava događaje zadane u scenariju po uputama bijelog tima. Plavi tim je tim koji se brani od napada za vrijeme vježbe. Žuti tim nije aktivni dio sudionika već samo dojavljuje bijelom timu što se dešava te javlja povratne informacije plavom timu.

3.4. Scenarij

Scenarij je dokument koji s dovoljno detalja objašnjava slijed događaja koji se odvijaju u vježbi i okolinu u kojoj se izvršava vježba. Scenarij oblikuje vježbu tako da se može lakše prezentirati sudionicima. Sudionici se stavljaju u realnu situaciju u kojoj moraju obraniti ili napasti sustave. U vježbama obrane se plavi tim često stavlja u ulogu sigurnosnog tima za spašavanje neke izmišljene države kojoj prijeti opasnost. Plavi tim treba osigurati infrastrukturu, istražiti potencijalnu krađu podataka i surađivati s ostalim timovima. U sklopu scenarija često postoji i simulacija medijskih portala koji izvještavaju o novostima, a sudionici mogu imati račune na društvenim mrežama. Dio scenarija je i kako izgleda mrežna topologija, koji su programi instalirani na sustavima ili koji su podaci za pristup tim sustavima. Vježba mora pratiti slijed događaja iz scenarija, u njemu je zapisano kada i što crveni tim treba napadati, kada će se izvršiti ubacivanja i koje su zadaće plavog tima.

Ubacivanja možemo podijeliti na četiri kategorije:

- scenarijska,
- medijska,
- pravna,
- forenzička.

Scenarijska ubacivanja su pripremljena od strane bijelog tima i sadrže upute sudionicima. Primjerice plavom timu da prati vijesti, skupljaju podatke o kibernetičkim napadima ili napišu izvještaj.

Funkcija medijskih ubacivanja je da u vježbi stvori medijsko okruženje kakvo postoji i u stvarnom svijetu te tako oteža posao plavom timu. U ubacivanjima su vijesti o kibernetičkim događajima koji su se dogodili u vježbi, negativni komentari o trenutnim napadima ili lažne vijesti o sudionicima.

Pravna ubacivanja testiraju sposobnost plavog tima da odgovara na pravna pitanja s kakvim bi se susreli u stvarnom životu. Pravni savjetnik plavog tima mora informirati svoj tim o pravnim obvezama koje plavi tim mora poštovati.

Forenzička ubacivanja trebaju dati odgovore na pitanja koja se tiču kibernetičkog napada koji se dogodio, to su pitanja poput „tko je to napravio?”, „što se dogodilo?”, „kada se dogodilo?”, „kako se dogodilo?” i „zašto se dogodilo?”.

3.5. Bodovanje timova

Bodovanje timova je jedna od najproblematičnijih komponenti u vježbi. Iako se sustav bodovanja pokušava standardizirati, gotovo uvijek se javljaju sumnje od strane timova zbog toga što se bodovi uglavnom dodjeljuju od strane bijelog tima. Zbog toga se

bodovanje izostavlja iz mnogo vježbi uz obrazloženje da bodovanje nije glavna svrha organiziranja kibernetičkih vježbi nego stvaranje natjecateljske atmosfere zbog koje će se svi više potruditi. Jedan od primjera vježbe gdje se ne koristi sustav bodovanja je *Cyber Europe*. Unatoč tome, sustav bodovanja timova u takvim vježbama je prepoznat kao motivacijski alat za sudionike, a stvaranje pozitivne natjecateljske atmosfere među timova doprinosi i uspješnijim rezultatima. Primjer vježbe u kojoj se koristi sustav bodovanja je *Locked Shields*. U toj vježbi bodovanje se temelji na osam različitih kategorija:

- *raspoloživost sustava* – plavi tim ima popis sustava čija se raspoloživost konstantno provjerava od strane sustava za bodovanje i koji za dobivanje pozitivnih bodova moraju biti raspoloživi;
- *bonus za poštivanje ugovora o razini usluge (engl. service level agreement)* – ako je raspoloživost sustava bila veća od 90% onda plavi tim dobiva dodatne bodove;
- *uspješni napad crvenog tima* – za svaki uspješni napad crvenog tima plavom timu se oduzimaju bodovi, crveni tim može više puta ponavljati napade;
- *kratki izvještaj o incidentima* – izvještaj plavog tima kojeg šalju svakih sat vremena;
- *izvještaj upraviteljima* – izvještaj plavog tima kojeg šalju dva puta dnevno;
- *odgovori na ubacivanja* – svi odgovori na ubacivanja se zasebno boduju od strane bijelog tima;
- *traženje pomoći od zelenog tima* – svaki put kad bi plavom timu bila pružena tehnička pomoć zbog njihovih grešaka oduzeli bi im se bodovi;
- *posebna bodovanja* – pozitivni bodovi se dodjeljuju za međusobnu suradnju i dijeljenje informacija, a negativni za kršenje pravila.

3.6. Praćenje vježbe

Praćenje vježbe i zapisivanje događaja je osnova sustava bodovanja i pomaže u identificiranju i rješavanju problema koji se događaju u vježbi. Kibernetičke vježbe se odvijaju u ograničenom vremenskom trajanju, a za vrijeme trajanja vježbe se događa mnogo napada i mrežne aktivnosti što onemogućava organizatorima da trenutačno znaju sve što se događa. Zbog toga treba uspostaviti sustave za praćenje kako bi se u slučaju poteškoća znalo gdje je nastao problem i omogućilo zelenom timu da u što kraćem roku riješi problem. Sustav za praćenje je također potreban i za pisanje izvještaja nakon akcije u kojem je detaljno opisana vježba.

4. Implementacija kibernetičkog poligona uz pomoć alata otvorenog koda

U sastavu kibernetičkog poligona mogu se uočiti sljedeće komponente koje su opisane u poglavlju Arhitektura kibernetičkog poligona:

- simulirana mreža,
- program za izvršavanje scenarija,
- generator prometa,
- virtualni sigurnosni operativni centar (*engl. soc*),
- alati za instruktora.

Uz navedene komponente potrebno je i negdje pokrenuti infrastrukturu na kojoj će se podesiti sve te komponente koje čine jedan kibernetički poligon. Stvaranje i podešavanje infrastrukture je prvi i najvažniji korak, kibernetički poligon ne mora imati sve navedene komponente da bi bio funkcionalan, ali infrastrukturu na kojoj će se ljudi uvježbavati mora imati. Kako kibernetički poligoni često imaju i više stotina poslužitelja povezanih u jednu simuliranu okolinu potrebno je automatizirati njihovo stvaranje i podešavanje.

U ovom poglavlju će biti opisani načini na koje se pomoću alata otvorenog koda može implementirati svaka komponenta koja čini kibernetički poligon.

4.1. Stvaranje i podešavanje infrastrukture

Za svaku vježbu na kibernetičkom poligonu potrebno je izgraditi okolinu na kojoj će se ta vježba izvesti. To podrazumijeva kreiranje poslužitelja i njihovo podešavanje, stvaranje mreže, sigurnosnih pravila i sve ostalo što je potrebno da bi se vježba mogla izvesti. Taj proces bi bio dugotrajan i morao bi se iznova ponavljati za svaku vježbu bez da se napravi nekakva automatizacija tog procesa. Potrebno je odvojiti korak stvaranja od koraka podešavanja infrastrukture, iako neki alati imaju mogućnost i stvaranja i podešavanja infrastrukture, većinom ne rade toliko dobro obje stvari nego su specijalizirani samo za jednu.

