UNIVERSITÀ DEGLI STUDI DI MILANO

FACOLTÀ DI SCIENZE E TECNOLOGIE

DIPARTIMENTO DI INFORMATICA
GIOVANNI DEGLI ANTONI

Corso di Laurea Magistrale in
Sicurezza Informatica

APPROCCIO MODEL DRIVEN PER GENERARE CYBER RANGES

Relatore: Prof.ssa Chiara Braghin

Tesi di Laurea di:
Michele Toccagni
Matr. Nr. 921917

ANNO ACCADEMICO 2020-2021

Abstract

Negli ultimi anni abbiamo assistito a un aumento costante delle minacce cibernetiche, che, congiuntamente a una capacità di offesa sempre più sofisticata e avanzata, ha necessariamente forzato le organizzazioni governative e le aziende ad aggiornare personale e strumenti, al fine di aumentare anche la propria capacità di difesa.

Ciò ha aumentato la necessità di formazione continua tramite tecniche e scenari realistici utilizzando metodologie all'avanguardia, come i Cyber Ranges. Questi ultimi possono simulare un ambiente ad hoc per i professionisti della sicurezza informatica per poter replicare i più comuni attacchi informatici, per valutare la gestione degli incidenti e le possibili procedure di risposta a essi.

Essi sono particolarmente utili sia in ambito professionale che scolastico, in quanto possono riprodurre azioni altrimenti non riproducibili in sistemi di produzione.

L'obiettivo di questa tesi è identificare una modalità di generazione di una piattaforma Cyber Range con un approccio model driven e che possa essere gestita in automatico, senza necessità di particolari configurazioni e che possa essere utilizzata sia in ambito accademico sia in ambito lavorativo.

Ringraziamenti

Desidero innanzitutto ringraziare la prof.ssa Chiara Braghin, relatrice di questa tesi, che ha creduto nelle mie capacità dandomi ampia libertà di scelta ed indipendenza, permettendomi di conoscere e approfondire nuove tematiche inerenti alla cybersecurity.

Ringrazio i miei genitori per il sostegno e la fiducia datomi nel corso degli anni e le mie sorelle, Eva e Sofia, per avermi dato supporto durante il mio percorso.

Un ringraziamento particolare va a Giulia, che, con infinita pazienza, mi ha supportato e sopportato durante le sessioni, le giornate e serate di studio, nei momenti di difficoltà e non solo, dandomi supporto morale costante.

Vorrei ringraziare tutti gli amici e colleghi, con i quali ho condiviso gran parte del mio percorso accademico, per avermi aiutato nel raggiungimento di questo traguardo.

Infine, un grazie speciale va a Ugo, compagno fedele nelle lunghe serate di studio.

Indice

Abstract

Ringraziamenti

Indice

1 Introduzione **1**
1.1 Minacce più comuni 1
1.2 L'importanza del training 2
1.3 Contributi della tesi 5
1.4 Organizzazione della tesi 5

2 Cyber ranges **7**
2.1 Cosa sono 7
2.2 A chi sono rivolti 12
2.2.1 Organizzazioni 13
2.2.2 Educatori 14
2.2.3 Singoli individui 16
2.3 Tecnologie e funzionalità 17

3 Stato dell'arte dei Cyber Ranges **21**
3.1 Panoramica delle ricerche accademiche 21
3.2 Principali Piattaforme 26
3.2.1 Virgina Cyber Range 26
3.2.2 Cisco Cyber Range 26
3.2.3 IBM X-Force 27
3.2.4 NATO Cyber Range 28
3.2.5 CRATE 28
3.2.6 KYPO 29

4 Generazione model driven di scenari **31**
4.1 Piattaforma THREAT ARREST 31
4.1.1 Architettura della piattaforma 32
4.1.2 Descrizione della piattaforma 32
4.1.3 Sviluppo Model Driven di un DSL 36
4.1.4 Cybersecurity Training Language 39
4.2 Contesto di partenza 41
4.2.1 OWASP Top 10 41
4.2.2 Capture the Flag 44
4.2.3 Juice Shop 49
4.3 Generazione Model Driven di scenari 57
4.3.1 Suddivisione in categorie 57
4.3.2 Creazione di più istanze ed esecuzione contemporanea 57

5 Conclusioni **65**
5.1 Sviluppi futuri 65

Bibliografia **68**

Capitolo 1

Introduzione

Il 2020 è stato un anno di cambiamenti significativi nel panorama delle minacce informatiche, sia per un'epidemia che ha colpito in maniera globale[1], sia per l'aumento sostanziale di attacchi informatici avanzati. Tali avvenimenti hanno sicuramente inasprito l'impatto che la pandemia ha avuto sul telelavoro.

I cambiamenti imposti ai lavoratori di tutto il mondo hanno reso necessaria l'adozione di misure ad hoc che permettessero un aumento delle misure informatiche di difesa esistenti, cercando di ridurre al minimo l'esposizione ad attacchi informatici. Tale esposizione è risultata essere aggravata dalla mancanza di reti protette nelle case dei dipendenti, rete che in azienda risulterebbe essere più controllata, o dall'utilizzo di dispositivi intelligenti e altri oggetti connessi a internet.

Questo scenario ha imposto un rapido cambiamento dei professionisti della sicurezza informatica, i quali hanno dovuto adattare le misure di sicurezza presenti precedentemente al telelavoro all'esposizione a rischi prima non valutati, trasformando la difesa dei perimetri aziendali in vere e proprie sfide.

1.1 Minacce più comuni

Secondo ENISA [1], il modus operandi degli attaccanti si è concentrato su una diversificazione dei vettori di attacco, con metodi avanzati per sottrarre credenziali, attacchi di phishing mirati e attacchi avanzati di Ingegneria Sociale.

Tutto ciò ha fatto in modo che, tra le principali minacce alle infrastrutture, oltre alla conferma in maniera preponderante di malware e attacchi sulle applicazioni web, sia scalato in terza posizione il phishing, minaccia da tenere strettamente sotto controllo in quanto sfrutta principalmente la debolezza umana.

Le minacce più comuni sono:

[1] `https://covid19.who.int/`.

1. I **Malware**, che si posizionano come il più comune attacco informatico, siano essi virus, ransomware o spyware, i cui obiettivi sono il furto di informazioni e l'interruzione del servizio. I vettori di attacco sono principalmente la posta elettronica e i protocolli web;

2. Gli **attacchi basati sul web**, i quali utilizzano script o url malevoli per indirizzare l'utente verso un sito compromesso per sottrarre informazioni. Fanno parte di questa categoria anche gli exploit dei browser e la conseguente compromissione del computer dell'utente;

3. Il **phishing**, che è il tentativo di truffa nel quale un malintenzionato cerca di ingannare l'utente tramite tecniche di ingegneria sociale, che, fingendosi un ente affidabile, cerca di convincerlo a seguire un link fraudolento o ad aprire un allegato dannoso;

4. Le **applicazioni web**, che rappresentano uno dei principali veicoli di comunicazione e contenitori di grandi basi di dati con informazioni sensibili, di conseguenza uno dei principali target per un attaccante. Essi possono essere di diverse tipologie, come attacchi di tipo SQL Injection, Cross Site Scripting o Remote Command Execution.

1.2 L'importanza del training

Il numero di data breach è sempre molto alto, e questo ha diverse cause, che spaziano dal non applicare una patch di sicurezza a una vulnerabilità nota, al non monitorare adeguatamente la propria infrastruttura, fino al non insegnare ai propri dipendenti certi comportamenti da evitare.

Solo nell'ultimo anno, sono state identificate decine, se non centinaia di violazioni di sicurezza e tra le più rilevanti troviamo:

- **Solawinds**: è stato uno degli attacchi hacker più sofisticato di sempre, nel quale è stato compromesso il sistema di compilazione della società di software Solarwinds.
  Una volta avuto accesso a esso gli attaccanti hanno modificato il codice sorgente e inviato gli aggiornamenti malevoli agli utenti del software, tra i quali troviamo l'azienda di Sicurezza Informatica FireEye, il governo degli Stati Uniti e molti altri. Quando veniva aggiornato, il software diventava un trojan, cifrando i dati sensibili e inviandoli ai server degli attaccanti; [2]

- **Twitter**: gli autori dell'attacco hanno utilizzato la tecnica del phishing per ottenere alcune credenziali dei dipendenti di Twitter, con le quali hanno

Top Threats 2019-2020		Assessed Trends	Change in Ranking
1	Malware	---	---
2	Web-based Attacks	---	↗
3	Phishing	↗	↗
4	Web application attacks	---	↙
5	Spam	↙	↗
6	Denial of service	↙	↙
7	Identity theft	↗	↗
8	Data breaches	---	---
9	Insider threat	↗	---
10	Botnets	↙	↙
11	Physical manipulation, damage, theft and loss	---	↙
12	Information leakage	↗	↙
13	Ransomware	↗	↗
14	Cyberespionage	↙	↗
15	Crytojacking	↙	↙

Legend: Trends: ↙ Declining, --- Stable, ↗ Increasing **Ranking:** ↗ Going up, --- Same, ↙ Going down

Figura 1: Principali minacce del 2020 secondo ENISA [1].

avuto accesso agli strumenti di supporto interno e preso successivamente il controllo di molteplici account Twitter; [3]

- **Zoom**: con una crescita esplosiva avvenuta durante la pandemia, Zoom ha dovuto affrontare diversi incidenti di sicurezza, uno dei quali ha fatto emergere centinaia di migliaia di account in vendita nel dark web, probabilmente ottenuti in precedenti violazioni e poi sfruttati per trarne profitto. [4]

Viste le frequenti minacce in costante aumento, è necessario identificare delle best practice che permettano ai professionisti della sicurezza informatica di essere in grado di reagire a situazioni critiche, sia in ambito offensive, quindi nell'identificare eventuali falle nei sistemi, sia in ambito defensive, nel quale devono essere messe in atto tutte quelle contromisure che permettono di evitare eventuali data breach.

Per questo motivo è necessario che l'istruzione e la formazione in questa materia diventi fondamentale e che debbano essere applicati nuovi metodi di insegnamento che si evolvono costantemente, uniti a esercizi pratici.

L'istruzione alla sicurezza informatica deve iniziare a scuola, in modo tale che, oltre a indurre i giovani a percorrere questa carriera, permetta loro di sviluppare le conoscenze e le abilità richieste nelle situazioni di cui sopra, tramite non solo classiche lezioni frontali ma anche e soprattutto con l'esercitazione pratica.

Lo scopo dell'addestramento tramite esercizi che simulano ambienti reali fa in modo che si possa analizzare il comportamento di un individuo e di come potrebbe reagire a situazioni di difficoltà, in maniera molto simile a una reale situazione lavorativa.

Le esercitazioni pratiche possono essere realizzate in vari modi, con macchine virtuali e piattaforme web che possono simulare varie caratteristiche della rete, come lo possono essere i Cyber Ranges.

1.3 Contributi della tesi

Il lavoro svolto si è concentrato sull'identificazione di una piattaforma libera, aggiornata e adatta a essere inserita in un modello che sia liberamente fruibile e utilizzabile da chiunque, senza necessità di particolari competenze e costi.

La scelta è ricaduta su Juice Shop, una delle più moderne e sofisticate applicazioni web sviluppata appositamente in maniera insicura per permetterne l'utilizzo in corsi di formazione. Sebbene come base di partenza Juice Shop fosse l'ideale, è stato necessario affrontare una serie di criticità per adattarla in modo tale da ottenere un modello di base per la generazione di Cyber Range.

Innanzitutto, l'applicazione è stata modificata così da avere a disposizione diversi scenari di attacco (categorizzati sulla base della OWASP Top 10) e permettere a un trainer di eseguirlo sulla base della categoria prescelta.

A seguire si è deciso di effettuare un'integrazione con CTFD, un framework CTF che permette di avere una piattaforma di scoring al fine di unire la gamification di una Capture The Flag insieme all'analisi, da parte del trainer, dei punteggi ottenuti in tempo reale dei trainee.

In questo modo, si è potuto configurare un deploy automatico, contemporaneo e multiplo di più istanze di Juice Shop basandosi sul numero di trainee, con un'istanza gestita dal trainer, attraverso la quale possa monitorare il livello dei trainee, i risultati raggiunti e identificare possibili problemi durante l'esercitazione.

1.4 Organizzazione della tesi

La tesi è organizzata come segue:

- Capitolo 1: Introduzione, panoramica degli attacchi informatici e principali minacce per le aziende;

- Capitolo 2: Definizione Cyber Range, tipologie esistenti e principali stakeholder con i conseguenti vantaggi inerenti all'utilizzo di CR. Successivamente si è svolta un'analisi delle tecnologie e funzionalità più ricorrenti in modo da identificare la strategia da utilizzare per il progetto;

- Capitolo 3: Analisi delle più recenti ricerche accademiche a livello bibliografico e descrizione delle piattaforme più utilizzate, sia in ambito accademico che aziendale;

- Capitolo 4: Descrizione di THREAT ARREST, piattaforma utilizzante un approccio Model Driven e scelta come possibile futura implementazione del lavoro svolto. Si è poi passati alla definizione delle vulnerabilità definite in

OWASP Top 10, all'analisi delle tipologie di CTF ed infine si è descritto il progetto creato con l'integrazione tra Juice Shop e CTFd.

Capitolo 2

Cyber ranges

2.1 Cosa sono

Un Cyber Range (CR) è un ambiente dove i professionisti della sicurezza informatica possono esercitarsi e dimostrare le proprie competenze in scenari di complessità reale.

Solitamente essi contengono l'intera infrastruttura necessaria, in modo che si possano eseguire attività di formazione e valutazione per rafforzare le proprie competenze e partecipare a scenari di cyber attacchi o difesa simulati. Le piattaforme possono comprendere la prevenzione, il rilevamento, la risposta o l'analisi di attacchi informatici al fine di testare nuove tecnologie.

L'European Cyber Security Organisation (ECSO) definisce i Cyber Ranges come:

> Un Cyber Range è una piattaforma per lo sviluppo, la realizzazione e l'uso di ambienti di simulazione interattivi. Un ambiente di simulazione è una rappresentazione di sistemi fisici, mobili, ICT e OT di un'organizzazione, delle applicazioni e delle infrastrutture, compresa la simulazione degli attacchi, degli utenti e delle loro attività e di qualsiasi altro servizio internet, pubblico o di terze parti da cui l'ambiente può dipendere. Un Cyber Range include una combinazione di tecnologie di base per la realizzazione e l'uso dell'ambiente di simulazione e di componenti aggiuntivi che sono, a loro volta, richiesti per aggiungere specifici casi d'uso (ECSO, 2020) [5]

L'uso dei CR deriva dalla necessità di avere ambienti protetti e sicuri dove svolgere attività di Cyber Security, isolati dal mondo esterno e senza la necessità di operare direttamente sulle reti o infrastrutture aziendali.

In base al tipo di infrastruttura si possono identificare tre tipi di Cyber Range [6]:

- **Fisico**: viene ricreata fedelmente una rete, spesso utilizzando componenti reali dell'infrastruttura di riferimento. Da una parte, è la soluzione migliore dal punto di vista della somiglianza con la rete originale, in quanto è una mera copia fisica, ma ha lo svantaggio di essere meno flessibile e prona a eventuali modifiche, che comportano ingenti costi e manutenzione sul posto. Un esempio di questa tipologia è CyberTropolis, l'ambiente di addestramento cyber fisico del Muscatatuck Urban Training Center. Esso è composto da autentici sistemi di controllo industriale (ICS), sistemi di acquisizione dati (SCADA), controllori logici programmabili (PLC) e una rete cellulare integrata che spazia dal 2G al 5G[1];

- **Virtuale**: in questo caso vengono utilizzate tecnologie di virtualizzazione che permettono di simulare i componenti dell'infrastruttura con il vantaggio di poter ricreare l'ambiente a costi minori e con una semplicità maggiore. Lo svantaggio è che non sarà sempre possibile riuscire a ricreare la stessa architettura voluta.
  Per esempio, il CR KYPO, sviluppato dall'Università di Masaryk, è interamente basato su approcci come container, infrastrutture, microservizi e software open-source, compresa la tecnologia del cloud provider, basata su OpenStack. Le macchine virtuali, le reti e gli addestramenti sono definiti in linguaggi leggibili dall'uomo, come JSON e YAML, e usano il software open-source Packer per creare le macchine virtuali e Ansible per descrivere il contenuto delle macchine[2];

- **Ibrido**: quest'ultima tipologia è un ibrido delle due precedenti e mira a combinare gli aspetti positivi di entrambi gli approcci, con la flessibilità di un ambiente virtuale e il realismo dato dalle componenti hardware reali.
  Un CR di questo tipo è EVA (Emulatore di Vero Acquedotto), che rappresenta un modello funzionante di un sistema di approvvigionamento idrico composto da componenti base come serbatoi, tubi e pompe[3].

In maniera indipendente dal tipo di infrastruttura, i CR possono essere utilizzati sia in maniera individuale sia in squadre.

Nel primo caso possiamo trovare i CTF (Capture The Flag), nei quali i trainee hanno un obiettivo da raggiungere (la flag, ossia una stringa che verifica che l'obiettivo è stato raggiunto) e solitamente il punteggio viene dato in base alla difficoltà dell'esercizio.

I CTF riescono a collimare il divario tra le sessioni teoriche e la parte pratica, in quanto permettono di sperimentare in prima persona i concetti appresi.