Kako bi automatizirali stvaranje infrastrukture možemo koristiti alate Terraform [8] i Foreman [36].

Terraform je alat otvorenog koda koji služi za sigurno i brzo stvaranje, mijenjanje i verzioniranje infrastrukture. Može upravljati komponentama niže razine poput računalnog sklopovlja, prostorom za pohranu, mrežom, kao i komponentama više razine poput zapisa domena (*engl. domain name system ili DNS*). Alat Terraform omogućuje korisnicima da definiraju i stvore kompliciranu infrastrukturu koristeći samo konfiguracijski jezik poznat kao HCL (*engl. Hashicorp Configuration Language*), ili opcionalno JSON *(engl. JavaScript Object Notation)*.

Ovaj oblik stvaranja infrastrukture se još naziva i infrastruktura kao kod *(engl. Infrastructure as a code)* zbog toga što je infrastruktura opisana u jeziku visoke razine te se može verzionirati poput bilo kojeg drugog programskog koda.

Prilikom prvog pokretanja Terraform kreira plan izvršavanja koji opisuje što će napraviti kako bi doveo infrastrukturu u željeno stanje. Stvara se graf svih resursa te se paralelizira stvaranje i promjena bilo kojih resursa koji ne ovise jedan o drugom. Zbog toga Terraform brzo i učinkovito stvara infrastrukturu te se može vidjeti međuovisnost resursa. Kako se željeno stanje kibernetičkog poligona mijenja ovisno o promjenama na infrastrukturi Terraform može prepoznati što se promijenilo i stvoriti inkrementalni plan izvršavanja.

Foreman je alat koji služi za kompletno upravljanje infrastrukturom tijekom cijelog životnog vijeka. Može stvarati, podešavati i pratiti stanje fizičkih i virtualnih poslužitelja. Pomoću dodataka je integriran s alatima za podešavanje infrastrukture i to mu omogućava da automatizira ponavljajuće zadatke, podesi programe i upravlja s drugim promjenama na infrastrukturi.

Foreman posjeduje grafičko *web* sučelje pomoću kojeg se može upravljati s infrastrukturom te programabilno *web* sučelje (*engl. web application programming interface ili web API*) pomoću kojeg se može programski upravljati infrastrukturom.

Za podešavanje infrastrukture koja se stvori u prethodnom koraku može se koristiti alat Puppet [42].

Puppet je alat otvorenog koda kojem je glavna svrha konfiguriranje već pokrenutih programa. Podrška postoji za Linux, Windows i Unix operacijske sustave te se pomoću njega mogu automatizirati ponavljajući zadatci poput dodavanja korisnika, instaliranja paketa ili mijenjanja konfiguracije poslužitelja.

Puppet funkcionira tako da postoji jedan glavni poslužitelj i više klijentskih poslužitelja. Klijenti komuniciraju s glavnim poslužiteljem i prate promjene u konfiguraciji. Ako otkriju neku promjenu onda spreme novu konfiguraciju i primijene sve potrebne promjene.

4.2. Pokretanje infrastrukture

Kako se kibernetički poligoni koriste za uvježbavanje ljudi potrebno je jako često uklanjati i ponovno stvarati te podešavati infrastrukturu na početne postavke na kojoj će se ti ljudi uvježbavati i iznova ponavljati vježbe. Za tu svrhu potrebno je imati mogućnost jednostavnog stvaranja, pokretanja i gašenja infrastrukture. Pogodno za takvu upotrebu je računarstvo u oblaku *(engl. cloud computing)* i proces virtualizacije.

Računarstvo u oblaku je mogućnost korištenja računalnih resursa prema zahtjevima korisnika, posebice diskovnog prostora i procesorskog vremena. To znači da korisnik ne mora imati fizički zakupljenu infrastrukturu i plaćati ju stalno nego da ju može koristiti kada je to potrebno i platiti onoliko koliko je koristio. Tako je omogućeno manjim tvrtkama da s minimalnim troškovima započnu posao i da ne plaćaju unaprijed resurse koje

vjerojatno neće potrošiti. Mana tog pristupa je da u slučajevima velike neočekivane potrošnje računi za korištenje resursa jednostavno budu preveliki i da tvrtka nema novca da ih plati. Računarstvo u oblaku se tako zove zato što krajnji korisnik uopće ne zna gdje se infrastruktura zapravo nalazi, zna samo da joj pristupa preko interneta, riječ oblak u imenu je zapravo asocijacija na internet. Trenutačno najpoznatija komercijalna usluga za računarstvo u oblaku je *Amazon Web Services* ili *AWS* [43]. Moguće je uspostaviti i svoj privatni oblak, nedostatak toga je što se treba brinuti o vlastitoj infrastrukturi. Najpoznatiji alat otvorenog koda za stvaranje privatnih oblaka je OpenStack [44].

Virtualizacija je način pokretanja računala pomoću kojeg se na jednom fizičkom računalu može pokrenuti više operacijskih sustava odjednom. Najveća prednost virtualizacije je učinkovitije korištenje fizičkog sklopovlja, moguće je pokrenuti više virtualnih računala na samo jednom računalu umjesto na više njih.

Najpoznatije platforme za kreiranje virtualiziranih okolina na bazi otvorenog koda su:

1. KVM (*engl. Kernel-based Virtual machine*) [45],
2. Oracle VirtualBox [46],
3. XEN [47].

KVM je virtualizacijski modul u Linux jezgri koji joj omogućava da se ponaša kao upravljač za virtualizaciju. Zbog toga se može izvršavati samo na Linux operacijskom sustavu, odnosno domaćin (*engl. host*) može biti samo poslužitelj s operacijskim sustavom Linux. KVM zahtjeva procesor s virtualizacijskom ekstenzijom poput Intel VT za Intel procesore i AMD-V za AMD procesore. KVM je tip 2 vrsta virtualizacijskog upravljača, to znači da se instalira kao program na postojeći operacijski sustav. Zbog toga što je KVM jezgrin modul on se zapravo pokreće kao tip 1 vrsta virtualizacijskog upravljača i tako direktno pristupa hardware-u računala.

Oracle VirtualBox je programsko rješenje napravljeno za općenitu upotrebu za x86 arhitekturu procesora. Dolazi u obliku programa koji ima grafičko sučelje, namijenjen je za sve operacijske sustave, tako da se može instalirati na Linux, Windows. MacOS, Solaris i OpenSolaris operacijske sustave. Pogodan je za lokalnu upotrebu na stolnim računalima zato što se virtualne slike računala distribuiraju kao pojedina datoteka s ekstenzijom OVA. To omogućava da na vrlo jednostavan način preko grafičkog sučelja dodamo virtualnu sliku računala u program i pokrenemo ju. VirtualBox je virtualizacijski upravljač tipa 2 što znači da se on može koristiti samo na već instaliranom operacijskom sustavu.

XEN je virtualizacijski upravljač koji podržava x86, x86_64, Itanium i ARM arhitekture procesora i može pokretati Linux, Windows, Solaris i neke od BSD obitelji operacijskih sustava. XEN je tip 1 virtualizacijskog upravljača što znači da se pokreće neovisno o operacijskom sustavu, time ima veće privilegije nego bilo koji program na operacijskom sustavu i direktan pristup sklopovlju računala.

4.3. Simulirana mreža

U virtualiziranim okruženjima računala su međusobno povezana preko virtualne lokalne mreže (*engl. Virtual Local Area Network ili VLAN*). Računala u istom VLAN-u će moći komunicirati jedna s drugima. Stvaranje VLAN-a i povezivanje računala u istom VLAN-u je ugrađeno u alate za stvaranje virtualizacije. U kibernetičkim vježbama obrane to nije dovoljno jer se treba simulirati stvarni svijet, crveni i plavi tim trebaju biti u odvojenim mrežama, baš kao i u stvarnom svijetu gdje napadači često napadaju izvana.

Komunikacija između različitih VLAN-ova se ostvaruje preko virtualnih usmjerivača (*engl. virtual router*). Virtualni usmjerivač se u simuliranu mrežu postavlja kao jedno od virtualnih računala koje radi poput pravog usmjerivača. Virtualni usmjerivač ima mrežno sučelje koje je povezano na svaki od VLAN-ova koje je potrebno međusobno povezati [37].

Jedan od poznatijih virtualnih usmjerivača otvorenog koda je pfSense [48]. PfSense je napravljen na temelju operacijskog sustava FreeBSD s izmijenjenom jezgrom i predinstaliranim besplatnim programima s kojima proširuje mogućnosti. To je kombinirani alat koji može raditi i kao vatrozid i usmjerivač. Potrebno ga je pokrenuti na fizičkom ili virtualiziranom računalu. Posjeduje *web* grafičko sučelje pomoću kojeg se može upravljati bez znanja o operacijskom sustavu koji upravlja s njim.

4.4. Program za izvršavanje scenarija

Najbitniji dio scenarija u vježbama obrane su napadi od kojih se plavi tim treba braniti. U tim vježbama napade najčešće ručno izvodi crveni tim kao u vježbi *Locked Shields*. Zbog toga se povećava broj ljudi koji trebaju sudjelovati u vježbi, a to kao rezultat ima veću kompleksnost vježbe, nepotrebno trošenje ljudskih resursa i manja mogućnost ponavljanja istog napada. Postoje brojna istraživanja na temu automatiziranja napada.

Stručnjaci iz Švedske agencije za istraživanje obrane su za potrebe CRATE kibernetičkog poligona stvorili alat pod imenom SVED (*engl. Scanning, Vulnerabilities, Exploits and Detection*) [38]. Napad u alatu SVED je prikazan kao niz logičkih koraka koji skupa čine graf napada. Alat SVED nažalost nije otvorenog koda i nije proučen detaljnije.

Jedan od alata otvorenog koda koji automatizira napade je DEW (*engl. Distributed Experiment Workflow*) [39]. Pomoću alata DEW se može napisati specifikacija scenarija u formatu koji ljudi mogu razumjeti, a računala obraditi. Moguće je automatizirati bilo kakve zadatke, ne samo napade, tako da se pomoću ovog alata može automatizirati i ponašanje *blonde usera* koji u nekim scenarijima predstavlja nepažljivog korisnika koji primjerice pokrene zlonamjernu datoteku.

4.5. Generator prometa

Generator prometa se može implementirati kao računalni program koji mora biti spojen na cijelu simuliranu mrežu kibernetičkog poligona. Neki komercijalni kibernetički poligoni poput Ixia imaju fizički uređaj kojemu je funkcija samo generiranje prometa. Za potrebe nekog jednostavnijeg kibernetičkog poligona može se koristiti alat otvorenog koda Ostinato [40].

Ostinato ima grafičko sučelje za jednostavnije korištenje i API za programsko korištenje. Ono što ga čini odličnim kandidatom da ga se implementira kao generator prometa za kibernetički poligon je njegova kontroler – agent arhitektura. Napravljen je tako da se može postaviti u način rada kada je jedan program kontroler, a svi ostali programi agenti. Agenti se mogu postaviti na različita računala odakle čekaju naredbe od kontrolera i prema tome generiraju promet.

4.6. VSOC

VSOC bi trebalo implementirati tako da ima *web* sučelje. Preko *web* sučelja se može brzo i jednostavno pristupiti VSOC-u. Najbitniji dio VSOC-a je sustav za sigurnost informacija i upravljanje događajima (*engl. Security Information and Event Management ili SIEM*). SIEM je program preko kojeg se skuplja i analizira sva aktivnost koja se događa na nekoj infrastrukturi. Računalni stručnjaci tako mogu na jednom mjestu prikupiti sve informacije, dobiti obavijesti i upozorenja pri otkriću zlonamjernih aktivnosti te reagirati na njih. Kao VSOC se može koristiti alat AlienVault OSSIM [41]. AlienVault Ossim je stvoren zbog nedostatka kvalitetnih SIEM programa otvorenog koda. Program ima *web* sučelje preko kojeg se upravlja i niz međusobno povezanih alata otvorenog koda koji služe za skupljanje informacija.

4.7. Alati za instruktora

Instruktor bi trebao imati *web* sučelje preko kojeg može upravljati kibernetičkim poligonom i nadzirati izvođenje kibernetičke vježbe. To *web* sučelje mora biti u potpunosti izrađeno ispočetka i povezano sa svakom komponentom koja čini kibernetički poligon. Sučelje mora biti napravljeno iz početka zato što sve komponente koje čine kibernetički poligon nisu standardizirane pa se ne zna njihova točna implementacija unaprijed. Zbog toga ne postoji neki alat otvorenog koda koji bi pomogao u izradi alata za instruktora nego se u ovisnosti o potrebama projekta mogu koristiti bilo koje dostupne tehnologije.

5. Kibernetički poligoni otvorenog koda

U ovom poglavlju bit će predstavljeni kibernetički poligoni otvorenog koda. U prvom potpoglavlju opisan je *AWS Cyber Range* [49], u drugom potpoglavlju je opisan *Open Cyber Challenge Platform* [50] ili *OCCP,* a na kraju je opisana *Network Defense* [51] vježba na *OCCP* platformi.

5.1. AWS Cyber range

AWS Cyber Range projekt je predstavljen kao prvi nacrt (*engl. blueprint*) otvorenog koda za izradu kibernetičkih poligona u svijetu. Pruža radni okvir i alate za učenje iz područja poput:

- napadačke sigurnosti (*engl. offensive security*),
- obrambene sigurnosti (*engl. defensive security*),
- obrnutog inženjerstva (*engl. reverse engineering*),
- sigurnosno-obavještajne inteligencije (*engl. security intelligence*).