[1] `https://www.atterburymuscatatuck.in.ng.mil/Muscatatuck/CyberTropolis/`.
[2] `https://crp.kypo.muni.cz/`.
[3] `http://ceur-ws.org/Vol-2597/paper-02.pdf`.

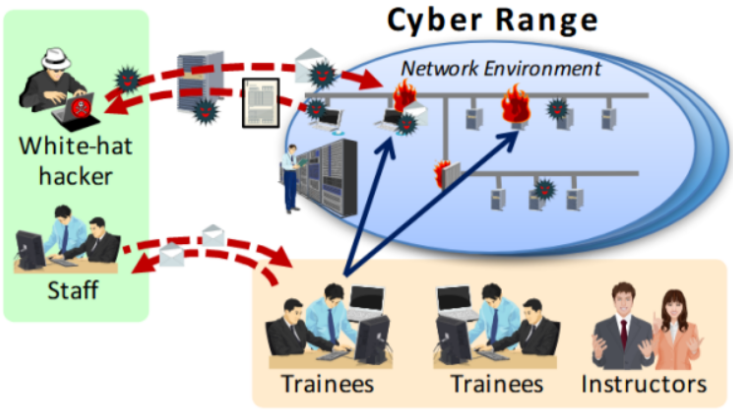


Figura 2: Cyber Range esemplificato

Ci sono due tipologie principali di CTF [7]:

1. **Task-based**: l'attività si riduce a risolvere quanti più problemi possibili in diverse aree (digital forensics, applicazioni web, crittografia, sicurezza mobile);

2. **Attacco-difesa**: l'addestramento si occupa alla protezione della rete, del server, delle informazioni riservate e al mantenimento della funzionalità dei servizi nello stesso momento in cui avviene l'implementazione di attacchi mirati a violare gli stessi dalla squadra nemica.

Nel caso di un CTF formato da squadre, una delle attività più formative e che richiede sicuramente più esperienza è quella in cui vengono contrapposti gli attaccanti e i difensori, i cosiddetti Red Team e Blue Team.

- **Red Team** è la squadra che simula gli attaccanti cercando di ottenere un accesso non autorizzato al sistema. Solitamente l'obiettivo è quello di prendere il controllo dell'infrastruttura senza essere identificato o bloccato dall'altra squadra;

- **Blue Team** è il team responsabile della difesa e della protezione degli ambienti. Essi cercano di rilevare e rispondere agli attacchi, difendendo e proteggendo l'infrastruttura.

Uno degli eventi principali di questo tipo è Locked Shield, nel quale un singolo red team è sfidato da molteplici blue team [4].

[4] `https://ccdcoe.org/exercises/locked-shields/`.

Un CR permette quindi sia al singolo individuo di applicare i concetti teorici appresi, sia ai professionisti di migliorare le proprie capacità in un ambiente interattivo che simula una situazione di pericolo reale, minimizzando l'impatto sui sistemi in produzione.

Un altro aspetto molto importante da tenere in considerazione nello sviluppo di un CR è la gamification. Secondo Huotari et. al. [8], la gamification è definita come:

> Un processo di miglioramento di un servizio con caratteristiche che permettono esperienze di gioco al fine di sostenere la creazione di valore aggiunto per l'utente.

Core service	**Enhancing service**	**Gamified service**
Profile in LinkedIn	Progress bar for measuring progress in filling personal details	The enhancing service increases the perceived value of filling all details by invoking progress-related psychological biases.
Café	Mayorship competition in Foursquare	The enhancing service creates a competition between customers where they have to visit the café frequently enough -> retention
Dry cleaner	Loyalty stamp card. You get 1 stamp for every visit	The enhancing service invokes the psychological biases related to progress and thus increases the perceived value of using the same dry cleaner service.
Gym	Heya Heya	Gym experience that sets goals and helps to monitor the progress of the training.

Figura 3: Esempi di Gamification

Esempi di gamification, come in Figura 3 [8], si possono trovare in servizi come Linkedin, dove viene visualizzato un indicatore di progresso del completamento del profilo, in modo da incentivare l'utente a inserire quante più informazioni possibili.

La gamification applicata alla teoria e alla meccanica dei giochi, come l'uso di premi e ricompense che si possono vedere nei CTF, aumenta di fatto il coinvolgimento e la motivazione degli utenti, in un contesto che è tradizionalmente non ludico.

Uno dei fiorenti campi di applicazione della gamification negli ultimi anni è appunto l'educazione, spostando la prospettiva del procedimento di apprendimento classico da passivo a attivo, trasformando di fatto gli studenti in protagonisti attivi del processo, aiutandoli a raggiungere obiettivi specifici. La motivazione è un elemento essenziale nel processo di apprendimento e riuscire a inserirlo in un campo dove è difficile trovare una parte ludica può essere un notevole vantaggio.

In una ricerca condotta nel 2020 da Vykopal et al. [9], un gruppo di studenti ha frequentato un corso introduttivo sulla sicurezza delle informazioni, prendendo parte, in piccoli gruppo o singolarmente, a delle sessioni di CTF. Al termine di tali sessioni, con la somministrazione di questionari e sondaggi, sono state conglobate le statistiche di gioco, da cui è emerso che la sostituzione dei tradizionali esercizi a favore dei CTF risulta essere ottimale e favorevole sia per gli istruttori che per gli studenti.

Per quanto riguarda i primi, c'è sicuramente un risparmio di tempo rispetto che correggere i classici esercizi, ma sono gli studenti che ne hanno giovato, in quanto i CTF forniscono più opportunità e un ambiente meno stressante per esplorare e imparare l'argomento e in contemporanea, avere un feedback istantaneo sulla correttezza di una soluzione permette di identificare gli studenti a rischio che non riescono a risolvere un determinato esercizio.

I vantaggi principali sono:

1. Determinare l'efficacia della formazione fornita al singolo;
2. Valutare la capacità di difesa del team preposto (a esempio il Security Operation Center, SOC) in modo da identificare eventuali punti di debolezza;
3. Valutare la capacità di attacco del team e permettere un aggiornamento costante delle competenze;
4. Migliorare la consapevolezza e la coordinazione di entrambi i team.

Il modello generale di un Cyber Range è composto dai seguenti livelli [10] (come schematizzato in Figura 4):

1. Livello di Orchestrazione: è il livello che contiene la tecnologia dei componenti del Cyber Range e li integra tra di loro. Alcuni sviluppano il proprio Orchestration layer in autonomia, di solito con software come OpenStack[5], mentre altri utilizzano prodotti commerciali, come CloudShell[6];
2. Livello di Infrastruttura: è l'infrastruttura che compone il CR, composta quindi da reti, server e storage. Alcuni sono costruiti direttamente su infrastrutture fisiche, mentre altri utilizzano quelle virtuali;

[5] `https://www.openstack.org/`.
[6] `https://cloud.google.com/shell`.

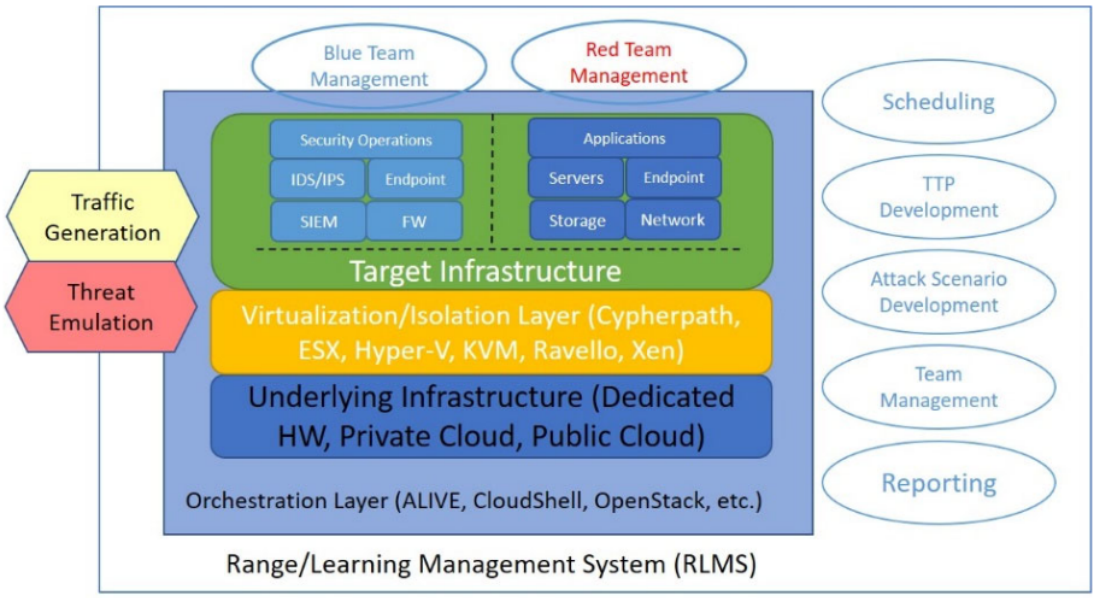


Figura 4: Modello generico di un Cyber Range.

3. Livello di Virtualizzazione: anche qui ci sono due possibili approcci: soluzioni basati su Hypervisor o software definiti sulla'infrastruttura.
   Nel primo caso il livello di divisione tra l'infrastruttura fisica sottostante e quella di destinazione influisce sul realismo del CR a causa della possibile latenza ma, d'altra parte, i CR economicamente sostenibili non sarebbero possibili senza l'utilizzo della virtualizzazione;

4. Infrastruttura target: è l'ambiente emulato nel quale il trainee si esercita. Il Range/Learn Management System genera degli script che permetteranno al livello di orchestrazione di creare l'infrastruttura target, inclusi gli indirizzi IP, le informazioni di route e il software endpoint.

2.2 A chi sono rivolti

I Cyber Ranges possono essere utilizzati da una vasta gamma di utenti ma prima è necessario identificare lo scopo e gli obiettivi da raggiungere al fine da ottenere i benefici maggiori.

Secondo un recente sondaggio [11], da cui è tratto il grafico di Figura 5, l'obiettivo principale è la formazione. La maggior parte dei partecipanti ha utilizzato i sistemi CR per la ricerca, la formazione e per esercitarsi.

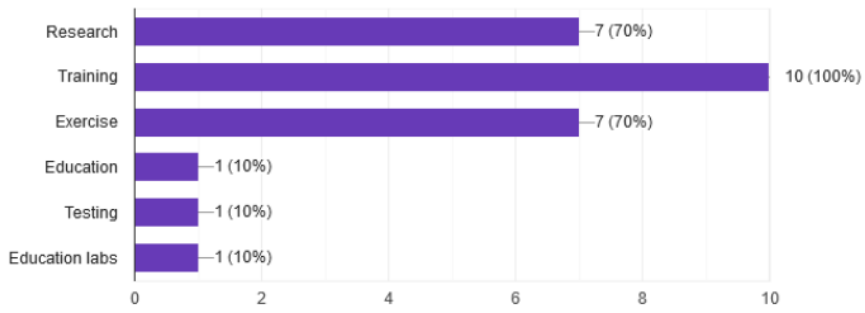


Figura 5: Obiettivi nell'utilizzo di un CR

2.2.1 Organizzazioni

Un'organizzazione, sia pubblica che privata, ha tre possibili applicazioni dei Cyber Ranges:

- Valutare la propria resilienza informatica;
- Formare i propri dipendenti;
- Identificare nuove figure da inserire nel proprio organico.

Secondo il NIST[7] la resilienza informatica è definita come "la capacità di anticipare, affrontare, recuperare e adattarsi a condizioni avverse, stress, attacchi o compromissioni di sistemi che includono risorse informatiche", e proprio in questo contesto, le esercitazioni informatiche permettono alle organizzazioni di dimostrare le capacità critiche e rilevare quanto efficacemente riescano a integrare persone, processi e tecnologie per proteggere i loro sistemi e risorse critiche.

La progettazione delle sfide si può basare su diversi approcci e seguendo le priorità necessarie, come sviluppare le competenze di sicurezza informatica, misurare le capacità dei team preposti all'attacco e alla difesa o aumentare la consapevolezza di aziende e dirigenti.

In un organizzazione, un Cyber Range può essere progettato per implementare le sfide con l'obiettivo che i partecipanti possano sviluppare, migliorare le loro capacità o costruirne di nuove sulla base delle esperienze acquisite sulle loro esperienze di Cyber Security. Nel caso della formazione ai propri dipendenti è possibile utilizzarlo per affiancare le lezioni teoriche con quelle pratiche.

Le organizzazioni utilizzano principalmente corsi online e in presenza per il training dei propri dipendenti, senza includere però sessioni di esercitazione che

[7] `https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-160v2.pdf`.

Table 6. Operational Areas that Account for Security Spending

Operational Area	% Response
Protection and prevention	72.4%
Detection and response	62.8%
Compliance and audit (including legal)	58.6%
Risk reduction	49.7%
End user training and awareness	45.5%
Governance/Policies	43.4%
Staff training and certification	39.3%
Security program or project management	38.6%
Design/Development	34.5%
Discovery and forensics	31.7%
Other	2.1%

Figura 6: Nel 2015 il 39% del budget delle aziende è stato usato per il training [12]

permetterebbero il mettere in pratica le nozioni acquisite. Sono invece necessarie attività di formazione e ambienti che possano sostenere situazioni impegnative, seguite da linee guida concrete, procedure e strumenti.

Queste piattaforme possono aiutare i dipendenti a reagire in situazioni diverse e imprevedibili in modo collettivo e collaborativo.

I CR dovrebbero fondere simulazioni ed emulazioni di componenti e sistemi reali, incorporando diversi meccanismi di attacco e difesa ed essere in grado di adattarsi a una varietà di incidenti diversi, al fine di essere attraenti per le organizzazioni e gli istituti di istruzione.

Per quanto riguarda l'identificazione di nuove figure, l'utilizzo dei CR potrebbe aiutare l'organizzazione durante la selezione di nuovi possibili dipendenti durante le fasi di selezione. Solitamente i colloqui vengono fatti con domande teoriche e senza testare le capacità del candidato, principalmente a causa delle tempistiche minori che richiede un colloquio e i costi necessati per implementare un CR.

Se si riuscisse a inserire nel percorso di recruiting anche una sessione pratica, includendo un CR sia singolo sia in team, l'azienda potrebbe valutare il candidato a 360° e identificare le persone più adatte per il ruolo cercato.

2.2.2 Educatori

I corsi universitari difficilmente inseriscono esercitazioni pratiche, principalmente a causa del costo e della complessità di fornire una piattaforma ad hoc. Una delle

lamentele ricorrenti dell'industria è infatti la mancanza di esperienza pratica da parte dei giovani laureati.

La causa principale di tale lacuna è il costo e la complessità di fornire agli studenti un'esperienza pratica e allo stesso tempo non diluire il valore educativo dei diplomi universitari.

Le università di tutto il mondo hanno iniziato a guardare ai Cyber Ranges come un mezzo per migliorare l'insegnamento e l'apprendimento. Un esempio lo è il Virginia Cyber Range[8], utilizzato da più di 220 istituti scolastici[9].

Non è inoltre da sottovalutare l'aspetto pedagogico della esercitazione pratica, definito a grandi linee in Figura 7, e i benefit che ne trae lo studente. Le abilità sono spesso costruite con l'insegnamento di tipo didattico nel senso dell'apprendimento comportamentale, dove l'importanza del materiale didattico dell'insegnante è centrale.

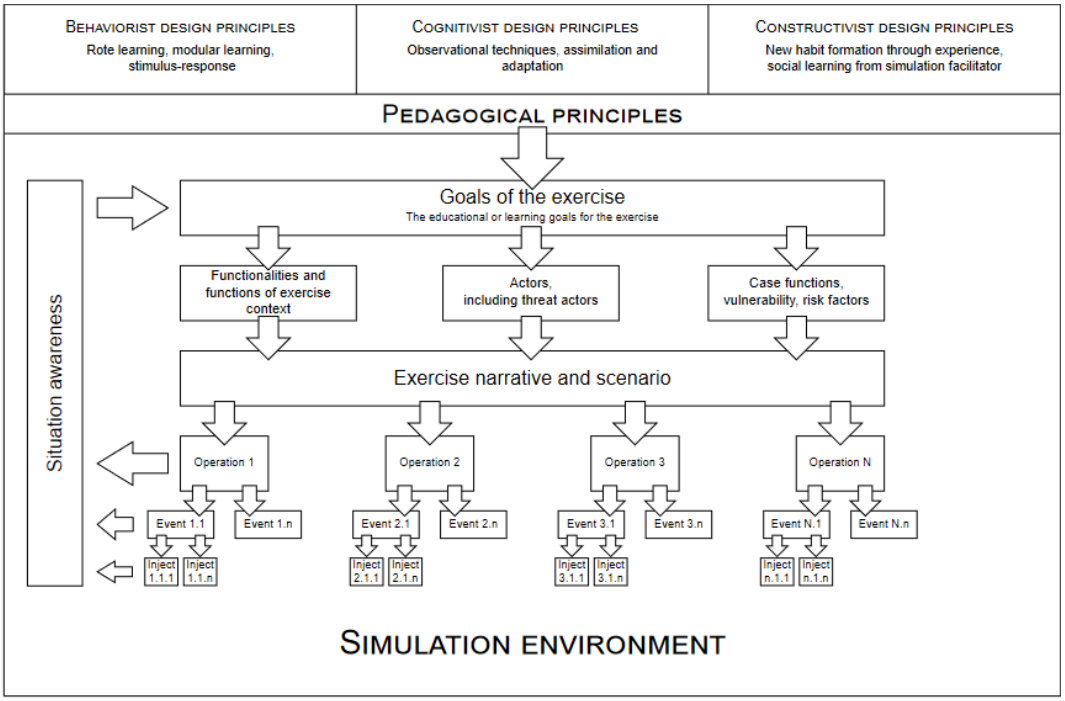


Figura 7: Ciclo di apprendimento di un esercizio

Ma quando la capacità di conoscenza di uno studente cresce, la sua elaborazione cognitiva dei dati aumenta, e lo studente lega le informazioni che ha imparato in precedenza.