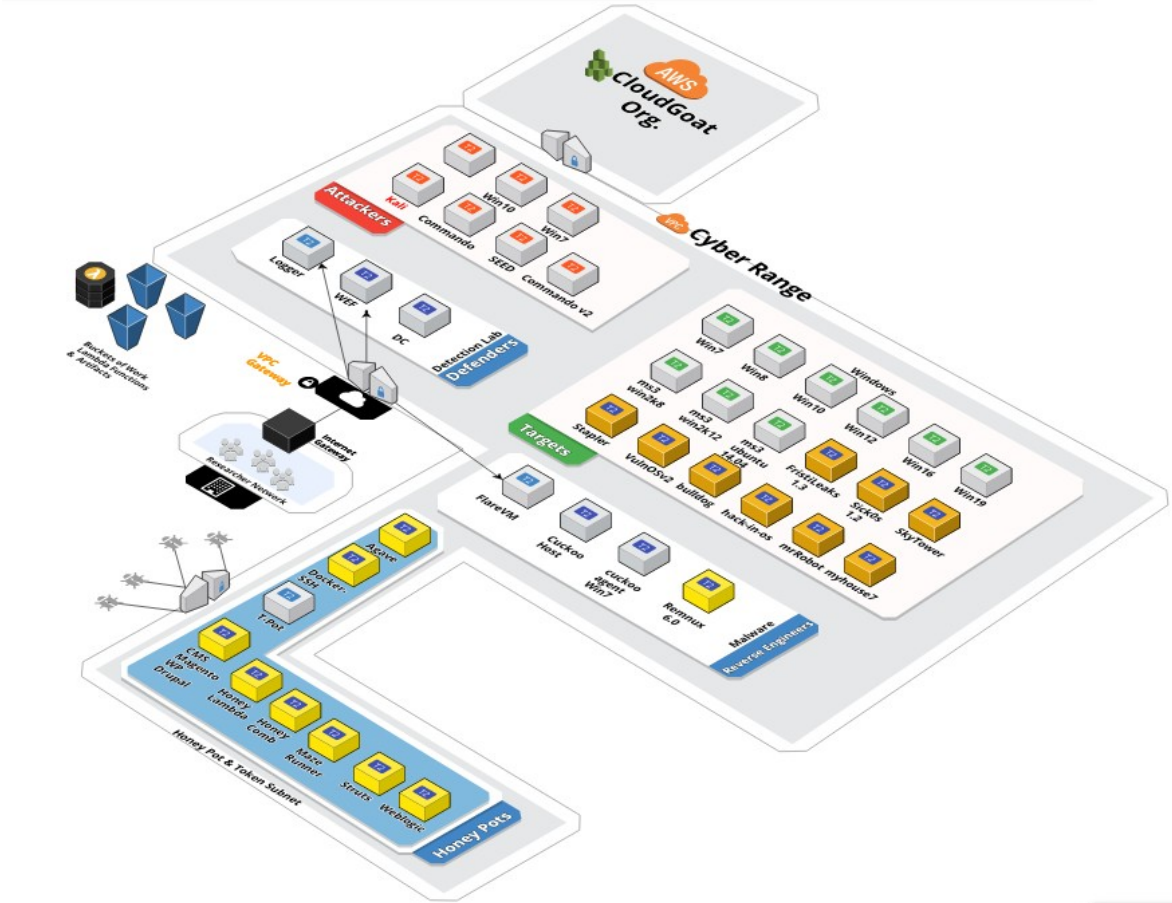
Projekt sadrži ranjive sustave i skup alata otvorenog koda na kojima se mogu učiti navedena područja. Kibernetički poligon se pokreće na *AWS* oblaku, a samo stvaranje okruženja od pedeset poslužitelja traje manje od pet minuta . Za stvaranje infrastrukture pokrenute na *AWS* oblaku koristi se alat Terraform. Alat Terraform omogućava da se infrastruktura kibernetičkog poligona opiše pomoću konfiguracijskog jezika HCL, a onda alatom Terraform stvori. U ispisu 5.1. je prikazan opis jednog virtualnog računala konfiguracijskim jezikom HCL koje se stvara u *AWS Cyber Range* kibernetičkom poligonu.

```
1 resource "aws_instance" "kali" {
2  count = "${var.kali_ct}"
3  ami = "${data.aws_ami.kali.id}"
4  instance_type = "${var.instance_type_kali}"
5  subnet_id = "${element(local.pen_subnet_ids, count.index)}"
6  vpc_security_group_ids = ["${aws_security_group.kali.id}"]
7  key_name = "${aws_key_pair.circleci_key.key_name}"
8  user_data = "${file("${path.module}/../../cloud-init/kali.yml")}"
9  root_block_device {
10     delete_on_termination = true
11     volume_size           = 160
12     }
13     tags = {
14     Name = "kali-linux-${count.index}"
15     Environment = "${var.environment}"
16     Terraform = "True"
17     }
18 }
```

Ispis 5.1. Opis virtualnog računala HCL konfiguracijskim jezikom [49]

U ispisu 5.1. prvi red označava da se sljedeći redci odnose na stvaranje infrastrukture na *AWS* oblaku, drugi i treći redak označavaju broj instanci poslužitelja koji će se stvoriti iz slike virtualnog računala spremljenog na *AWS* oblaku dok se u četvrtom retku definira tip, odnosno koliko resursa će dobiti to virtualno računalo. Peti redak označava kojem će VLAN-u pripadati stvoreno računalo dok šesti redak definira sigurnosne postavke koje će se primijeniti. Sedmi redak označava privatni ključ pomoću kojeg će se moći spojiti na poslužitelj. U osmom retku se definira skripta preko koje se obično instaliraju i podese dodatne stvari koje su izvan mogućnosti Terraform-a. Od devetog do dvanaestog retka se definiraju podaci o prostoru za pohranu, a od trinaestog do šesnaestog retka su razni metapodaci.

Prilikom stvaranja projekta vodilo se računa da troškovi pokretanja infrastrukture u *AWS* oblaku koštaju što manje. *AWS* svojim korisnicima nudi određene računalne resurse besplatno (*engl. free tier*) te *AWS Cyber Range* iskorištava te pogodnosti tako da većinu infrastrukture pokreće tako da računalni resursi ne premašuju besplatnu kvotu. Tako se cijeli *AWS Cyber Range* od pedeset poslužitelja može pokrenuti za otprilike jedan dolar po satu. Troškovi se mogu dodatno smanjiti tako da se prilikom pokretanja kibernetičkog poligona izuzmu poslužitelji koji se naplaćuju i nisu potrebni za potrebe vježbanja. Na slici 5.1. je prikazana cijela arhitektura *AWS Cyber Range* projekta kada se pokrene.



Slika 5.1. AWS Cyber Range arhitektura [49]

Slika 5.1. prikazuje kompletnu arhitekturu *AWS Cyber Range* kibernetičkog poligona nakon što se pokrene u oblaku. Pokrenuta infrastruktura se može podijeliti na sljedeće grupe poslužitelja i virtualnih računala prema njihovoj ulozi u kibernetičkom poligonu:

- napadačka,
- obrambena,
- obrnuto inženjerstvo,
- mete,
- mamci (*engl. honeypot*).

Napadačka infrastruktura sadrži poslužitelje i virtualna računala koja imaju instalirane alate otvorenog koda koje napadači najčešće koriste tijekom napada. Obrambena infrastruktura sadrži poslužitelje koje branitelji koriste za nadziranje mreže i odgovaranje na napade. Na virtualnim računalima za obrnuto inženjerstvo branitelji mogu isprobavati i analizirati datoteke za koje misle da sadrže zlonamjerni kod. Mete su poslužitelji i virtualna računala koja simuliraju normalne korisnike, a sadrže namjerno postavljene ranjivosti koje napadači mogu iskoristiti. Mamci su poslužitelji koji sadrže namjerno postavljene ranjivosti, ali koje su postavljene tako da ih napadači ne mogu iskoristiti. Mamci omogućuju braniteljima da lakše otkriju napadače, a da pritom nema rizika da se ošteti njihova infrastruktura.

AWS Cyber Range, iako koristi najmodernije tehnologije i alate, za sada nema neke napredne mogućnosti koje imaju komercijalni kibernetički poligoni. Nema generator prometa iako bi se isti mogao lako dodati, ne postoji mogućnost pokretanja scenarija kao ni alate za instruktora preko kojih bi se mogle nadzirati kibernetičke vježbe i upravljati s njima.

5.2. OCCP

OCCP je besplatna, konfigurabilna platforma otvorenog koda napravljena na *Rhode Island* fakultetu. Platforma je razvijena s ciljem da uz male troškove omogući pojedincima vježbanje i učenje o vještinama iz područja računalne forenzike i kibernetičke sigurnosti. Pomoću platforme se mogu vježbati sljedeća područja:

- mrežna obrana,
- penetracijska testiranja,
- sigurno programiranje,
- računalna forenzika.

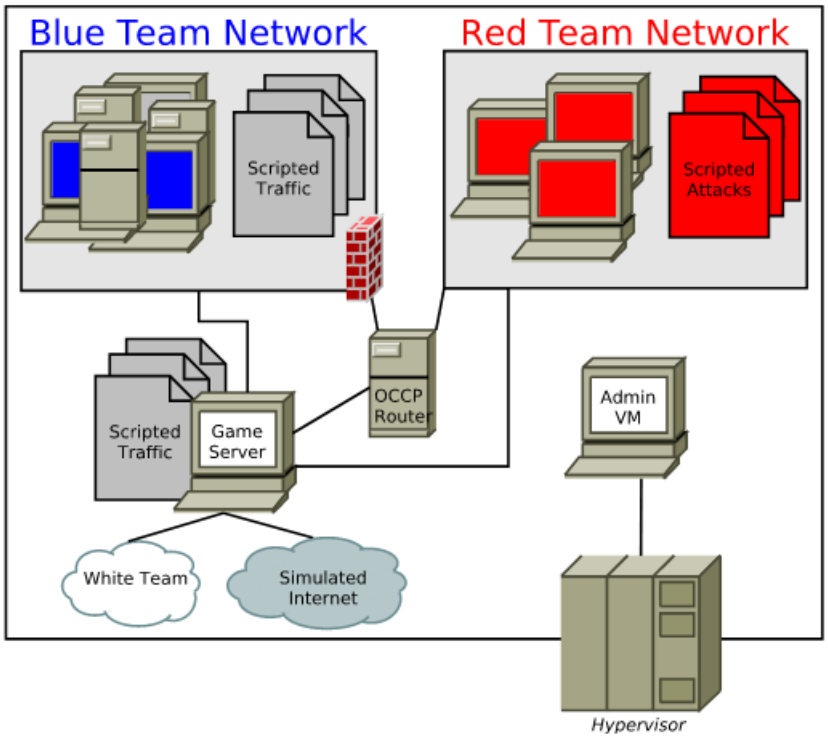
Uvježbavanje mrežne obrane je zamišljeno tako da su napadi crvenog tima automatizirani, a zadatak plavog tima je da se brani od tih napada. Negativni bodovi su dodijeljeni plavom timu u slučaju krađe podataka i onemogućavanju usluge od strane crvenog tima.

Penetracijska testiranja se mogu vježbati tako da je ponašanje plavog tima automatizirano. Crveni tim su studenti i oni se pozitivno boduju u slučaju krađe podataka i onemogućavanja usluge.

U vježbama sigurnog programiranja plavi tim su studenti, a crveni tim je automatiziran. Zadaća plavog tima je da pronađe programe koji imaju namjerno ubačene sigurnosne ranjivosti i ispravi ih. Negativni bodovi im se dodjeljuju za svaki uspješno izvršeni napad.

U vježbama računalne forenzike crveni tim je automatiziran, a plavi tim mora otkriti koji su podaci ukradeni i tko ih je ukrao.

OCCP projekt infrastrukturu pokreće pomoću virtualizacije, a podržava *Oracle VirtualBox* i *Vmware ESXi* tehnologije. Platforma se sastoji od više virtualnih računala od kojih svako ima svoju ulogu. Na slici 5.2. je prikazana arhitektura *OCCP* platforme.



Slika 5.2. Arhitektura *OCCP* platforme [50]

OCCP platforma je podijeljena na sljedeće komponente:

- administracijsko virtualno računalo,
- poslužitelj igre,
- virtualna simulirana mreža.

Administracijsko virtualno računalo je odvojeno virtualno računalo koje se prvo pokreće na jednoj od podržanih virtualizacijskih tehnologija, *Oracle VirtualBox* ili *Vmware ESXi.* Instruktori koji organiziraju kibernetičku vježbu koriste administracijsko računalo prije početka vježbe. Na administracijsko računalo se prvo učitava paket scenarija koji sadrži sve potrebno za održavanje kibernetičke vježbe.

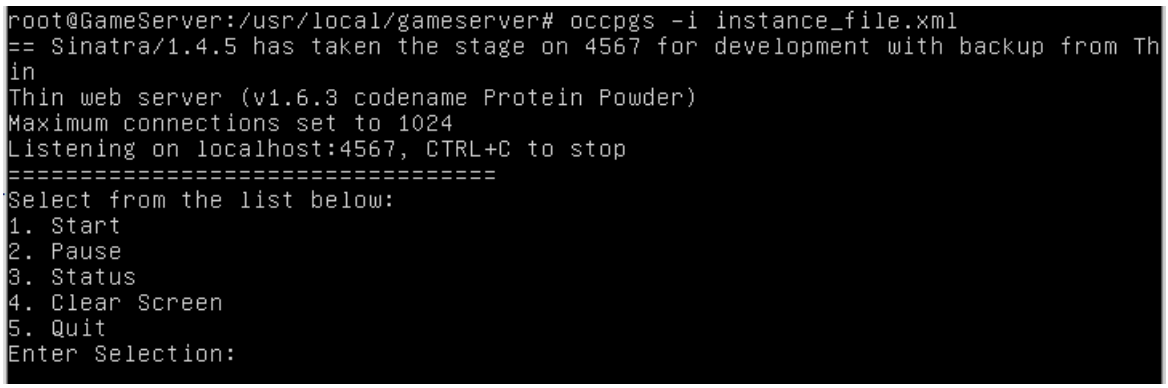
Administracijsko računalo sadrži konzolni program *OccpAdmin* napravljen posebno za *OCCP* platformu kako bi olakšao instalaciju scenarija. *OccpAdmin* je zadužen za:

- stvaranje poslužitelja igre,
- stvaranje virtualne simulirane mreže,
- ponovno podešavanje virtualnih računala ako je to potrebno,
- uspostavljanje virtualne privatne mreže za pristup udaljenim računalima.