[8] https://www.virginiacyberrange.org/.

[9] https://edtechmagazine.com/higher/article/2019/10/university-cyber-ranges-immerse-students-cybersecurity-education-perfcon.

L'apprendimento esperienziale è una tecnica educativa che propone il coinvolgimento attivo dei partecipanti per aiutarli a imparare attraverso l'esperienza, un metodo efficiente per fornire esercizi di apprendimento esperienziale come parte di giochi.

L'apprendimento basato sui problemi spesso inizia direttamente con la risoluzione di un problema reale e l'esplorazione delle informazioni di base a esso collegate. L'idea alla base è di reagire all'apprendimento come un ricercatore, di porre domande significative e interessanti dell'argomento da imparare e poi risolverle. Ciò significa che l'insegnamento basato sui problemi assomiglia alla ricerca scientifica. [13]

2.2.3 Singoli individui

Singoli individui o ricercatori utilizzano i CR per formarsi in maniera verticale in determinati contesti e per imparare nuove tecniche di attacco o difesa. Questo perché difficilmente è possibile testare nuovi attacchi e vulnerabilità senza l'ausilio di una ambiente virtualizzato creato appositamente. I CR sono un mezzo fondamentale per condurre ricerche sulla sicurezza nei vari domini di sicurezza.

Per loro stessa natura, esse sono sviluppate da ricercatori di tutto il mondo per cercare nuovi metodi di rilevamento e mitigazione degli attacchi, emulazione di malware e molto altro.

Per quanto riguarda gli studenti, essi non sempre imparano come individuare e correggere i difetti di progettazione, configurazione e applicazione nei sistemi esistenti ed esercizi come i CTF possono insegnare queste abilità.

Oltre alla possibilità di percorrere i primi passi nella sicurezza informatica, i CR trovano ampio spazio anche nella ricerca, in quanto danno la possibilità di creare un ambiente virtuale o fisico per trovare vulnerabilità non conosciute, dette 0-day. La possibilità di avere un ambiente ad hoc al fine di cercare nuove vulnerabilità all'interno di prodotti utilizzati quotidianamente dalle persone permette una rapido sviluppo e interesse nella materia.

Unendo quest'ultimo punto e facendolo confluire in un CTF, possiamo vedere una della competizioni più famose degli ultimi anni, nella quale si uniscono ricercatori di tutto il mondo in una sfida al fine di trovare le vulnerabilità più gravi nei prodotti (software o hardware) più utilizzati.

Pwn2Own[10] è un concorso di hacking informatico che si tiene annualmente alla conferenza sulla sicurezza CanSecWest dove i concorrenti sono sfidati a sfruttare software e dispositivi mobili ampiamente utilizzati con vulnerabilità precedentemente sconosciute. Il concorso Pwn2Own serve a identificare le vulnerabilità dei

[10] `https://www.secwest.net/pwn2own`.

dispositivi e del software in uso diffuso, fornendo anche un checkpoint sui progressi compiuti nella sicurezza dall'anno precedente.

Solo nell'ultima sfida tenuta nel 2021, sono stati trovati 23 0-day in software utilizzati da milioni di persone, come Zoom e Microsoft Teams[11].

Questo evento differisce da un CTF classico perché è caratterizzato da solo attacchi e il suo obiettivo è quello di aiutare i venditori a trovare le vulnerabilità 0-day e non quello di aiutare ad educare i partecipanti. Ciò non toglie il fatto che l'unire la gamification all'interno di un evento atto a identificare vulnerabilità in prodotti commerciali porti benefici ai partecipanti.

2.3 Tecnologie e funzionalità

Indipendentemente dal fatto che un Cyber Range utilizzi una virtualizzazione (con hypervisor o container) o sia situata in un cloud (pubblico o privato), essa solitamente prevede determinate funzionalità in comune con altri CR, per la gestione di utenti, per la gestione dei punteggi, l'amministrazione della piattaforma e via discorrendo.

Le funzionalità più ricorrenti sono [5]:

1. Orchestrazione: si riferisce alla tecnologia responsabile alla creazione dei flussi di lavoro, alla configurazione automatica e al coordinamento delle varie risorse, che possono includere la creazione, modifica e cancellazione delle macchine virtuali o l'inserimento automatico delle valutazioni in sede di esercitazione.

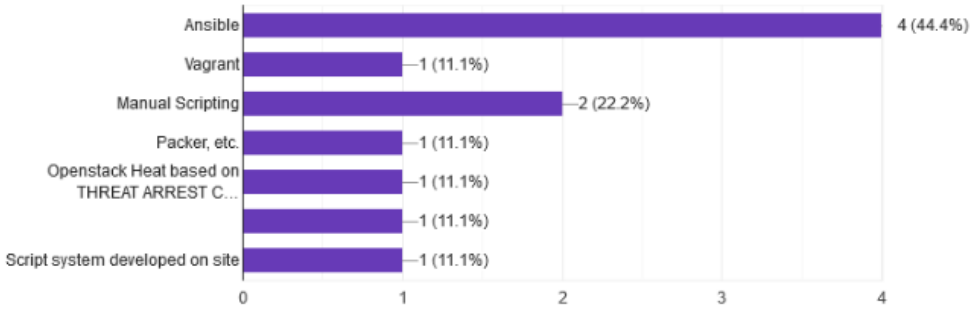


Figura 8: Principali Orchestratori utilizzati [11]

Essa può essere utilizzata anche per automatizzare certi compiti e le interazioni tra vari componenti, come la programmazione di attacchi informatici

[11] https://portswigger.net/daily-swig/pwn2own-2021-zero-click-zoom-exploit-among-winners-as-payout-record-smashed.

o la raccolta delle attività degli utenti. Poiché l'utilizzo dell'orchestrazione può influenzare notevolmente l'usabilità e il costo di un CR, ne è consigliato l'uso nel caso di molteplici macchine virtuali, in quanto senza di essa il coordinamento richiederebbe un enorme sforzo manuale. In Figura 8 sono mostrati i principali orchestratori utilizzati secondo il sondaggio di Chouliaras et. al. [11];

2. Simulazione di servizi intenet: in questa categoria troviamo la simulazione di qualsiasi servizio internet, come protocolli di routing, servizi DNS, social media e altri servizi. La simulazione di essi permette di aggiungere del realismo all'interno degli scenari, necessario nel caso di CR realistici che vogliono simulare Internet e i suoi servizi in maniera completa. In molto casi non vengono simulati a causa della complessità aggiuntiva richiesta;

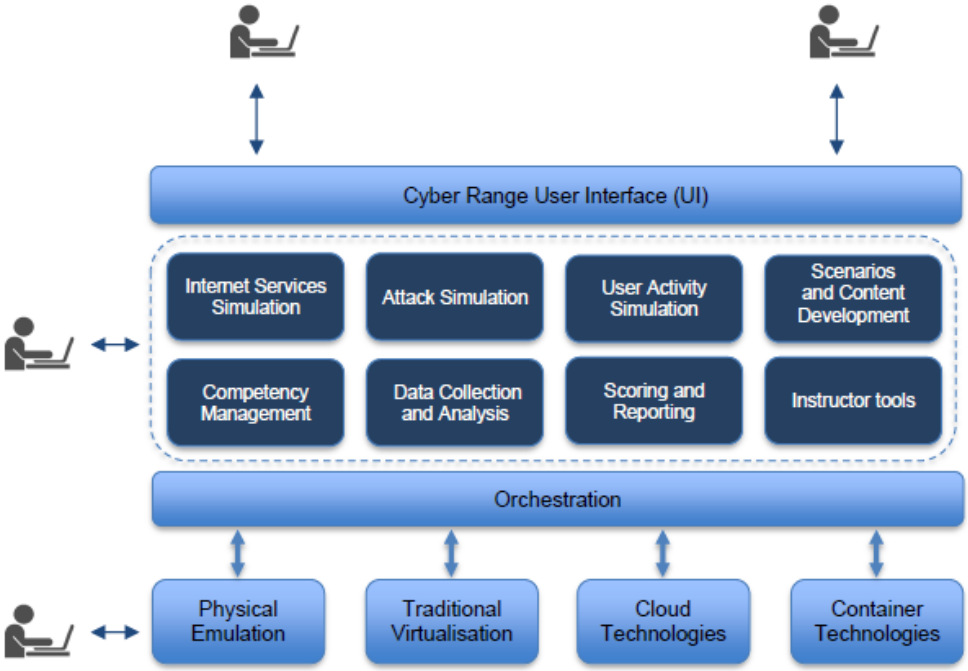


Figura 9: Ciclo di apprendimento di un esercizio

3. Simulazione di attacchi: si riferisce alla capacità di simulare attacchi all'interno dell'ambiente simulato e l'obiettivo principale è il effettuare test semi-automatici.

Mentre i programmi commerciali di scansione delle vulnerabilità si concentrano sull'identificazione di vulnerabilità nei sistemi, gli strumenti di simulazione degli attacchi consentono di simulare le diverse fasi di un attacco, fornendo raccomandazioni su come migliorare la sicurezza dell'infrastruttura. Sebbene il campo sia ancora in via di sviluppo, la simulazione di attacchi si sta concentrando sulle tattiche definite nel framework MITRE ATT&CK™[12], come con SVED [14];

4. Simulazione attività utente: simulando la presenza e il comportamento di utenti all'interno del CR permette di rendere l'ambiente ancora più realistico, sia essi siano utenti esterni che interni. La simulazione include, oltre alla classica navigazione internet, anche l'utilizzo di telefoni cellulari e altre tecnologie presenti nel sistema;

5. Scenario: l'utilità di un CR è correlata anche allo scenario nella quale viene collocata. L'idea è paragonabile all'esperienza che avviene con l'utilizzo di una gioco per computer, il quale è composto da uno scenario principale ma anche dalla possibilità di integrare più scenari, magari sviluppati da aziende di terze parti o dagli utenti stessi.
   Questo ragionamento può essere applicato anche ai Cyber Ranges, migliorando notevolmente l'esperienza d'uso e l'applicabilità in più domini;

6. Gestione delle competenze: i sistemi di gestione delle competenze (CMS) sono sistemi che permettono a un'organizzazione di gestire un programma di competenze dall'analisi dei gap di competenze e dal profilo degli utenti fino alla definizione dei percorsi di apprendimento e alla valutazione delle competenze.
   Possono anche includere un sistema di gestione dell'apprendimento (LMS) per l'amministrazione, la documentazione, il monitoraggio, il reporting e l'erogazione di contenuti di apprendimento e valutazione;

7. Collezione e analisi di dati: la raccolta di dati permette di raccogliere l'interazione degli utenti, tramite il traffico generato, i dump di memoria, gli strumenti utilizzati e via discorrendo. Tale capacità permette di apprendere come il CR venga usato, come si comportano gli utenti e la possibilità di ricevere un feedback diretto durante l'utilizzo. In Figura 10 si possono vedere le principali tipologie di dati secondo il sondaggio di Chouliaras et al. [11];

8. Sistema di punteggio: Il punteggio permette agli utenti di essere valutati in base alla loro attività e le interazioni con il CR. I sistemi di punteggio possono essere semplici come la raccolta punti delle sfide superate, o più avanzati

[12] `https://attack.mitre.org/`.

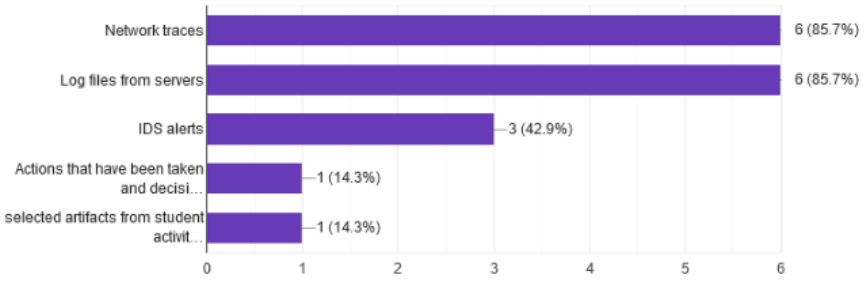


Figura 10: Dataset di un CR [11]

come l'analisi del traffico generato e i comandi lanciati. Il punteggio può poi essere mostrato direttamente durante l'utilizzo o alla fine con un report personalizzabile dall'utente;

9. Strumenti per l'istruttore: Si riferiscono alla capacità dell'istruttore di monitorare la sessione dell'utente e valutare le sue azioni. Questi dati aggiuntivi possono supportare il feedback finale e valutare gli obiettivi sopraggiunti. Alcune funzionalità possono essere la possibilità di avere una chat diretta con il trainee, mostrare suggerimenti per superare certe sfide, la capacità di registrare lo schermo degli utenti o analizzare le azioni eseguite.

Capitolo 3

Stato dell'arte dei Cyber Ranges

I Cyber Ranges hanno guadagnato popolarità negli ultimi anni come mezzo per convalidare le capacità organizzative. Si stanno esplorando sempre più campi di utilizzo e c'è un riconoscimento da parte della comunità scientifica e aziendale sui benefici dello sviluppo delle capacità umane insieme alla valutazione del livello di resilienza informatica.

Poiché i CR sono un dominio ancora in evoluzione, la progettazione di processi di formazione efficaci rimane ancora una questione aperta.

3.1 Panoramica delle ricerche accademiche

Nel passato sono stati eseguite diverse analisi dei moderni e più utilizzati Cyber Ranges, sia in ambito industriale che in ambito di training:

1. nel 2013 Magrath e Davis [15] hanno categorizzato i CR correnti sulla base della tipologia (simulazione, overlay o emulazione) e del settore di utilizzo (accademico, militare o commerciale) al fine di delineare le diversità e aiutare l'utilizzatore a scegliere l'approccio che soddisfa al meglio i requisiti necessari. La revisione effettuata ha identificato che la maggior parte dei CR utilizza la simulazione o l'emulazione:

   - Nella prima si fanno uso di software che simulano l'infrastruttura del mondo reale e ciò permette di effettuare simulazioni con un basso costo e una scalabilità ampia ma è difficile verificare se riflettono in maniera accurata la realtà.
     Un esempio è il National Cyber Range (NCR) [1];

[1] `https://www.peostri.army.mil/national-cyber-range-ncr`.

- Nella seconda viene creato un ambiente fisico rappresentativo con un conseguente vantaggio di eseguire esperimenti con alta fedeltà ma lo svantaggio principale è l'alto costo dovuto ai requisiti infrastrutturali. Un esempio è il CODE Cyber Range [2];

Gli autori fanno notare inoltre che gli approcci ibridi combinanti tra emulazione e simulazione stanno guadagnando popolarità;

2. Yamin et al. [16] effettuano una revisione sistematica della letteratura di circa 100 pubblicazioni relative a CR e sviluppa una tassonomia per classificare le capacità e le funzionalità generiche di CR;

3. Chaskos [17], nella sua tesi magistrale "Cyber-security training: A comparative analysis of cyber-ranges and emerging trends", revisiona 16 CR al fine di evidenziate le componenti chiave e le loro caratteristiche;

4. Chouliaras et al. [11] hanno condotto un indagine su dieci CR sviluppati nell'ultimo decennio tramite un intervista agli stakeholder, in modo da identificare dettagli sui componenti essenziali e agli strumenti utilizzati per progettare, implementare e gestire una piattaforma di CR;

5. CyberSec4Europe [18] ha recentemente effettuato un sondaggio online a utenti operatori e sviluppatori di CR, con risultati interessanti, come per esempio in Figura 11.
   L'obiettivo è stato quello di avere una comprensione dei CR esistenti e in fase di sviluppo, oltre che i casi d'uso, le capacità e l'infrastruttura.
   I risultati dell'indagine mostrano che i CR non vengono utilizzati solo per sviluppare le competenze e le conoscenze degli individui, ma anche per formare ed esercitare le organizzazioni o le aziende e i loro fornitori di servizi per sviluppare la loro resilienza informatica;

6. Ukwandu et al. [19] nell'articolo "A Review of Cyber-Ranges and Test-Beds: Current and Future Trends" hanno valutato le piattaforme CR e TB segmentandole per tipologia, tecnologia, scenari di minaccia e applicazioni e hanno sviluppato una tassonomia per ampliare una comprensione dei futuri CR.
   Durante l'analisi sono stati categorizzati 44 CR, nei quali si evince un utilizzo predominante in ambito accademico e di ricerca, mostrato in Figura 12, risultato in contrasto con l'analisi effettuata da Magrath [15] nel quale risultava un principale utilizzo in ambito training.
   Per quanto riguarda la tassonomia (Figura 13), per costruire il nucleo di un CR secondo Ukwandu deve essere definito uno scenario, che comprende diversi elementi:

[2] `https://www.concordia-h2020.eu/code-cyber-range/`.