Nakon što program *OccpAdmin* stvori i podesi sve što je potrebno za održavanje kibernetičke vježbe, instruktor se treba spojiti na poslužitelj igre i pokrenuti vježbu. Poslužitelj igre je zadužen za:

- pokretanje scenarija,
- upravljanje sa službenim satom igre,
- izvršavanje događaja scenarija u pravo vrijeme – to uključuje pokretanje automatiziranih timskih skripti, pokretanje skripti koje simuliraju crveni ili plavi tim,
- generiranje prometa,
- praćenje rezultata,
- komunikaciju sa sudionicima.

Poslužitelj igre sadrži konzolni program *OccpGs* pomoću kojeg se pokreće kibernetička vježba. *OccpGs* program pruža jednostavno korisničko iskustvo za instruktore. Na slici 5.3. je prikazano sučelje programa OccpGs nakon učitavanja scenarija.



Slika 5.3. Sučelje programa *OccpGs*

Program *OccpGs* je jednostavan za korištenje i pruža osnovne mogućnosti za upravljanje kibernetičkom vježbom, mogu se dohvatiti jednostavne informacije o status vježbe te se može pokrenuti, pauzirati ili zaustaviti vježba.

5.3. Network Defense vježba

Network Defense je jedina vježba stvorena za *OCCP* platformu. Cilj vježbe je vježbanje obrane od napada. Vježba dolazi u obliku datoteke koja se naziva paket scenarija, to je komprimirana datoteka s ekstenzijom tar. Paket scenarija sadrži sve potrebno da bi se vježba stvorila i pokrenula poput dokumentacije za instruktore i sudionike, slike virtualnih računala koja čine vježbu i datoteku scenarija.

Datoteka scenarija je najbitniji dio paketa scenarija i same vježbe. U njoj je definirano koja virtualna računala čine simuliranu mrežu, kako se ona trebaju podesiti te koje događaje i u kojem vremenu ih treba izvršiti poslužitelj igre. Datoteka scenarija dolazi u obliku datoteke s ekstenzijom xml. U OccpAdmin se prosljeđuje datoteka scenarija te ju program obradi i proslijedi na poslužitelj igre koji je zadužen za pravovremeno izvršavanje događaja definiranih u datoteci scenarija.

U ovoj vježbi se uvježbava plavi tim. Plavi tim nakon pokretanja dobije podatke o poslužiteljima koje treba štititi. Crveni tim je automatiziran te napade crvenog tima pokreće poslužitelj igre. Postoji i sivi tim koji je također automatiziran pomoću skripti, a njegova uloga je da generira promet u vježbi. U igri još sudjeluje bijeli tim koji čine instruktori koji nadziru odvijanje vježbe i brinu se oko bodovanja.

Network Defense vježba traje šezdeset minuta te crveni tim pokreće sedam različitih napada koji se ponavljaju u ovisnosti o parametrima zadanim u datoteci scenarija. Napadi crvenog tima poredani kronološki su:

1. napad ljuske poslužitelja grubom silom,
2. indeksiranje *(engl. crawl) web* poslužitelja,
3. napad grubom silom u potrazi za direktorijima *web* poslužitelja,
4. iskorištavanje ranjivosti *web* aplikacije,
5. zamjena početne datoteke *web* aplikacije,
6. sql umetanje u jednu od datoteka *web* aplikacije.

U ispisu 5.2. je prikazano kako je zapisan napad iskorištavanja ranjivosti web aplikacije u datoteci scenarija.

```
<team-event
   name="egallery deface" id="" guid="" handler="exec-handler-1"
   ipaddress="hacker" starttime="900" endtime="9999999" frequency="900"
   drift="100"
   command="/opt/metasploit-framework/msfcli
            unix/webapp/egallery_upload_exec
            RHOST=${occp:www_server_ip}
            TARGETURI=/rhinos/
            PAYLOAD=php/exec
            CMD='rm index.php'
   ">
   <score-atomic when="success" score-group="redteam" points="5" />
   <score-atomic when="fail" score-group="redteam" points="-3" />
</team-event>
```

Ispis 5.2. Zapis iskorištavanja ranjivosti *web* aplikacije u *Network Defense* vježbi [50]

U zapisu se definira s koje IP adrese dolazi napad, kada počinje te kojom se frekvencijom ponavlja kroz vrijeme trajanja vježbe. Također se definira koja naredba izvršava taj napad i parametri koji su potrebni za uspješno izvršavanje napada. Na kraju zapisa se nalaze bodovi koji se pridjeljuju proizvoljno imenovanim grupama. Te grupe se mogu koristiti u formulama za bodovanje pojedinih timova. Crveni tim za uspješno izvršen napad dobiva pet bodova, a za neuspješno izvršen napad mu se oduzimaju tri boda.

Uz napade crvenog tima automatizirano je i ponašanje sivog tima koji simulira normalne korisnike koji koriste poslužitelje koje plavi tim brani. U ispisu 5.3. je prikazan zapis koji simulira kako član sivog tima šalje elektroničku poštu.

```
<team-event
   name="Email Grey" id="" guid="" handler="exec-handler-1"
   ipaddress="grey" starttime="0" endtime="9999999" frequency="60"
   drift="90" command="echo `${occp:random_email_usernames[]}' |
          sendmail.py ${occp:mail_server_ip}${occp:company_domain}">
   <score-atomic when="success" score-group="blueteam_service"
   points="5" />
   <score-atomic when="fail" score-group="blueteam_service"
   points="-3" />
</team-event>
```

Ispis 5.3. Zapis slanja elektroničke pošte u *Network Defense* vježbi [50]

U zapisu se definira s koje IP adrese, kada i kojom frekvencijom se šalje elektronička pošta. Također se definira koja skripta izvršava slanje elektroničke pošte. Na kraju zapisa je definirano koliko bodova se dodjeljuje odnosno oduzima posebnoj grupi u koju se spremaju bodovi za raspoloživost poslužitelja.

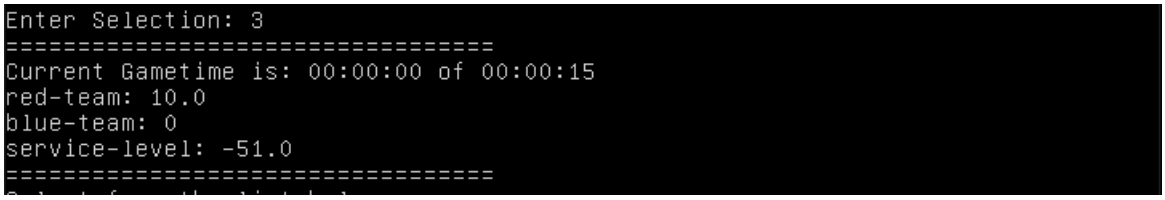
Bodovanje na *OCCP* platformi je modularno te se može za potrebe vježbe mijenjati. Dobiveni bodovi se spremaju u bazu podataka, a način bodovanja je definiran u datoteci scenarija. U ispisu 5.3. se nalazi način bodovanja u vježbi *Network Defense.*

```
<score-labels>
    <score-label name="blueteam5min" sql="select SUM(value) from score
where groupname='blueteam_service' AND (time &gt; (strftime('%s', 'now')
- 300))"/>
    <score-label name="blueteam_service" />
    <score-label name="redteam" />
</score-labels>

<score-names>
    <score-name name="blue-team" descr="Blue Team"
formula="blueteam_service - redteam" />
    <score-name name="red-team" descr="Red Team" formula="redteam" />
    <score-name name="service-level" descr="Service Level"
formula="blueteam_service" />
    <score-name name="service-level-5" descr="Service Level (Last 5
minute)" formula="blueteam5min" />
</score-names>
```

Ispis 5.4. Zapis o načinu bodovanja u *Network Defense* vježbi [50]

Prvo se definiraju oznake te vrijednosti iz baze koji se pridjeljuju svakoj oznaci. Nakon što se definiraju oznake one se mogu međusobno kombinirati kako bi se dobila formula za konačan broj bodova. Tako se dobije da je broj bodova plavog tima zapravo broj bodova koji je postigao crveni tim oduzet od broja bodova dodijeljenih za raspoloživost poslužitelja. Na slici 5.4. je prikaz izračunatih bodova iz sučelja programa *OccpGs*.



Slika 5.4. Prikaz izračunatih bodova u sučelju programa *OccpGs*

OCCP platforma je zbog dobre arhitekture i lake proširivosti imala potencijal da se razvije u kompleksan i potpun kibernetički poligon. Nažalost platforma je napravljena na starijim tehnologijama, a razvoj same platforme je stao prije par godina. Unatoč tome iz arhitekture platforme se može puno naučiti, te se ta znanja mogu prenijeti na izgradnju novog kibernetičkog poligona.

6. Zaključak

Ubrzani razvoj tehnologije i sve veća izloženost ljudi i različitih uređaja internetu dovela je do toga da se za internet govori da je peta bojišnica svjetskog ratovanja [52]. Kibernetički napadi na kritičnu infrastrukturu države mogu napraviti štetu jednaku kao i prirodne nepogode. Tvrtke više ne postavljaju pitanje hoće li, nego kada će biti napadnute, a njihovi podatci kompromitirani. Zbog načina kako internet funkcionira krivci za napade se mogu teško otkriti, a jedan od problema je i taj što je napade lakše izvesti nego se obraniti od njih. U tradicionalnom ratovanju branitelji su obično imali prednost dok su napadači morali ostvariti nadmoć u tehnologiji, brojnosti ili strategiji kako bi pobijedili. U kibernetičkom ratovanju to nije slučaj zato što napadači moraju samo jednom pobijediti dok se branitelji moraju konstantno braniti i unaprjeđivati.

Kako su se komplicirani kibernetički napadi počeli tek nedavno događati tako se pojavila i potreba za razvijanjem i vježbanjem vještina iz kibernetičke sigurnosti. Vještine iz područja kibernetičke sigurnosti je teško vježbati zato što se vježbe ne smiju odvijati na produkcijskim okolinama jer im se može naštetiti. Okoline na kojima se provode vježbe moraju biti potpuno kontrolirane i mora postojati način da se lako ponove. Zato se pojavila potreba za simuliranim okolinama koje to omogućuju.

Razvoj kibernetičkih poligona je omogućio sve navedene funkcionalnosti. Nažalost, takvi poligoni su komercijalizirani i dolaze s velikom cijenom. U ovom radu je istraženo što su točno kibernetički poligoni, koje su njihove mogućnosti i kako se organiziraju kibernetičke vježbe na tim poligonima. Istraženi su alati otvorenog koda i predložena su rješenja pomoću kojih se mogu izraditi komponente za vlastiti kibernetički poligon i značajno smanjiti cijena izrade jednog takvog poligona. Na kraju je istraženo koji kibernetički poligoni otvorenog koda postoje i pokazano je da trenutačno nema poligona koji se mogu po funkcionalnostima usporediti s komercijalnim rješenjima .

7. Literatura

[1] Mor Ahuvia, *Cyber Security Training Platform*, Cyberbit, 8.5.2019., https://www.cyberbit.com/blog/security-training/cyber-security-training-platform/, pristupljeno: 15.11.2019.

[2] Wikipedia, *Stuxnet*, https://en.wikipedia.org/wiki/Stuxnet, pristupljeno: 15.11.2019.

[3] Braje, Timothy, *Advanced Tools for Cyber Ranges,* MIT Lincoln Laboratory Lexington United States, 2016.

[4] Ixia, *Breakingpoint on Amazon Web Services — Cloud application and security testing*, https://www.ixiacom.com/sites/default/files/2019-08/Ixia-S-DS-BreakingPoint-on-AWS.pdf, pristupljeno: 15.11.2019.

[5] Ben Canner, *A successful security operations center (SOC) Framework in 6 questions*, Solutions Review, 7.6.2019., https://solutionsreview.com/security-information-event-management/a-successful-security-operations-center-soc-framework-in-6-questions/, pristupljeno: 20.11.2019.

[6] European Union, *Data protection under GDPR*, https://europa.eu/youreurope/business/dealing-with-customers/data-protection/data-protection-gdpr/index_en.htm, pristupljeno: 14.12.2019

[7] Cyberbit, *Security Automation and Orchestration*, https://www.cyberbit.com/solutions/security-operations-automation-orchestration/, pristupljeno: 17.11.2019.

[8] Hashicorp, *Terraform by Hashicorp*, https://www.terraform.io/, Pristupljeno: 07.01.2020.

[9] AIT, *AIT Cyber Range*, https://cyberrange.at/, pristupljeno: 28.11.2019.

[10] Cyberbit, *Cyber Range Training and Simulation*, https://www.cyberbit.com/solutions/cyber-range/, pristupljeno: 15.11.2019.

[11] Paloaltonetworks, *Why Cyber Range - Paloaltonetworks*, https://www.paloaltonetworks.com/solutions/initiatives/cyberrange-overview, pristupljeno: 15.11.2019.

[12] Ixia, *Cyber Range*, https://www.ixiacom.com/solutions/cyber-range, pristupljeno: 15.11.2019.

[13] Čeleda, Pavel, et al. "*KYPO–A Platform for Cyber Defence Exercises*." M&S Support to Operational Tasks Including War Gaming, Logistics, Cyber Defence. NATO Science and Technology Organization (2015).

[14] CRATE, *Cyber Range And Training Environment*, 17.1.2019., https://www.foi.se/en/foi/resources/crate---cyber-range-and-training-environment.html, pristupljeno: 20.11.2019.

[15] Peostr, *National Cyber Range*, https://www.peostri.army.mil/national-cyber-range-ncr. pristupljeno: 20.11.2019.

[16] Cyberbit, *Success Stories*, https://www.cyberbit.com/solutions/cyber-range/success-stories/, pristupljeno: 20.11.2019.

[17] Cyberbit, *Cloud Based Cyber Range*, https://www.cyberbit.com/solutions/cyber-range/cloud-based-cyber-range/, pristupljeno: 20.11.2019.

[18] Palo Alto Networks, *Palo Alto Networks Launches Global Cyber Range Initiative and Opens First Dedicated Facility in Amsterdam,* 15.1.2018.*,* https://www.paloaltonetworks.com/company/press/2018/global-cyber-range-initiative-and-opens-first-dedicated-facility-in-amsterdam, pristupljeno: 21.1.2019.

[19] Palo Alto Networks, *DIY Cyber Range*, https://www.paloaltonetworks.com/solutions/initiatives/cyberrange-diy, pristupljeno: 20.11.2019.

[20] Palo Alto Networks, *Cyber Defense Training Center And Exercises*, https://www.paloaltonetworks.com/apps/pan/public/downloadResource?pagePath=/content/pan/en_US/resources/techbriefs/cyber-range, pristupljeno: 20.11.2019.