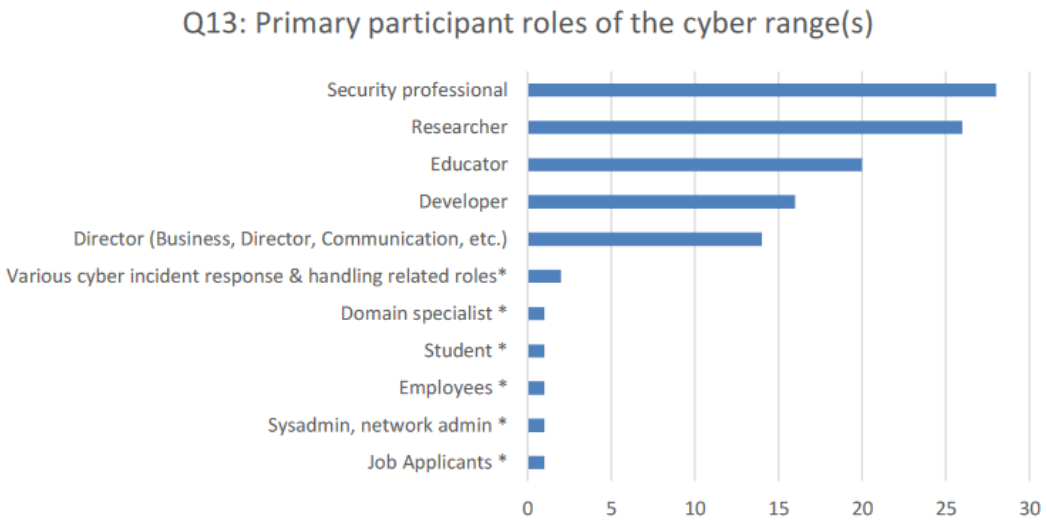


Figura 11: Principali Utilizzatori di CR secondo CyberSec4Europe [18]

- Una narrazione, che permette all'utente di evolvere nella storia e proseguire con le sfide;
- Un dominio di applicazione;
- La tipologia di istruzione, come casi di studio o momenti di riflessione durante le sfide;
- Utilizzo di gamification;
- Tipologia di scenario, che può essere statico o dinamico;

La definizione della tassonomia copre diverse caratteristiche possibili, tra le quali:

(a) Management: il livello di gestione di un CR può variare parecchio, presentando le informazioni agli utenti tramite un pannello di controllo insieme ai possibili scenari e tipi di attacco;

(b) Monitoraggio: l'amministratore può monitorare gli utenti sulla piattaforma, verificando i progressi ed effettuando valutazioni durante il CR. Questo punto permette anche la verifica di stato di salute della piattaforma e i vari servizi forniti;

(c) Infrastruttura: può essere un emulazione, simulazione o ibrida;

(d) Recovery: il componente di recovery assicura che la piattaforma sia aggiornata con le ultime patch, eseguendo anche backup regolari limitando anche gli attacchi informatici che potrebbero compromettere la piattaforma;

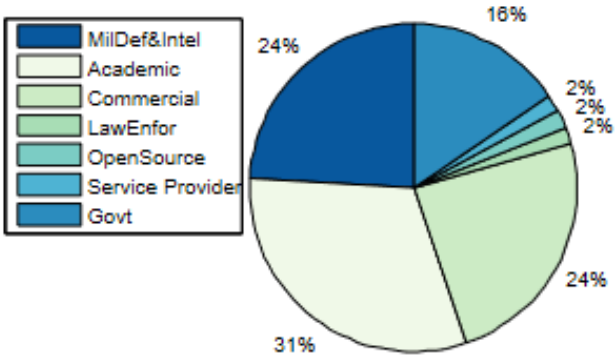


Figura 12: Principali Categorie di CR secondo Ukwandu et al. [19]

(e) Tipi di attacco: comprende le descrizioni dei diversi attacchi, incluse le configurazioni di sicurezza per le vulnerabilità all'interno degli scenari.

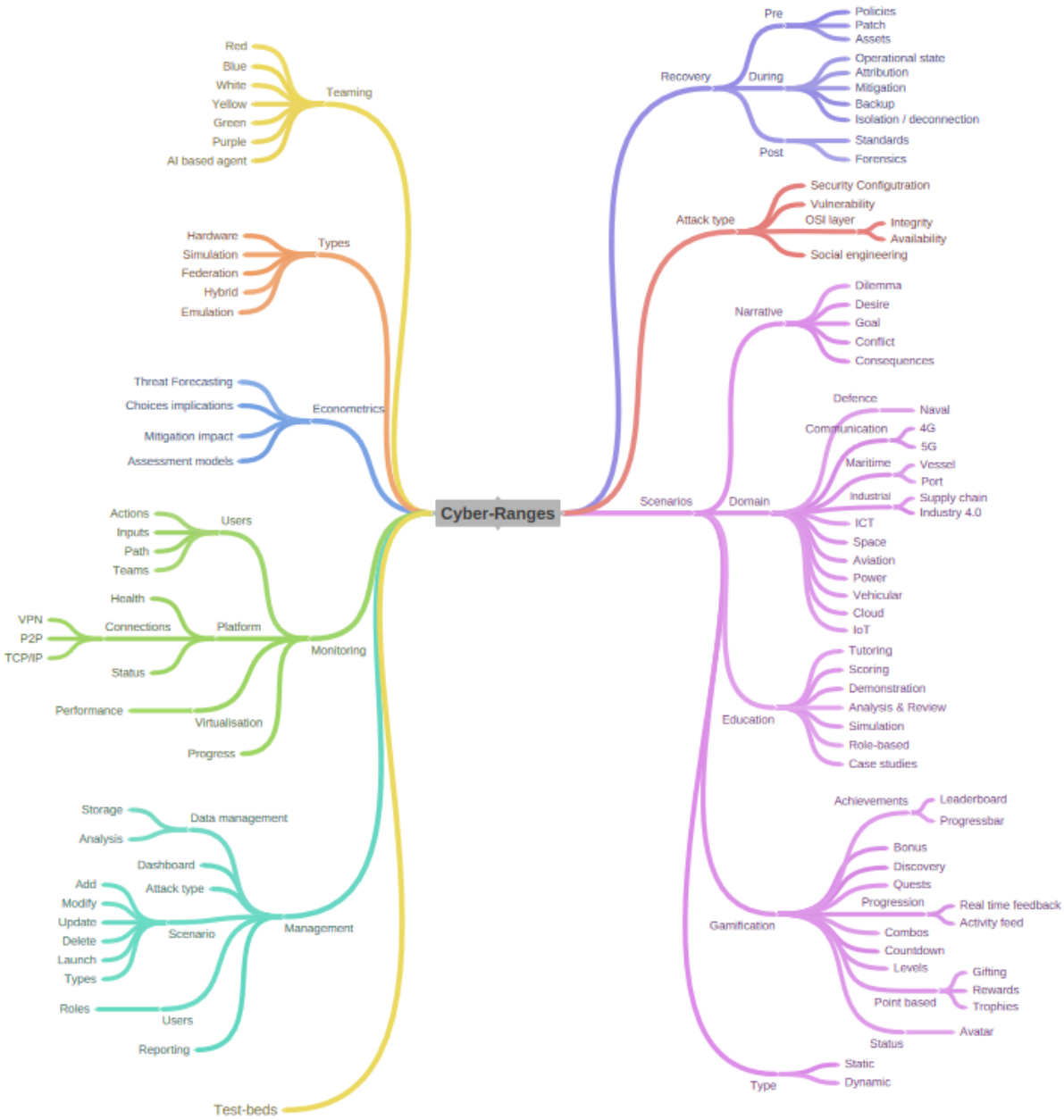


Figura 13: Tassonomia secondo Ukwandu et al. [19]

3.2 Principali Piattaforme

La richiesta e l'utilizzo dei CR sta aumentando notevolmente negli ultimi anni, e di conseguenza la loro presenza è sempre più rilevante. Di seguito una lista non esaustiva delle principali piattaforme, scelte sulla base della diversità e del livello di maturità.

3.2.1 Virgina Cyber Range

Il Virginia Cyber Range[3] è un'iniziativa dello stato del Virginia atta a migliorare l'educazione alla Cyber Security nelle scuole superiori, college e università. Il CR fornisce una ambiente basato sul cloud con laboratori pratici di sicurezza informatica e fornisce un repository di materiale didattico in modo da combinare la parte teorica con la parte pratica. Il materiale del corso viene offerto in maniera modulare per permettere la flessibilità dei corsi in modo da soddisfare le esigenze delle singole istituzioni.

Essendo basato sul cloud, gli studenti possono accedere al CR tramite un browser e permette anche esercitazioni con Capture The Flag in modo da effettuare esercitazioni su larga scala. Esso inoltre supporta una varietà di esercizi, tra i quali:

1. Analisi delle vulnerabilità di applicazioni web;
2. Analisi di digital forensics;
3. Esercizi di crittografia;
4. Hardening di server;
5. Rilevamento di intrusioni;

3.2.2 Cisco Cyber Range

Il CR sviluppato da Cisco[4] in Figura 14 emula situazioni del mondo reale in un ambiente di gioco di guerra per aiutare il personale a costruire e rafforzare competenze per far fronte agli incidenti informatici. Esso ha quattro componenti:

- Modelli basati sulle operazioni da eseguire in base a possibili scenari di attacchi;

[3] `https://www.virginiacyberrange.org/`.

[4] `https://www.ciscolive.com/c/dam/r/ciscolive/apjc/docs/2016/pdf/BRKSEC-2653.pdf`.

- Strumenti di sicurezza incorporati in piattaforme;
- Ambienti che simulano applicazioni reali;
- Ambienti basati sul cloud;

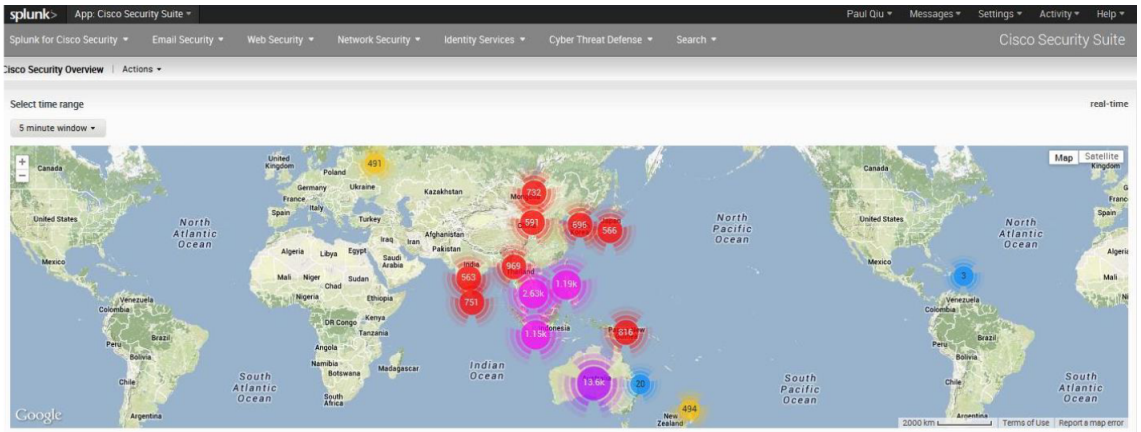


Figura 14: Cisco Cyber Range

L'infrastruttura supporta l'accesso cablato e da remoto, utilizza centinaia di campioni di malware per effettuare attacchi basati su esempi reali e permettere l'analisi di possibili scenari da parte degli esperti.

3.2.3 IBM X-Force

IBM X-Force[5] è uno dei primi CR fisici a fini commerciali, creato con lo scopo di aiutare i propri clienti a rispondere agli incidenti di Cyber Security. Esso si concentra su come le aziende possono affrontare massicce violazioni di dati dopo che sono state identificate ma anche come prevenirle.

Il suo CR utilizza la piattaforma Incident Response di Resilient System[6], la quale possiede uno dei più grandi database di Threat Intelligence e può gestire l'intero ciclo di vita di un attacco, partendo dal rilevamento di anomalie nei log fino all'identificazione di vulnerabilità.

I clienti che utilizzano questo CR possono testare i loro ambienti, eseguire possibili scenari di attacco e identificare i processi critici per gettare le basi e identificare eventuali problemi nel loro processo di Cyber Security.

[5] `https://www.ibm.com/it-it/security/services/managed-security-services/security-operations-centers`.

[6] `https://www.ibm.com/security/intelligent-orchestration/soar`.

3.2.4 NATO Cyber Range

La NATO conduce esercitazioni, addestramento e formazione tramite il suo Cyber Coalition[7], evento che si tiene annualmente per aiutare gli esperti di Cyber Security a sviluppare la loro capacità attraverso sfide realistiche.

Cyber Coalition, evento iniziato nel 2008, riunisce gli organismi e i partner NATO per rafforzare le capacità di identificare, difendere e contrastare le minacce cyber, tramite una forte collaborazione tra le varie squadre. L'esercitazione viene eseguita utilizzando il Training Center estone, detto CR14, uno degli ambienti più moderni di CR.

Rispetto a una mera competizione, la sfida è più una collaborazione tra i partecipanti che lavorano insieme per raggiungere un obiettivo comune. Oltre che verificare che i meccanismi preesistenti siano solidi, esso permette di sviluppare la consapevolezza della situazione e condividere le conoscenze tra i vari partner.

Può inoltre fornire ai suoi clienti un workshop di formazione tecnica specializzata per aiutarli a costruire l'esperienza necessaria per identificare e contrastare le minacce informatiche, sotto forma di un ambiente che permette di sfidarsi tra attaccanti e difensori.

3.2.5 CRATE

Il Cyber Range and Training Environment (CRATE)[8] è sviluppato e mantenuto dall'agenzia della difesa svedese. Esso permette di eseguire e configurare diverse macchine virtuali in un ambiente capace di generare traffico ad hoc emulando il comportamento di utenti.

L'infrastruttura fisica, mostrata in Figura 15, comprende circa 800 server ed è capace di progettare reti di computer, distribuire diversi sistemi operativi con le loro applicazioni e specificare gli utenti per ogni macchina eseguita. Ogni attività di configurazione può essere gestita da un interfaccia grafica, la quale permette di manipolare i parametri che controllano i sistemi operativi, i firewall e le applicazioni. Esso utilizza poi strumenti visivi e testuali per il rilevamento delle intrusioni e strumenti di simulazione per la formazione.

[7] `https://www.act.nato.int/cyber-coalition`.

[8] `https://www.foi.se/en/foi/resources/crate---cyber-range-and-training-environment.html`.

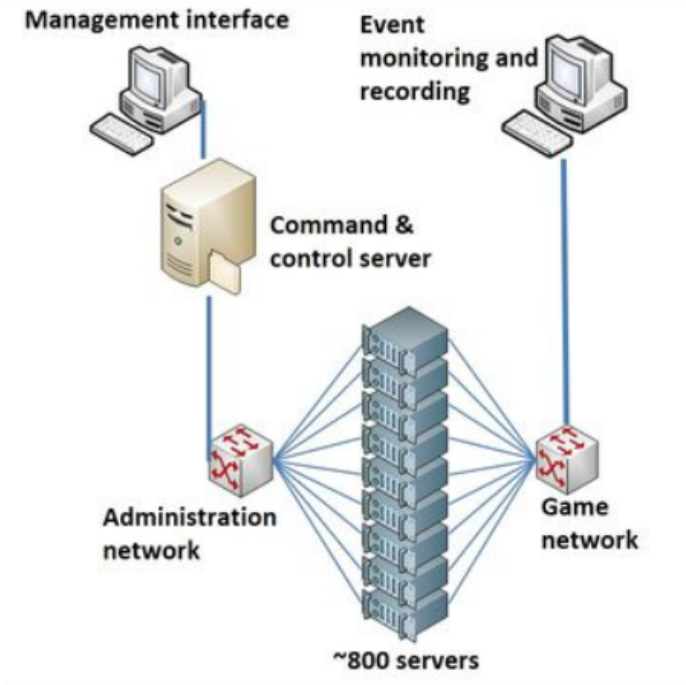


Figura 15: CRATE Cyber Range

3.2.6 KYPO

Sviluppata dall'università di Masaryk, la piattaforma KYPO[9] è fortemente orientata all'utente e concentrata sulla fornitura di soluzioni pratiche per l'istruzione degli studenti e la formazione dei professionisti della Cyber Security. I casi d'uso presenti sono utilizzati da classi universitarie e corsi di formazione per le aziende.

I casi d'uso sono diversi, tra i quali:

1. Ricerca e sviluppo: uno dei motivi principali per cui è stata sviluppata, i ricercatori possono creare reti di server e desktop e la piattaforma fornisce sandbox per le ricerche da effettuare;

2. Analisi forensi: utilizzata da esperti di forensics, gli utenti possono creare immagini virtuali ed eseguire software dannosi per effettuare analisi dinamiche automatizzate;

3. Educazione e formazione: gli utilizzatori possono partecipare ad attività di apprendimento pratico, facendo sia esercizi di Capture The Flag, sia esercitazioni singole;

[9] `https://is.muni.cz/publication/1386573/2017-ICSOFT-kypo-cyber-range-design-presentation.pdf`.

Il CR è stato rilasciato sotto licenza open source[10] sotto licenza MIT ed è completamente gratuito.

[10] https://docs.crp.kypo.muni.cz/.