[21] Cisco, *India - Cisco Inaugurates Cyber Range Lab In India*, 12.4.2017., https://newsroom.cisco.com/press-release-content?articleId=1839017, pristupljeno: 20.11.2019.

[22] Airbus, *Cyber Range*, https://airbus-cyber-security.com/products-and-services/prevent/cyberrange/, pristupljeno: 21.11.2019.

[23] Virginia Cyber Range, *Virginia Cyber Range*, https://www.virginiacyberrange.org/, pristupljeno: 21.11.2019.

[24] Merit, *The Michigan Cyber Range*, https://www.merit.edu/cyberrange/, 22.11.2019.

[25] IBM, *IBM X-Force Command Centers*, https://www.ibm.com/security/services/managed-security-services/security-operations-centers, 22.11.2019.

[26] Republic of Estonia, *Estonian Ministry of Defence launches Cyber Security Training Centre*, 8.4.2019., https://www.kaitseministeerium.ee//en/news/estonian-ministry-defence-launches-cyber-security-training-centre, pristupljeno: 23.11.2019.

[27] Brilingaitė, Agnė, Linas Bukauskas, and Eduardas Kutka. "*Development of an Educational Platform for Cyber Defence Training*." European Conference on Cyber Warfare and Security. Academic Conferences International Limited, 2017.

[28] Eichler, Zdenek, Radek Ošlejšek, and Dalibor Toth. "*KYPO: A Tool for Collaborative Study of Cyberattacks in Safe Cloud Environment.*" International Conference on Human Aspects of Information Security, Privacy, and Trust. Springer, Cham, 2015.

[29] Virginia Cyber Range, *Customizing exercises*, https://kb.virginiacyberrange.org/features/customizing-exercise.html, pristupljeno: 06.01.2020.

[30] The Hague Security Delta, *National Cyber Testbed Programme*, https://www.thehaguesecuritydelta.com/projects/project/89-national-cyber-testbed, pristupljeno: 08.01.2020.

[31] Ogee, A., et al. "*The 2015 Report on National and International Cyber Security Exercises*." Technical Report. European Network and Information Security Agency, 2015.

[32] Seker, Ensar, and Hasan Huseyin Ozbenli. "*The concept of cyber defence exercises (cdx): Planning, execution, evaluation*." 2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security). IEEE, 2018.

[33] Kick, J. "*Cyber Exercise Playbook*, The MITRE Corporation, 2014." (2018).

[34] ENISA, *Good Practice Guide on National Exercises*, ENISA, 2009

[35] Locked Shields, Cyber Defence Exercise. "*After Action Report*." NATO CCD COE, Tallinn (2014)

[36] The Foreman, *Foreman*, https://theforeman.org/, 09.01.2020.

[37] Dragos Tanasache, Florin, et al. "*Building an Emulation Environment for Cyber Security Analyses of Complex Networked Systems*." arXiv preprint arXiv:1810.09752 (2018).

[38] Holm, Hannes, and Teodor Sommestad. "*Sved: Scanning, vulnerabilities, exploits and detection*."MILCOM 2016-2016 IEEE Military Communications Conference. IEEE, 2016.

[39] Mirkovic, Jelena, Genevieve Bartlett, and Jim Blythe. "*{DEW}: Distributed Experiment Workflows.*" 11th {USENIX} Workshop on Cyber Security Experimentation and Test ({CSET} 18). 2018.

[40] Srivats P., *Ostinato Packet generator*, https://ostinato.org/, pristupljeno: 11.01.2020.

[41] AT&T, *OSSIM – The Open Source SIEM*, https://cybersecurity.att.com/products/ossim, pristupljeno: 12.01.2020.

[42] Puppet, *Powerful infrastructure automation and delivery*, https://puppet.com/, pristupljeno: 14.01.2020.

[43] Amazon Web Services, *Cloud computing services*, https://aws.amazon.com/, pristupljeno: 14.01.2020.

[44] Openstack, *Build the future of Open infrastructure*, https://www.openstack.org/, pristupljeno: 15.01.2020.

[45] KVM, *KVM*, https://www.linux-kvm.org/page/Main_Page, pristupljeno: 15.01.2020.

[46] Oracle, *Oracle VM VirtualBox*, https://www.virtualbox.org/, pristupljeno: 15.01.2020.

[47] XEN, *Home – Xen Project*, https://xenproject.org/, pristupljeno: 15.01.2020.

[48] pfSense, *World's most trusted firewall*, https://www.pfsense.org/, pristupljeno: 15.01.2020.

[49] T. Capetta, https://github.com/secdevops-cuse/CyberRange, pristupljeno: 16.01.2020.

[50] University of Rhode Island, *OCCP*, https://opencyberchallenge.net/, pristupljeno: 16.01.2020.

[51] Wagner, Richard H. "*Designing a network defense scenario using the open cyber challenge platform*." (2013).

[52] Business insider, *Cyber attacks are the newest frontier of war and can strike harder than a natural disaster. Here's why the US could struggle to cope if it got hit*, 23.5.2019., https://www.businessinsider.com/cyber-attack-us-struggle-taken-offline-power-grid-2019-4, pristupljeno: 20.01.2020.

[53] NATO *Cooperative Cyber Defence Centre of Excellence, Crossed Swords,* https://ccdcoe.org/exercises/crossed-swords/ , pristupljeno: 20.01.2020.

[54] NATO, *NATO's flagship cyber exercise begins in Estonia*, https://www.nato.int/cps/ic/natohq/news_149233.htm, pristupljeno: 20.01.2020.

[55] ENISA, *Cyber Europe*, https://www.enisa.europa.eu/topics/cyber-exercises/cyber-europe-programme, pristupljeno: 20.01.2020.

[56] NATO *Cooperative Cyber Defence Centre of Excellence, Locked Shields, https://ccdcoe.org/exercises/locked-shields/*, pristupljeno: 20.01.2020.

Sažetak

Naslov rada: Izrada i ispitivanje kibernetičkog poligona za uvježbavanje tehničkih sposobnosti

Sažetak: U ovom radu je istraženo što su točno kibernetički poligoni, koje su njihove mogućnosti i kako se organiziraju kibernetičke vježbe na tim poligonima. Istraženi su alati otvorenog koda i predložena su rješenja pomoću kojih se mogu izraditi komponente za vlastiti kibernetički poligon i značajno smanjiti cijena izrade jednog takvog poligona. Na kraju je istraženo koji kibernetički poligoni otvorenog koda postoje i pokazano je da trenutačno nema poligona koji se mogu po funkcionalnostima usporediti s komercijalnim rješenjima.

Ključne riječi: Kibernetički poligon, kibernetička sigurnost, kibernetička vježba

Abstract

Title: Design and testing of a cyber range training ground for practicing technical skills

Summary: This thesis explores what are cyber ranges, what are their capabilities and how are cyber exercises organized at these cyber ranges. Open source tools have been explored and solutions have been proposed to build components of your own cyber range that significantly reduce the cost of producing one such cyber range. Finally, it was investigated which open source cyber ranges exists and it was shown that there are currently no cyber ranges that are comparable in functionality to commercial solutions.

Keywords: Cyber range, cyber security, cyber exercise