Capitolo 4

Generazione model driven di scenari

La preparazione di un buon scenario per un Cyber Range è un'operazione costosa in termini di tempo e di competenze. L'approccio Model Driven permette di diminuire i tempi di sviluppo e, tra i vari benefici, permette di eseguire operazioni a costi inferiori (dati anche dal numero ridotto di persone affiancate al progetto). Tale approccio implica anche meno errori, in quanto i possibili problemi tecnici sono concentrati nello sviluppo del modello, ottenendo come risultato una qualità maggiore rispetto alla classica programmazione. Inoltre, esso impone anche una separazione delle competenze, dove gli esperti del dominio possono concentrarsi sulla modellazione delle specifiche e gli specialisti tecnici si occupano della costruzione degli strumenti necessari per lo sviluppo del modello.

Il lavoro svolto si è concentrato sull'identificazione di una piattaforma libera, aggiornata e adatta ad essere inserita in un modello per poter essere liberamente fruibile e utilizzabile da chiunque, senza necessità di particolari competenze.

L'idea alla base è stata quella di avere a disposizione diversi scenari di attacco (in questo caso categorizzati nella OWASP Top 10[1]) per poter stabilire in anticipo l'esercitazione da effettuare e, in contemporanea, avere una piattaforma di analisi del punteggio, utile all'istruttore per analizzare l'andamento dell'esercitazione.

4.1 Piattaforma THREAT ARREST

L'approccio model driven per creare un Cyber Range è già stato affrontato dal progetto THREAT-ARREST[2] e potrebbe essere una possibile futura implementazione del lavoro svolto durante la stesura di questa tesi.

[1] `https://owasp.org/www-project-top-ten/`.
[2] `https://www.threat-arrest.eu/`.

4.1.1 Architettura della piattaforma

Nella piattaforma THREAT-ARREST viene utilizzato un approccio ingegneristico per sviluppare un framework che permetta la costruzione di scenari di Cyber Security. È stato sviluppato un linguaggio ad hoc per la costruzione di scenari, che permette la descrizione dell'architettura del sistema in analisi e un meccanismo per distribuirli sull'infrastruttura OpenStack tramite modelli HEAT.

4.1.2 Descrizione della piattaforma

La piattaforma THREAT-ARREST utilizza un'infrastruttura di emulazione che permette la compilazione e la distribuzione automatica degli scenari di formazione tramite macchine virtuali a cui possono accedere i trainer e rispettivi trainee. Essa fornisce anche un'infrastruttura di controllo e monitoraggio con interagisce con altri strumenti componenti l'infrastruttura. La piattaforma di emulazione (Figura 16) si connette tramite le API fornite dai framework OpenStack e HEAT.

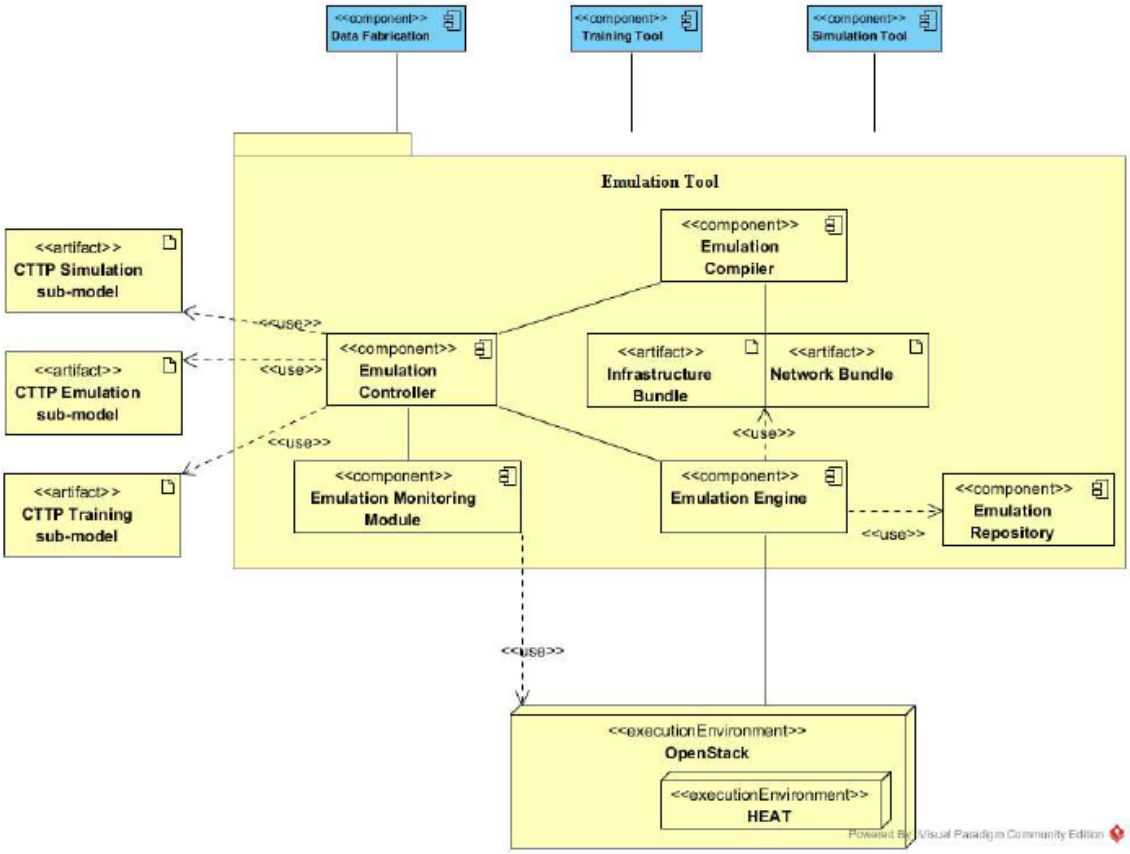


Figura 16: Piattaforma di emulazione

OpenStack

OpenStack è una piattaforma open source che utilizza risorse virtuali in comune per costruire e gestire cloud privati e pubblici. Gli strumenti che compongono la piattaforma OpenStack, chiamati progetti, gestiscono i servizi fondamentali di cloud computing di calcolo, networking, storage e identità. Esso utilizza un insieme di API per gestire le proprie risorse, utilizzabili con meccanismi di autenticazione.

Il cloud è tutto ciò che riguarda l'offerta di elaborazione per gli utenti finali in un ambiente remoto, dove il software effettivo viene eseguito come servizio su server affidabili e scalabili piuttosto che sul computer di ciascun utente finale. Il cloud computing può riferirsi a molte cose diverse, ma tipicamente l'industria parla di eseguire diversi elementi come servizio: software, piattaforme e infrastrutture. OpenStack rientra nell'ultima categoria ed è considerato Infrastruttura come servizio (IaaS).

Fornire un'infrastruttura significa che OpenStack rende semplice per gli utenti aggiungere rapidamente nuove istanze, su cui altri componenti del cloud possono essere eseguiti. OpenStack è composto da molti componenti (Figura 17). Proprio grazie al fatto che sia un software open-source, chiunque può aggiungere ulteriori componenti a OpenStack per aiutarlo a soddisfare le proprie esigenze.

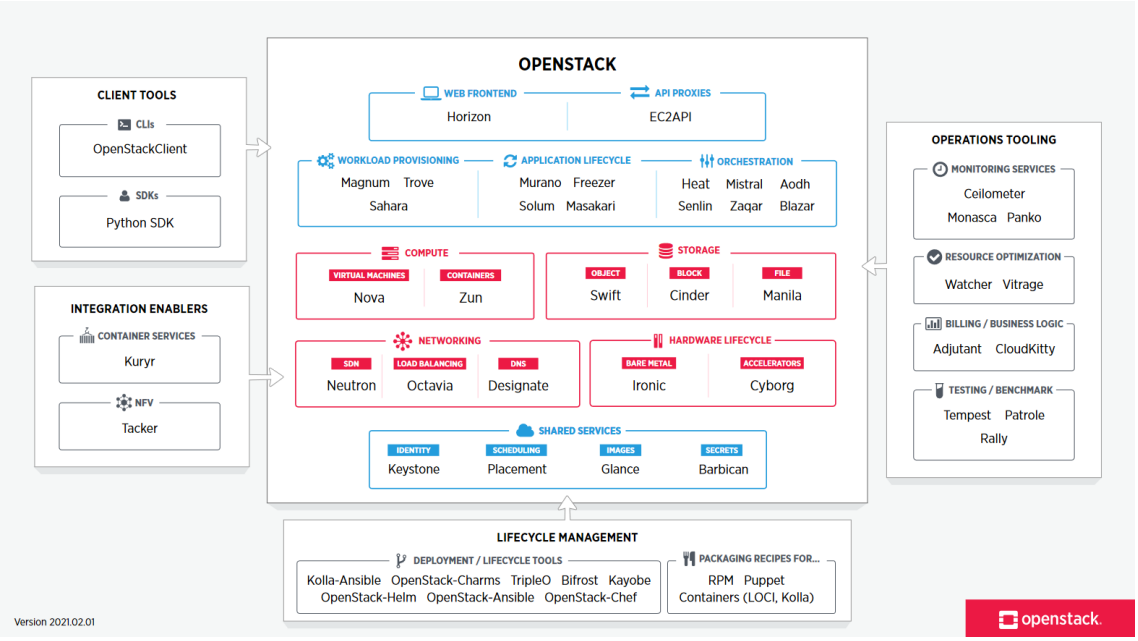


Figura 17: Componenti di OpenStack

La community di OpenStack ha collaborativamente identificato nove componenti chiave che fanno parte del core, e che sono distribuiti come parte di ogni

sistema OpenStack e ufficialmente mantenuti dalla community:

1. Nova: è il motore di calcolo principale. È usato per distribuire e gestire le macchine virtuali, i container e altre istanze;

2. Swift: è un sistema di archiviazione per oggetti e file. Piuttosto che l'utilizzo tradizionale di riferirsi ai file per la loro posizione sull'unità del disco, gli sviluppatori possono fare riferimento a un identificatore univoco che si riferisce al file o al pezzo di informazione e lasciare che OpenStack decida dove memorizzare queste informazioni. Questo rende la scalabilità più semplice, poiché gli sviluppatori non devono preoccuparsi della capacità di un singolo sistema dietro al software;

3. Cinder: è un componente di archiviazione a blocchi, che è più analogo alla nozione tradizionale di un computer in grado di accedere a posizioni specifiche su un'unità disco;

4. Neutron: fornisce la capacità di networking per OpenStack. Aiuta a garantire che ciascuno dei componenti di un'implementazione OpenStack possa comunicare tra loro in modo rapido ed efficiente;

5. Horizon: è l'interfaccia grafica di OpenStack, quindi per gli utenti che vogliono provare OpenStack, questo può essere il primo componente che effettivamente vedono. Gli sviluppatori possono accedere a tutti i componenti di OpenStack individualmente attraverso un'interfaccia di programmazione delle applicazioni (API), ma la dashboard fornisce agli amministratori di sistema uno sguardo a ciò che sta accadendo nel cloud, e per gestirlo secondo le necessità;

6. Keystone: fornisce servizi di identità per OpenStack. È essenzialmente un elenco centralizzato di tutti gli utenti del cloud OpenStack, mappato rispetto a tutti i servizi forniti dal cloud, che hanno il permesso di utilizzare. Fornisce molteplici mezzi di accesso, il che significa che gli sviluppatori possono facilmente mappare i loro metodi di accesso utente esistenti all'interno di esso;

7. Glance: fornisce le immagini (copie virtuali) dei dischi rigidi. Esso permette di utilizzarle come modelli quando si distribuiscono nuove istanze di macchine virtuali;

8. Ceilometer: si occupa della telemetria e mantiene un conteggio dell'utilizzo del sistema di ogni utente di ciascuno dei vari componenti di un cloud OpenStack;

9. Heat: è il componente di orchestrazione di OpenStack, che permette agli sviluppatori di memorizzare i requisiti di un'applicazione cloud in un file che definisce quali risorse sono necessarie per tale applicazione. In questo modo, aiuta a gestire l'infrastruttura necessaria per l'esecuzione di un servizio cloud.

HEAT

Heat è un motore di orchestrazione per lanciare più applicazioni cloud (chiamate stack) basate su modelli (HEAT Orchestration Template - HOT) in forma di file di testo leggibili e scrivibili dall'uomo che possono essere trattati come codice dal sistema. Esso fornisce agli utenti OpenStack un modo per automatizzare la creazione di componenti cloud come reti, istanze, dispositivi di storage e molto altro.

Un modello HOT permette la creazione della maggior parte dei tipi di risorse OpenStack (come istanze, reti e subnet, floating IP, volumi, gruppi di sicurezza, utenti, ecc.), così come alcune funzionalità più avanzate come la possibilità di configurare il software installato sulle VM lanciate. I modelli HOT sono definiti secondo la notazione YAML, leggibili e scrivibili dall'uomo come in Figura 18.

```yaml
heat_template_version: 2018-08-31

parameters:
  key1:
    default: threatarrest_key
    type: string

  cirros:
    default: cirros-0.4.0-x86_64-disk
    type: string

  m1.small:
    default: m1.small
    type: string

resources:
  minimal_instance:
    type: OS::Nova::Server
    properties:
      name: test1_v3_svr
      key_name: get_param: threatarrest_key
      image: get_param: cirros
      flavor: get_param: m1.small
```

Figura 18: Esempio di template HOT

Ogni template HOT deve includere almeno:

- la chiave di versione del template con una versione valida di HOT[3];
- la sezione risorse, che contiene la dichiarazione delle risorse da distruibuite, selezionate dall'elenco OpenStack Resource Types [4];
- la sezione dei parametri, che è invece facoltativa e permette di includere tutti i dettagli delle configurazioni che saranno utilizzate nello scenario. Alcuni parametri devono essere specificati per produrre un valido template, come:
  - image: specificando l'ID dell'immagine della macchina virtuale che il sistema dovrà creare;
  - flavor: descrive la proprietà della VM come memoria, CPU virtuali, etc.;
  - key: chiave SSH pubblica da inserire nell'istanza al momento della creazione, in modo da poter accedere al sistema;
  - security groups: un'insieme di regole di accesso di rete utilizzate per limitare il traffico che ha accesso alle istanze;
  - network: la macchina virtuale deve essere connessa alla rete.

4.1.3 Sviluppo Model Driven di un DSL

Il Model Driven Development è una metodologia di sviluppo del software che permette agli utenti di costruire applicazioni complesse tramite astrazioni semplificate di componenti pre-costruiti. Esso combina diverse tecnologie, tra cui [20]:

1. Linguaggi di modellazione specifici per il dominio: vengono descritti come metamodelli che definiscono le relazioni tra i concetti in un dominio e specificano con precisione le chiavi semantiche e i vincoli associati ai concetti di dominio. Gli sviluppatori usano i DSML (Domain Specific Modeling Language) per costruire applicazioni usando elementi del sistema di tipi catturati dai metamodelli ed esprimono l'intento di progettazione in modo dichiarativo piuttosto che imperativo;

2. Motori di trasformazione e generatori che analizzano determinati aspetti dei modelli e poi sintetizzano vari tipi di artefatti, come codice sorgente, input di simulazione, descrizioni di distribuzione XML, rappresentazioni alternative del modello.

[3] `https://docs.openstack.org/heat/rocky/template_guide/hot_spec.html#hot-spec`.
[4] `https://docs.openstack.org/heat/latest/template_guide/openstack.html`.

L'utilizzo di un metamodello per caratterizzare gli elementi della sintassi permette di separare la sintassi astratta e la semantica dei costrutti del linguaggio dalle loro notazioni concrete.

Un linguaggio specifico del dominio (DSL) è un linguaggio di programmazione o di specifica eseguibile che offre, attraverso annotazioni e astrazioni appropriate, una potenza espressiva focalizzata e solitamente limitata a un particolare dominio di problemi [21]. Essi sono precisi, possono essere utilizzati per diversi scopi, aiutano nella produttività e nella portabilità e permettono la convalida e l'ottimizzazione a livello di dominio. Gli svantaggi nell'utilizzo includono invece il costo di design e implementazione, il costo nell'educazione a essi e la disponibilità limitata. Alcuni tra i più conosciuti DSL sono SQL, HTML e XML.

Nello sviluppo di un linguaggio ci possono essere due approcci:

1. Basati sulla grammatica: essi generano metamodelli partendo dalla definizione della grammatica esistente;

2. Basati sui metamodelli: generano grammatiche partendo da metamodelli esistenti.

Ad oggi, la denotazione di una sintassi DSL tramite metamodello è supportata da diversi ambienti di modellazione, anche se l'approccio più sfruttato attualmente è una combinazione tra i due approcci, dove da una parte si ha Xtext [5] che utilizza un approccio grammaware mentre dall'altra EMF (Eclipse Metamodeling Framework [6]) che è il toolkit come standard di fatto per il modelware:

- L'Eclipse Modeling Framework (EMF) è un insieme di plug-in di Eclipse che può essere usato per modellare un modello di dati e per generare codice o altri output basati su questa modalità. EMF ha una distinzione tra il meta-modello e il modello effettivo. Il meta-modello descrive invece la struttura del modello;

- Xtext invece si concentra sulla descrizione di una sintassi testuale concreta per un linguaggio e deriva tutto dalla sintassi stessa, incluso un modello EMF Ecore che rappresenta l'AST (Abstract Syntax Tree);

Per implementare un DSL è necessario seguire i seguenti passi:

1. Definizione di una sintassi concreta (testuale, grafica, mista);

2. Definizione di una sintassi astratta;

3. Sviluppo di un ambiente di editing/modellazione dei modelli;

[5] `https://www.eclipse.org/Xtext/`.
[6] `https://www.eclipse.org/modeling/emf/`.

4. Serializzazione/Deserializzazione per i modelli;
5. Definizione della semantica statica (soprattutto controllo dei tipi);
6. Definizione della semantica dinamica (interprete, compilatore di modelli, generatore di codice);

Xtext usa modelli EMF[7] come rappresentazione in memoria di qualsiasi file di testo analizzato. La rappresentazione è un grafo di oggetti chiamato AST (abstract syntax tree). A seconda della comunità questo concetto è anche chiamato modello di oggetto del documento (DOM) o modello semantico. L'AST contiene l'essen-

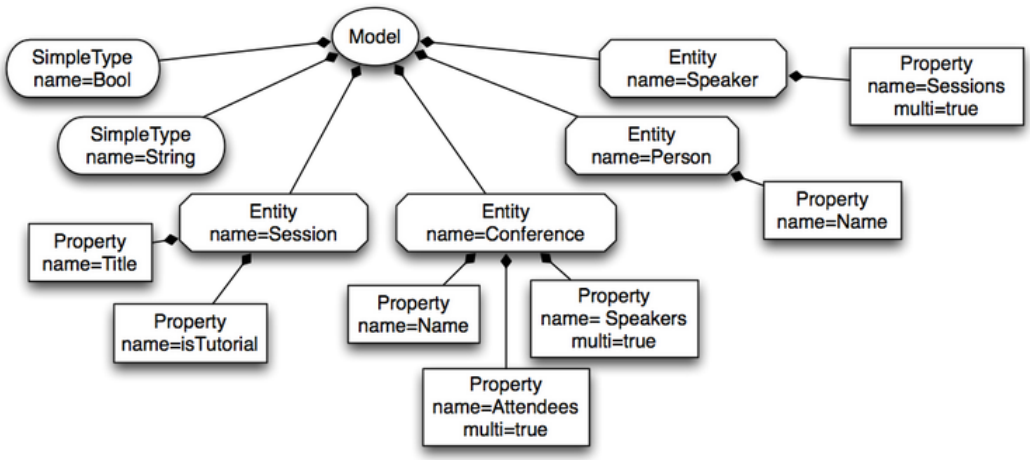


Figura 19: Esempio AST

za dei modelli testuali e astrae le informazioni sintattiche (Figura 19 mostra un esempio di AST per la gestione delle informazioni relative ai relatori di una conferenza). Esso è poi usato da fasi successive di elaborazione, come la validazione, la compilazione o l'interpretazione.

Il linguaggio con il quale il metamodello è definito viene chiamato Ecore. In sostanza, il metamodello è il modello Ecore del linguaggio sviluppato.

Il parser DSL viene generato automaticamente da Xtext e può essere poi completato da un validatore per eseguire l'analisi statica dei modelli DSL e dare un feedback informativo agli utenti.

La validazione standard di Xtext funziona controllando precisi vincoli sugli elementi di metamodellazione espressi in termini di formule OCL (Object Constraint Language).

[7] `https://www.eclipse.org/Xtext/documentation/308_emf_integration.html`.

Una volta che la grammatica è stata definita, può essere generato un editor Eclipse all'interno di Xtext, il quale permette all'utente una leggibilità e un'assistenza al contesto.

La generazione del codice è poi fatta con Xtend, un linguaggio di programmazione generale di alto livello per la Java Virtual Machine. Successivamente viene generato in maniera automatica un XML tramite IText.

4.1.4 Cybersecurity Training Language

L'obiettivo della piattaforma THREAT-ARREST è quello di sviluppare Cyber Ranges di alta qualità che permettano di essere rapidamente implementabili e configurabili. Per questo motivo essa include passaggi semi automatici che rendono molto semplice il passaggio dal modello di uno scenario a un ambiente funzionante dove poter eseguire il training.

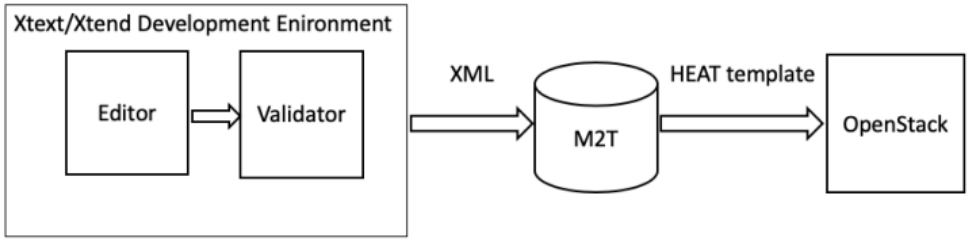


Figura 20: Architettura

Il modulo che permette al modellatore di fornire una descrizione del sistema è simulato utilizzando un DSL, il Cybersecurity Training Language. Il modello viene poi tradotto automaticamente in una descrizione intermedia (tramite XML) che viene poi tradotta in un modello HEAT e fornita come input per HEAT.

Come anticipato, EMF fornisce strumenti per produrre un insieme di classi Java per il modello, insieme a un insieme di classi che permettono la visualizzazione dello stesso.

EMF include anche un meta-modello (Ecore) che permette di descrivere i modelli e il supporto runtime per i modelli, compresa la modifica dei cambiamenti, il supporto della persistenza e un API per modellare gli oggetti EMF[8].

In EMF sono supportati tre livelli di generazione del codice:

- Model: fornisce le interfacce Java e le classi di implementazione per tutte le classi del modello, insieme a una classe di implementazione del package;

[8] `https://www.eclipse.org/modeling/emf/`.

- Adapters: genera le classi di implementazioni che adattano le classi del modello per la modifica e la visualizzazione;
- Editor: produce un editor conforme allo stile che viene raccomandato per gli editor di modelli EMF di Eclipse.

Come menzionato precedentemente, per implementare il linguaggio è stato usato il framework Xtext Eclipse.

La grammatica generata è composta da una serie di regole che definiscono i singoli elementi del linguaggio generato. Prendendo come esempio la Figura 21, la prima regola denota il primo elemento del modello del CTL, il quale dichiara che un modello di CTL è una collezione di elementi di attività, mentre nella regola successiva si specifica che un asset può essere di tre tipi diversi.

```
5  Model:
6      'emulation' name=ID '{' assets+=Asset* '}';
7
8  Asset:
9      Node | NetworkDevice | NetworkConnection;
10
11 Node:
12     GenericVM | ExternalDevice;
13
14 GenericVM:
15     CustomVM | 'VM' name=ID '{' properties += VMProperty* '}';
16
17 VMProperty:
18     VMHardwareProperty ';' |
19     VMSoftwareProperty ';' |
20     VMVulnerability ';';
```

Figura 21: Esempio di CTL

Dopo aver creato tutti gli artefatti derivanti dalla grammatica, Xtext genera automaticamente un editor testuale e un parser. Un esempio di creazione di un modello (Figura 22) è uno scenario in cui ci sono due macchine virtuali: una che rappresenta un sistema LAMP con un'applicazione web vulnerabile a una SQL Injection e l'altra utilizzata da un client per navigare l'applicazione Web. Il parser esegue la validazione della sintassi basata sulla grammatica del linguaggio e per esempio, grazie alla denominazione delle connessioni di rete il parser è in grado di controllare che le VM sulla rete siano già state definite come nodi.

Successivamente viene eseguito un processo di traduzione, che permette di passare da CTL a HEAT e l'infrastruttura Open Stack è pronta per l'esecuzione del modello definito precedentemente.

```
emulation emulation_scenario {
    VM server {
        RAM_size: 16 GB;
        CPU_speed: 2.5 GHz;
        Disk_size: 250 GB;
        Web_server: "Apache HTTP Server";
        Operating_System: "Linux";
        Database: "MySQL";
        Programming_language: "PHP"
        Web_application: "https://www.homebanking.com";
        Vulnerability SQLi: "SQL injection";
    }
    VM client {
        Operating_System: "Ubuntu";
        Web_browser: "Mozilla Firefox";
    }

    Network_connection {
        VM server has this public IP ...
        ...
    }
}
```

Figura 22: Esempio di Configurazione

4.2 Contesto di partenza

4.2.1 OWASP Top 10

L'Open Web Application Security Project® (OWASP) è una fondazione senza scopo di lucro che lavora per migliorare la sicurezza del software. Attraverso progetti di software open-source guidati principalmente da volontari, centinaia di sedi locali in tutto il mondo e conferenze educative e formative di primo piano, la Fondazione OWASP si occupa di migliorare la consapevolezza e rendere più sicuro il web[9].

L'OWASP Top Ten[10] è un documento di riferimento che delinea i dieci problemi di sicurezza più critici per la sicurezza delle applicazioni web. Il documento è creato da un team di esperti di sicurezza di tutto il mondo, mentre i dati per la classificazione provengono da molteplici organizzazioni.

Esso si riferisce alla classificazione come un documento di consapevolezza e raccomanda a tutte le organizzazioni di incorporarlo nei loro processi al fine di mitigare i rischi per la sicurezza. È bene ricordare che il documento non è uno standard ma può essere definito come una matrice da definire in base all'ambiente dell'organizzazione che lo incorpora.

La classificazione del 2017, mostrata in Figura 23, è la seguente:

[9] `https://owasp.org/`.

[10] `https://owasp.org/www-project-top-ten/`.

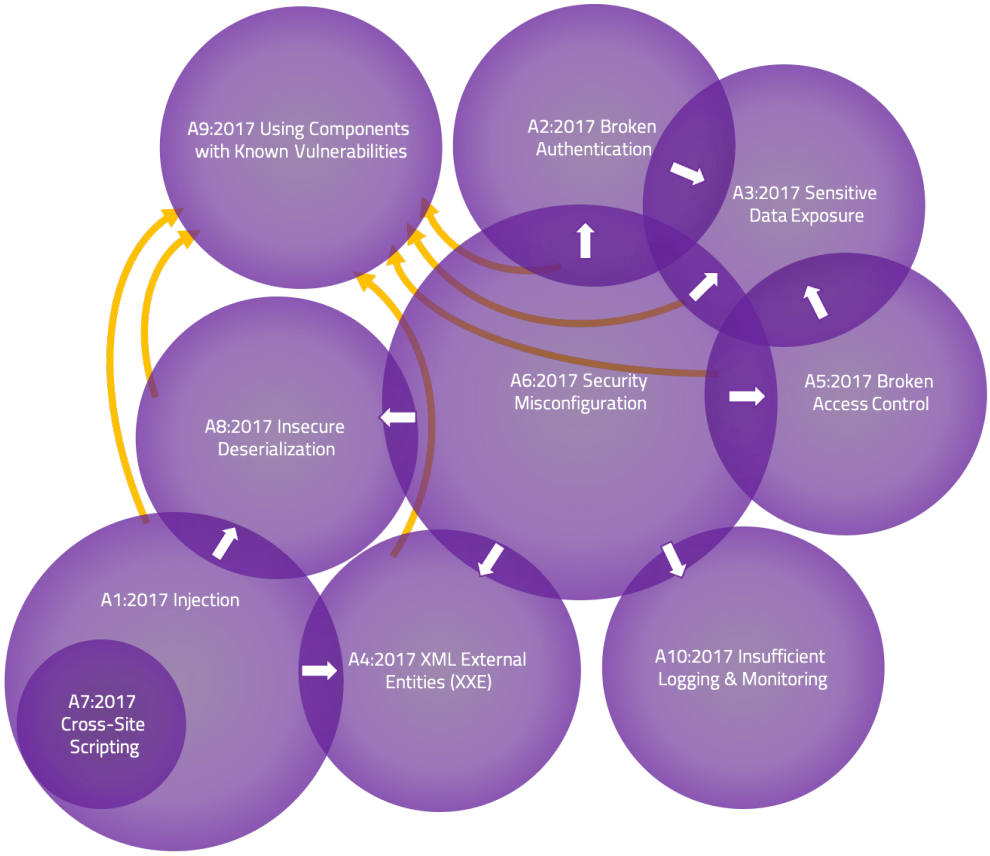


Figura 23: Owasp Top 10

1. Injection Flaws: si verificano quando i dati inviati dall'utente vengono inviati a un interprete come parte di un comando o di una query. I dati possono iniettare nell'interprete comandi non voluti o permettere l'accesso a dati senza averne autorizzazione. Sono parte di questa categoria vulnerabilità come SQL, NoSQL, OS e LDAP injection;

2. Broken Authentication: le funzioni delle applicazioni relative all'autenticazione e alla gestione delle sessioni sono spesso implementate in modo non corretto, permettendo a un utente malintenzionato di assumere le identità di altri utenti o compromettere password o token di sessione;

3. Sensitive Data Exposure: molte applicazioni web e API non proteggono adeguatamente i dati sensibili. Un attaccante potrebbe sottrarre o modificare tali dati;

4. XML External Entities: molti parser XML non aggiornati o mal configurati

valutano i riferimenti a entità esterne all'interno dei documenti XML. Queste entità possono essere usate per rivelare file interni al sistema, effettuare scansioni di porte interne o esecuzione di codice remoto;

5. Broken Access Control: le restrizioni su ciò che gli utenti autenticati sono autorizzati a fare sono spesso applicate in maniera non corretta. Un utente malevolo potrebbe sfruttare queste falle per accedere a dati non autorizzati, come ad esempio dati di altri utenti o file sensibili;

6. Security Misconfiguration: le misconfigurazioni di sicurezza sono uno dei problemi più comuni. Il motivo è che quando un software viene installato spesso ha delle configurazioni di default che non tengono conto della sicurezza dello stesso. I software, oltre a dover essere configurati in modo sicuro, devono essere sottoposti a regolari aggiornamenti in modo tempestivo;

7. Cross Site Scripting: le vulnerabilità XSS si verificano quando un'applicazione include dati provenienti dall'esterno (come l'input dell'utente) senza che venga effettuata un'adeguata validazione. Questo permette all'attaccante di eseguite Javascript nel browser della vittima e potenzialmente sottrarre la sessione di quest'ultimo o modificare le funzionalità di un sito web;

8. Insecure Deserialization: una deserializzazione insicura spesso porta all'esecuzione di codice remoto. Spesso inclusa in software sviluppati con linguaggio Java (ma non solo), può essere utilizzata per eseguire molteplici attacchi;

9. Using Components with Known Vulnerabilities: i componenti come le librerie, i framework e altri moduli software devono essere tenuti aggiornati quotidianamente, in quanto una software obsoleto ha più probabilità di avere vulnerabilità note che possono essere sfruttate senza grandi difficoltà;

10. Insufficient Logging and Monitoring: un monitoraggio insufficiente permette agli aggressori di attaccare i sistemi, mantenere l'accesso e passare ad altri sistemi interni senza che l'organizzazione si accorga di nulla.

Gli attaccanti possono potenzialmente usare molti percorsi per riuscire a sfruttare una vulnerabilità o una catena di vulnerabilità e ottenere accesso a dati sensibili. A volte questi percorsi possono essere banali da identificare e sfruttare, mentre in altri casi possono essere estremamente complessi.

Per determinare il rischio di un'organizzazione bisogna valutare la probabilità associata al vettore di attacco e combinarla con una stima dell'impatto tecnico o economico, in modo da pesarne tutti i fattori.

4.2.2 Capture the Flag

Nel contesto della sicurezza informatica, le Capture The Flag (CTF) sono competizioni tra gruppi o individui in cui l'obiettivo è quello di risolvere sfide di sicurezza informatica, guadagnando punti man mano che vengono risolte. L'ambiente può comprendere una singola pagina web o può estendersi su un'intera rete di computer, può essere virtualizzato o fisico e può differenziarsi in diverse categorie.

Il primo CTF di sicurezza informatica fu sviluppato nel 1996 al DEFCON a Las Vegas. DEFCON è la più grande conferenza sulla sicurezza informatica negli Stati Uniti ed è ufficialmente iniziata nel 1993, ideata da Jeff Moss[11].

Mentre gli eventi CTF sembrano essere solo un grande divertimento per gli appassionati, c'è molto di più che risolvere semplicemente delle sfide. Essi offrono sia una piccola finestra nel campo della sicurezza informatica per incontrare nuove persone, trovando individui di talento all'interno di discipline specifiche, sia un modo per i professionisti della sicurezza per mostrare le loro abilità.

Gli studenti imparano la sicurezza informatica in diversi modi. La lettura di documenti di conferenze o articoli di riviste per esempio permette a essi di acquisire nozioni teoriche e di imparare ciò che è stato appreso durante le lezioni. Anche la progettazione e l'implementazione di sistemi difensivi è importante, poiché permette l'applicazione di conoscenze astratte per costruire soluzioni reali e funzionanti.

Oltre a queste classiche modalità educative, praticare l'attacco e la difesa di sistemi informatici in tempo reale e in situazioni che simulano la realtà ha un grande valore educativo, in quanto permette agli studenti di mettere in pratica le conoscenze acquisite senza paura di conseguenze o rappresaglie, soprattutto se in esercizi di offensive security. La componente offensiva sfida il ragionamento del trainee e permette a esso di ragionare in modo da identificare le possibili strategie di attacco, portando una profonda comprensione dell'informatica.

I CTF possono essere di diverse tipologie, ma le forme più comuni sono Jeopardy e Attacco e Difesa. [22]

Jeopardy

È uno dei format più frequenti per le competizioni odierne, essi contengono una serie di sfide digitali create appositamente per l'evento nelle quali i partecipanti affrontano ogni sfida come un problema a sé stante e la risoluzione di una sfida comporta l'aumento del punteggio (la cattura della bandiera) che può essere diversificato in base alla difficoltà della sfida.

Le sfide possono essere altamente variabili (Figura 24), possono essere indipendenti l'una dall'altra e i partecipanti dovrebbero poterle risolvere in qualsiasi

[11] `https://www.defcon.org/html/links/dc-ctf-history.html`.

ordine le eseguano. Le sfide possono essere contenute in file scaricabili (come nel caso di sfide di binary exploitation, dove è necessario scaricare il file e analizzarlo sul proprio computer) o applicazioni web, che, per esempio, possono essere affette da una determinata vulnerabilità e una volta identificata e sfruttata permettono l'accesso alla bandiera.

Figura 24: Esempio di Jeopardy CTF

Le categorie sono molte, alcune più comuni, come web, binary exploitation o crittografia, altre meno comuni, come la categoria misc che esplora aree più generali, solitamente incorporando all'interno di sé diverse categorie. Alcune categorie più comuni sono:

1. Pwning: replicano attacchi su reali servizi vulnerabili. In certi casi possono comportare la compromissione di un software remoto mentre in altri casi sono file binari creati ad hoc dagli organizzatori contenenti vulnerabilità come buffer overflow o command injection. Nel caso la vulnerabilità venga sfruttata in maniera adeguata, il binario ritornerà la flag;

2. Reversing: comportano il reverse engineering di un software al fine di estrarre la flag. In genere non hanno un servizio remoto da compromettere ma la flag è contenuta all'interno del file ed è solamente da estrarre. Esse richiedono la comprensione di come funziona un determinato tipo di software, le logiche all'interno di esso e la possibilità di dover raggirare determinati blocchi nel codice compilato;

3. Web: seguono uno schema molto simile alle pwning ma l'obiettivo ora è il server web, solitamente tramite HTTP e un database sottostante. I partecipanti cercano di attaccare le pagine dell'applicativo, con SQL Injection e vulnerabilità insite nel codice del server;

4. Crittografia: in questo caso coinvolgo l'analisi crittografica di messaggi o protocolli in modo da rivelare la flag nascosta;

5. Forensics: coinvolgono i partecipanti a indagare su un dato a prima vista sconosciuto, cercando o costruendo uno strumento in grado di estrapolare le informazioni. Solitamente la flag è contenuta all'interno del dato ma ofuscato e i partecipanti devono capire il formato per poterlo leggere. Ci possono essere diverse sfide da superare per ottenerlo, tra le quali aggiustare file corrotti, montare unità disco non chiare o cercare file nascosti;

6. Miscellaneous: sfide varie che non rientrano in nessuna delle categorie precedenti o che sono la composizione di più categorie diverse. Questa tipologia permette ai partecipanti di pensare a tutte le possibili strade e soluzioni e applicare la creatività;

7. Programming: permettono agli utenti di testare la loro capacità nello scrivere codice, molto simile alle competizioni hackaton specifiche per la programmazione;

Per ogni sfida completata con successo, il partecipante ottiene l'accesso alla flag, la quale può essere sottoposta a un sistema di punteggio che lo premia con un certo numero di punti. Il numero può variare a seconda delle regole della competizione e può essere influenzato da:

- un punteggio di base per la sfida, decisa dagli organizzatori in base alla difficoltà della sfida;

- il tempo impiegato per risolvere la sfida, con le varie sfide che diminuiscono di valore con il passare del tempo e con la risoluzione degli altri partecipanti;

- il numero di squadre che ha risolto la sfida, con un punteggio dinamico deciso sulla base di chi risolve la sfida per primo, per secondo o per ultimo.

Questa tipologia permette un ottimo coinvolgimento dei partecipanti a tutti i livelli, poiché ognuno può risolvere una sfida indipendentemente dagli altri e non si risulta bloccati nel caso di difficoltà. L'ampia varietà di categorie e difficoltà garantisce che tutti i partecipanti siano soddisfatti, indipendentemente del background o dal libello di abilità.

Però, a differenza degli altri formati, le competizioni Jeopardy sono più gamificate e meno rappresentative della realtà. Quando le squadre partecipano a questi contest è possibile che ogni membro affronti diverse categorie di sfide, rendendo la cooperazione meno centrale e riducendo l'interazione tra i partecipanti.

Alcuni esempi di Jeopardy CTF sono:

1. Google CTF[12]: Google tiene la sua CTF una volta all'anno e quella del 2021 consisterà in due parti:

   - Una competizione online per chiunque voglia partecipare;
   - Una seconda competizione aperta solo alle prime 16 squadre vincitrici del primo evento;

   Le sfide sono di varia natura e coinvolgono il reverse engineering, l'analisi della memoria, la crittografie e molte altre. Come in una classica Jeopardy CTF, quando i giocatori risolvono una sfida ottengono un bandiera che può essere verificata dal sistema di punteggio e far guadagnare punti alla squadra che di conseguenza sale di classifica.

2. PicoCTF[13]: È un CTF sviluppata dalla Carnegie Mello University con una struttura classica. Le squadre si sfidano per risolvere sfide di sicurezza informatica, cercando le flag nascoste nei server, a volte offuscate in programmi binari a volte in testi cifrati. Durante la competizione, che in genere dura da uno a due giorni, le squadre guadagnano punti dopo aver trovato le flag. Le sfide sono generalmente progettate con diverse soluzioni possibili e possono essere di computer forensics, crittografie, reverse engineering e sicurezza web;

3. Macchine virtuali vulnerabili: sebbene non appartenenti alla classica categoria Jeopardy, su internet sono disponibili diverse macchine virtuali vulnerabili (un esempio è Vulnhub[14]) le quali, una volta scaricate e avviate, contengono all'interno una sequenza di sfide da risolvere in ordine che permettono al giocatore di, tramite una serie di vulnerabilità, ottenere il possesso del sistema a livello di amministratore.

Attacco e Difesa

Nel format Attacco Difesa, alle squadre viene dato accesso a una serie di host target, e l'obiettivo di ogni squadra è quello di prendere e mantenere il controllo sul maggior numero possibile di host. Per fare ciò, gli organizzatori della sfida creano

[12] `https://capturetheflag.withgoogle.com/`.
[13] `https://picoctf.org/`.
[14] `https://www.vulnhub.com/`.

una serie di servizi vulnerabili e le squadre devono bilanciare sia la necessità di attaccare altri host, conquistarli e guadagnare punti, sia applicare patch e cercare di proteggere gli host che hanno conquistato, per impedire alle altre squadre di comprometterli.

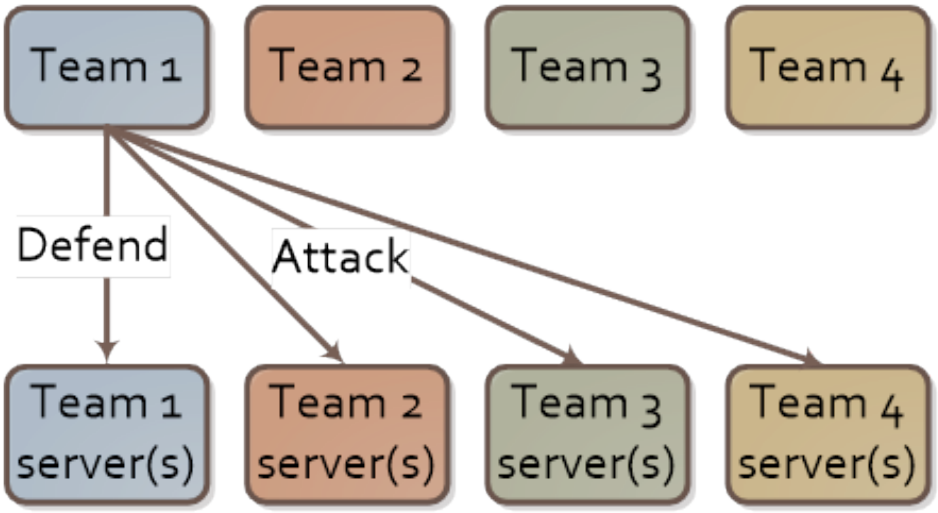


Figura 25: Esempio di architettura Attack-Defense

Questo format è stato uno dei primi, creato dalle attività di wargame nelle comunità militari e di hacker, e anche uno dei più complessi da organizzare, vista la quantità e il livello di dettagli da tenere in conto.

I partecipanti hanno quindi il compito di compromettere con successo (e poi mettere in sicurezza) dei server bersaglio, i quali contengono uno o più servizi vulnerabili, che possono essere reali o create ad hoc dagli organizzatori.

Le squadre dopo aver conquistato un host, devono riuscire a mantenerne il controllo, cioè eseguire azioni difensive come la mitigazione delle vulnerabilità, l'applicazione di path per aggiornare lo stack applicativo o infrastrutturale o l'abilitazione di misure generali come la creazione di firewall, il rafforzamento delle password o la disabilitazione di certi servizi.

Solitamente il punteggio viene assegnato a intervalli regolari, con ogni squadra che riceve un certo numero di punti per ogni host che controlla in quel momento.

In questo modo i partecipanti sono incoraggiati a compromettere e successivamente a proteggere i server controllati. Le regole della competizione richiedono anche che certi servizi rimangano attivi per poter assegnare i punti, per evitare che le squadre disabilitino tutti i servizi per guadagnarsi la vittoria.

Questo format offre un modello molto pratico di scenari di sicurezza reali, con i partecipanti che acquisiscono esperienza sia nel red teaming che nel blue teaming. Queste competizioni sono meno gamificate rispetto alle precedenti, più adatte a

degli spettatori ed eventi dal vivo, poiché è possibile assistere al cambiamento del controllo dei server durante l'evento.

I partecipanti sono più stimolati, in quanto devono gestire gli obiettivi, che spaziano dall'identificazione di vulnerabilità nei servizi dei target, nella compromissione degli stessi e nel proteggere quelli già conquistati.

Questi eventi sono però più scoraggianti per i meno esperti e richiedono un livello di esperienza e skill notevole. Un altro punto debole è nel caso di squadre molto sbilanciate, in quanto nel caso ci sia una squadra troppo capace, in poco tempo si guadagnerebbe la maggior parte degli host, renderli quasi impenetrabili lasciando le altre squadre senza obiettivi da raggiungere.

Un esempio di questa tipologia è CISCO SECCON AD-CTF[15], organizzata da CISCO e tenutasi nel 2020.

4.2.3 Juice Shop

A favore del lavoro svolto, si è scelto di utilizzare la piattaforma Juice Shop[16].

Uno dei progetti di punta della fondazione OWASP, Juice è una delle più moderne e sofisticate applicazioni web, sviluppata appositamente in maniera insicura per permetterne l'utilizzo in corsi di formazione sulla sicurezza, CTF o come esercizio per studenti di Cyber Security. Essa comprende tutte le vulnerabilità della OWASP Top Ten, insieme ad altre vulnerabilità.

È scritta in Node.js, Express e Angular, comprende un notevole numero di sfide di sicurezza web di varia difficoltà che l'utente deve superare e il progresso delle sfide superate viene tracciato su di un tabellone.

È un'applicazione implementata completamente in Javascript e TipeScript. Nel frontend il framework Angular[17] è usato per creare una Single Page Application e utilizza Angular Flex per ottenere la dinamicità delle pagine.

Javascript viene usato anche nel backend come linguaggio di programmazione dove una applicazione Express ospitata in un server Node.js[18] fornisce il codice lato client al browser e rende disponibili le funzionalità di backend necessarie al client tramite APIs RESTful. Come database viene invece utilizzato un SQLite, grazie alla sua semplicità e alla sua natura basata su file.

Come ulteriore database, viene utilizzato anche MarsDB, un derivato Javascript del database NoSQL MongoDB, ampiamente utilizzato e compatibile con la maggior parte delle operazioni di query.

Quando una sfida viene superata con successo viene mostrata una notifica push all'utente, implementata tramite il protocollo WebSocket. L'applicazione

[15] `https://medium.com/csictf/cisco-seccon-2020-ad-ctf-2614b27f387a`.
[16] `https://owasp.org/www-project-juice-shop/`.
[17] `https://angularjs.org/`.
[18] `https://nodejs.org/it/`.

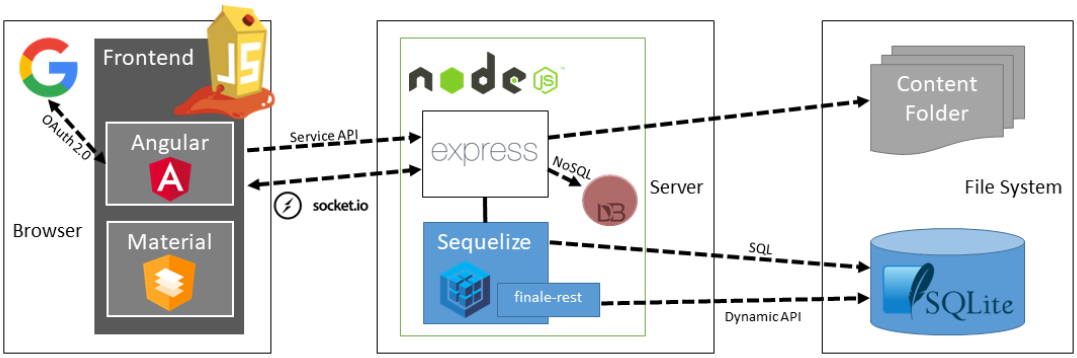


Figura 26: Architettura di Juice Shop

offre anche una comoda registrazione utente tramite Oauth 2.0, in modo che gli utenti possano accedere con il loro account Google.

L'utilizzo di Juice Shop per creare un Cyber Range ha diversi vantaggi, tra i quali:

1. Open Source: essendo licenziato sotto licenza MIT, non ha costi ed è disponibile liberamente;

2. Semplice da deployare e da installare;

3. Di facile utilizzo per chiunque lo usi e con sfide di diversi gradi di difficoltà;

4. Possibilità di brandizzarlo in base al contesto in cui ci si trova, cambiarne i contenuti o le immagini;

5. Gamificazione: l'applicazione notifica all'utente le sfide risolte e tiene traccia delle vulnerabilità sfruttare con successo su una dashboard privata;

6. Interoperabilità: integrazione con diversi sistemi di formazione tramite WebHook.

Le vulnerabilità presenti in Juice sono categorizzate in diverse classi. La maggior parte copre diversi tipi di rischio come la OWASP Top 10, la Common Weakness Enumeration del MITRE[19] e la Web Application Security Consortium Threat Classification (WASC)[20].

[19] `https://cwe.mitre.org/`.
[20] `http://projects.webappsec.org/w/page/13246978/Threat%20Classification`.

Categoria	OWASP	CWE	WASC
Broken Access Control	A5:2017, API1:2019, API5:2019	CWE-22, CWE-285, CWE-639	WASC-02, WASC-09, WASC-16
Broken Anti-Automation	OWASP-AT-004, API4:2019, OWASP-AT-010, OAT-009, OAT-015, OAT-008	CWE-362	WASC-11, WASC-21
Broken Authentication	A2:2017, API2:2019, P6:2021	CWE-287, CWE-352	WASC-01, WASC-49
Cross Site Scripting (XSS)	A7:2017	CWE-79	WASC-8
Cryptographic Issues	A3:2017	CWE-326, CWE-327, CWE-328, CWE-950	-
Improper Input Validation	ASVS V5, API6:2019	CWE-20	WASC-20
Injection	A1:2017, API8:2019, P1:2021	CWE-74, CWE-89	ASC-19, WASC-28, WASC-31
Insecure Deserialization	A8:2017	CWE-502	-
Miscellaneous	P5:2021	-	-
Security Misconfiguration	A6:2017, A10:2017, API7:2019, API9:2019, API10:2019	CWE-209	WASC-14, WASC-15
Security through Obscurity	P5:2021	CWE-656	-
Sensitive Data Exposure	A3:2017, API3:2019, OTG-CONFIG-004, P2:2021	CWE-200, CWE-530, CWE-548	WASC-13
Unvalidated Redirects	A10:2013	CWE-601	WASC-38
Vulnerable Components	A9:2017	CWE-829, CWE-506, CWE-1104	-
XML External Entities (XXE)	A4:2017	CWE-611	WASC-43

Per motivare gli studenti a trovare le vulnerabilità è disponibile inoltre una lista delle sfide in una pagina segreta (Figura 27), nella quale si possono vedere le sfide risolte, quelle mancanti e il livello di difficoltà. Nella stessa pagina è possibile avere istruzioni per alcune sfide, alcune molto dettagliate, altre decisamente vaghe.

Figura 27: Scoreboard di Juice Shop

OWASP Juice Shop può essere inoltre personalizzato nel suo inventario di prodotti e nel suo look & feel per soddisfare eventuali esigenze aziendali o universitarie, sostituendo il logo dell'applicativo o altri dettagli grafici.

La personalizzazione è permessa da un file di configurazione YAML posto nella cartella /config. Per poter eseguire una propria customizzazione è necessario:

- Aggiungere il proprio file di configurazione .yml in /config;
- Impostare una variabile d'ambiente NODE_ENV al nome del file di configurazione creato senza l'estensione .yml;
- Lanciare l'applicazione con il comando npm start.

Di seguito un esempio di configurazione yml, utilizzata da Mozilla Foundation per una sua CTF[21]:

```
application:
  domain: mozilla-ctf.op
  name: 'Mozilla CTF'
  logo: 'https://github.com/mozilla/ctf-austin/raw/master/
      app/public/images/MozillaCTF.png'
```

[21] https://hacks.mozilla.org/2018/03/hands-on-web-security-capture-the-flag-with-owasp-juice-shop/.

```
  favicon: 'https://github.com/mozilla/ctf-austin/raw/
      master/app/public/favicon_v2.ico'
  theme: deeporange-indigo
  showGitHubLinks: true
  localBackupEnabled: false
  altcoinName: Mozquito
  privacyContactEmail: 'donotreply@mozilla-ctf.op'
  customMetricsPrefix: mozctf
  chatBot:
    name: 'Foxy'
    avatar: 'https://upload.wikimedia.org/wikipedia/commons
        /9/9f/Fennec_Fox_Vulpes_zerda.jpg'
  social:
    twitterUrl: 'https://twitter.com/mozcloudsec'
    facebookUrl: null
    slackUrl: null
    redditUrl: null
    pressKitUrl: 'https://blog.mozilla.org/press/kits'
    questionnaireUrl: null
  recyclePage:
    topProductImage: Gear-200155340.jpg
    bottomProductImage: Gear-200155753.jpg
  welcomeBanner:
    showOnFirstStart: false
  cookieConsent:
    backgroundColor: '#e95420'
    textColor: '#ffffff'
    buttonColor: '#2778c5'
    buttonTextColor: '#ffffff'
    message: 'This website uses a myriad of 3rd-party
        cookies for your convenience and tracking pleasure.'
    dismissText: 'Never mind!'
    linkText: 'How can I turn this off?'
    linkUrl: 'https://support.mozilla.org/en-US/kb/disable-
        third-party-cookies'
  securityTxt:
    contact: 'mailto:donotreply@mozilla-ctf.op'
    encryption: ~
challenges:
  showHints: false
```

```
  xssBonusPayload: '<iframe width="560" height="315" src="
      https://www.youtube.com/embed/0u9g0LhRkGk" frameborder
      ="0" allow="accelerometer; autoplay; encrypted-media;
      gyroscope; picture-in-picture" allowfullscreen></
      iframe>'
hackingInstructor:
  isEnabled: false
products:
  -
    name: 'Champion Sweatshirt with a Drawstring Tote'
    price: 68.89
.....
```

Un'altra funzionalità molto interessante di Juice Shop è quella di poter essere esportata in un formato speciale che permette di importarla successivamente in certe piattaforme di punteggio, come CTFD[22], FBCTF[23] e RootTheBox[24] (Figura 28).

Tramite il tool juice-shop-cli[25] è possibile, sia in modalità guidata che in modalità non interattiva, esportare tutte le sfide presenti sulla piattaforma, già divise per categoria, in formato zip e poterle poi importare in una piattaforma di scoring in modo da avere già le sfide, le flag e il punteggio preconfigurato.

```
kali@kali:~/Downloads/juice-shop$ juice-shop-ctf

Generate OWASP Juice Shop challenge archive for setting up CTFd, FBCTF or RootTheBox score server
? CTF framework to generate data for? CTFd
? Juice Shop URL to retrieve challenges? https://juice-shop.herokuapp.com
? Secret key <or> URL to ctf.key file? https://raw.githubusercontent.com/bkimminich/juice-shop/master/ctf.key
? Insert a text hint along with each challenge? No text hints
? Insert a hint URL along with each challenge? No hint URLs

Backup archive written to /home/kali/Downloads/juice-shop/OWASP_Juice_Shop.2021-06-24.CTFd.zip

After the import you will have to set up the CTF name and administrator credentials again!

For a step-by-step guide to import the ZIP-archive into CTFd, please refer to
https://pwning.owasp-juice.shop/part1/ctf.html#running-ctfd
```

Figura 28: Export di Juice Shop

CTFd

Juice Shop supporta i CTF in stile Jeopardy generando un codice per ogni sfida risolta. Per abilitarlo è necessario inserire nel file YAML la variabile

[22] https://ctfd.io/.
[23] https://github.com/facebookarchive/fbctf.
[24] https://github.com/moloch--/RootTheBox.
[25] https://github.com/juice-shop/juice-shop-ctf.

```
ctf:
  showFlagsInNotifications: true
```

in modo che ogni volta che viene risolta una sfida appaia un pop-up con la flag e la si possa inserire in modo manuale all'interno della piattaforma di scoring.

Una volta impostata la modalità CTF, è possibile esportare le varie flag con lo strumento juice-shop-ctf-cli e importarle in una delle piattaforme più utilizzate per lo scoring durante i CTF, come a esempio CTFD.

CTFD è un framework CTF (Figura 29) progettato per la facilità d'uso sia per gli organizzatori sia per gli utenti. Esso permette di creare le proprie sfide, categorie, suggerimenti e flag all'interno dell'interfaccia di amministrazione.

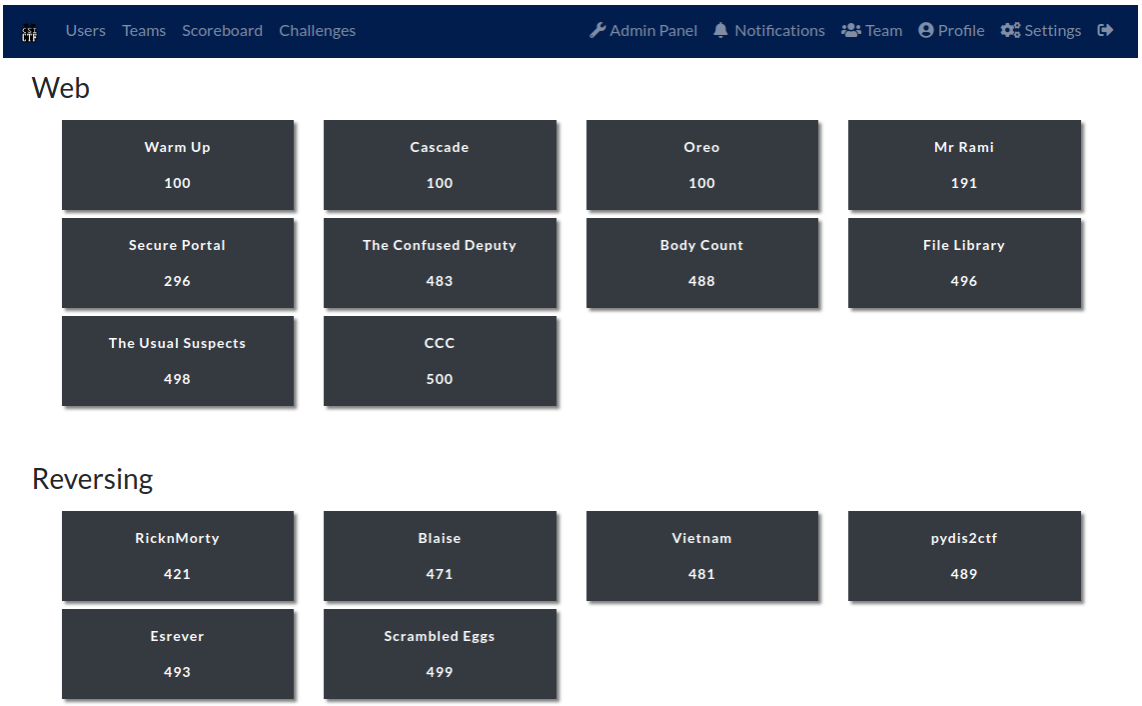


Figura 29: Piattaforma CTFD

Alcune delle feature sono:

- Sfide con punteggio dinamico;
- Flag statiche e basate su regex;
- Suggerimenti sbloccabili per gli utenti;
- Caricamento di file sul server per sfide come reversing o binary exploitation;

- Protezione automatica contro attacco bruteforce;
- Competizioni individuali o a squadre;
- Grafici per la verifica del punteggio tra i vari utenti;
- Supporto email con SMTP;
- Personalizzazione della grafica;
- Importazione ed esportazione di dati.

Per importare una lista di sfide in CTFD è quindi sufficiente andare in Impostazioni - Import ed effettuare l'upload dell'archivio creato precedentemente con juice-shop-ctf, come mostrato in Figura 30).

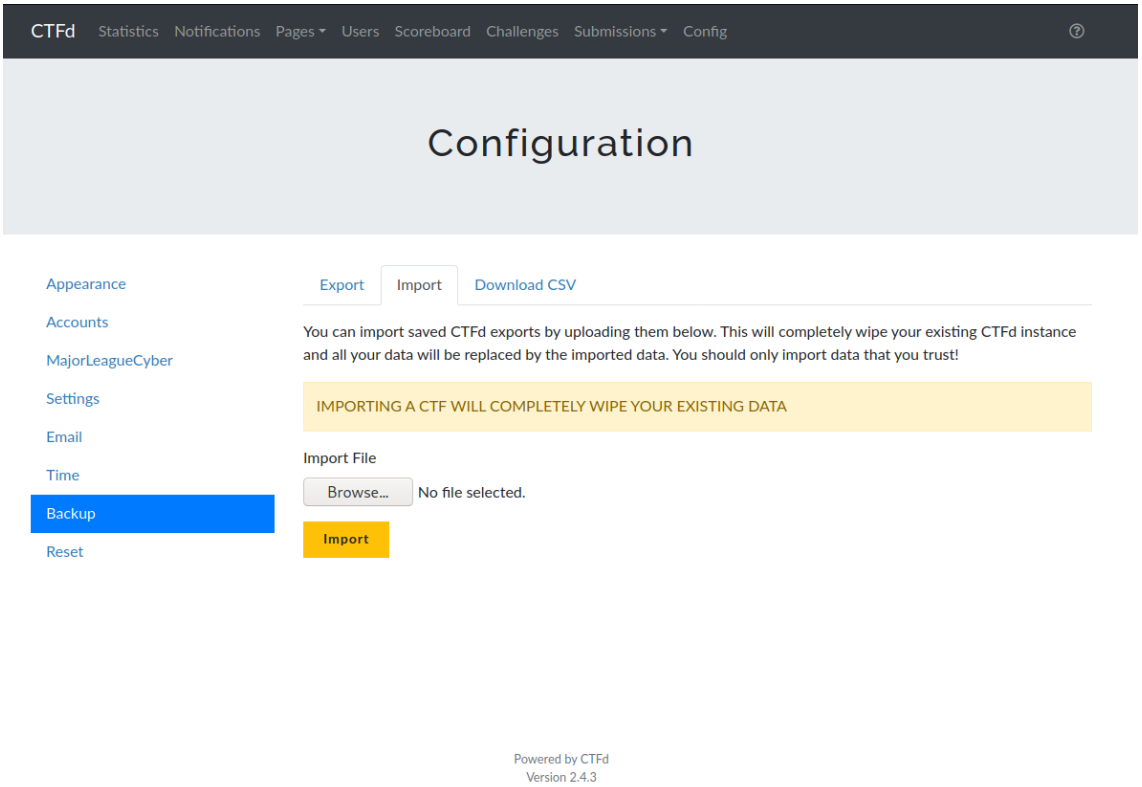


Figura 30: Import di sfide in CTFD

4.3 Generazione Model Driven di scenari

Juice Shop è quindi una delle piattaforme più versatili e moderne presenti a oggi tra quelle libere di essere scaricate. Ma ha però diversi svantaggi nel suo utilizzo in certe situazioni, come possono essere quelle di formazione in azienda o durante un esercitazione di un corso universitario, in quanto non permette di essere gestita da più utenti, ma è piuttosto una piattaforma a uso individuale, dove ogni utente sfida se stesso per risolvere quante più sfide.

Un'altro svantaggio in una situazione di formazione è quello che non può essere deployato scegliendo determinate sfide, ma sono attive globalmente, senza possibilità di disattivarne nessuna.

4.3.1 Suddivisione in categorie

Ciò che si è voluto fare come prima modifica è quindi la suddivisione in categorie, permettendo l'esecuzione di Juice Shop sulla base di una categorizzazione simil Owasp Top 10, creando diversi file contenenti le sfide suddivise per categoria.

Per permettere all'utente di decidere la categoria di sfide da inserire è stato necessario modificare il core dell'applicazione, inserendo nuove variabili e condizioni per filtrare le categorie ed eseguire solo quelle selezionate. Un esempio delle modifiche effettuate è in Figura 31.

Queste modifiche hanno permesso di inserire una nuova variabile nel file YAML caricato all'avvio dell'applicazione, in modo che possa essere decisa la categoria delle sfide (Figura 32).

In questo modo l'utente per eseguire solo le sfide con le quali vorrà esercitarsi, potrà eseguire Juice Shop con la variabile d'ambiente NODE_ENV modificata (Figura 33):

```
NODE_ENV=Categoria npm start
```

Una volta effettuate queste modifiche, si è passati alla risoluzione del problema dell'utilizzo contemporaneo da parte di più utenti, dove da una parte c'è il trainer, ovvero l'istruttore o il docente che vuole effettuare una determinata esercitazione e dall'altra ci sono i trainee, ossia gli studenti che devono esercitarsi.

4.3.2 Creazione di più istanze ed esecuzione contemporanea

Per risolvere il problema dell'individualismo di Juice Shop, si è pensato a un deploy automatico, contemporaneo e multiplo di più istanze di Juice Shop sulla base del numero di trainee, con un'istanza gestita dal trainer con cui possa monitorare

```
const challenges = await loadStaticData('challenges')

await Promise.all(
  challenges.map(async ({ name, category, description, difficulty, hint, hintUrl, mitigationUrl, key, disabledEnv, tutorial, tags }) => {
    const effectiveDisabledEnv = utils.determineDisabledEnv(disabledEnv)
    description = description.replace('juice-sh.op', config.get('application.domain'))
    description = description.replace('&lt;iframe width=&quot;100%&quot; height=&quot;166&quot; scrolling=&quot;no&quot; frameborder=&quot;no&
    quot; allow=&quot;autoplay&quot; src=&quot;https://w.soundcloud.com/player/?url=https%3A//api.soundcloud.com/tracks/771984076&amp;
    color=%23ff5500&amp;auto_play=true&amp;hide_related=false&amp;show_comments=true&amp;show_user=true&amp;show_reposts=false&amp;
    show_teaser=true&quot;&gt;&lt;/iframe&gt;', entities.encode(config.get('challenges.xssBonusPayload')))
    hint = hint.replace(/OWASP Juice Shop's/, `${config.get('application.name')}'s`)
    try {
      if (config.get('challenges.cat') != "*") {
        if (category == config.get('challenges.cat')) {
          logger.info(`Challenge: ${name} `)
          datacache.challenges[key] = await models.Challenge.create({
            key,
            name,
            category,
            tags: tags ? tags.join(',') : undefined,
            description: effectiveDisabledEnv ? (description + ' <em>(This challenge is <strong>' + (config.get('challenges.safetyOverride') ?
            'potentially harmful' : 'not available') + '</strong> on ' + effectiveDisabledEnv + '!)</em>') : description,
            difficulty,
            solved: false,
            hint: showHints ? hint : null,
            hintUrl: showHints ? hintUrl : null,
            mitigationUrl: showMitigations ? mitigationUrl : null,
            disabledEnv: config.get('challenges.safetyOverride') ? null : effectiveDisabledEnv,
            tutorialOrder: tutorial ? tutorial.order : null
          })
        }
      } else {
        logger.info(`Challenge: ${name} `)
          datacache.challenges[key] = await models.Challenge.create({
            key,
            name,
            category,
            tags: tags ? tags.join(',') : undefined,
```

Figura 31: Una delle modifiche effettuate

```
kali@kali:~/Downloads/juice-shop/config$ ls
A1_Injection.yml    A4_XXE.yml                    A7_XSS.yml                      ALL.yml                              CWE-362_Broken_Anti-Automation.yml
A2_Broken_Auth.yml  A5_Broken_Access_Control.yml  A8_Insecure_Deserialization.yml CWE-200_Sensitive_Data_Exposure.yml  CWE-601_Unvalidated_Redirects.yml
A3_Sensitive.yml    A6_Security_Misconfiguration.yml A9_Vulnerable_Components.yml CWE-20_Improper_Input_Validation.yml CWE-656_Security_through_Obscurity.yml
kali@kali:~/Downloads/juice-shop/config$
kali@kali:~/Downloads/juice-shop/config$ cat A1_Injection.yml | grep cat: -A3 -B1
  safetyOverride: false
  cat: 'Injection'
hackingInstructor:
  isEnabled: true
  avatarImage: JuicyBot_MedicalMask.png
kali@kali:~/Downloads/juice-shop/config$
```

Figura 32: File YAML suddivisi per categoria

il livello dei trainee, i risultati raggiunti e identificare possibili problemi durante l'esercitazione.

Gli step effettuati dal trainer sono i seguenti:

1. Esegue su un server la sua copia di Juice Shop con la categoria prescelta per l'esercitazione;

2. Esporta, tramite l'utilizzo di juice-shop-ctf, l'archivio contenente le sfide e le flag;

3. Importa all'interno di CTFD l'archivio in modo che le sfide siano pronte per essere verificate.

Questo passaggio può essere eseguito anche in maniera automatica da uno script, che esegue Juice Shop, esporta le sfide ed importa l'archivio in CTFD.

```
kali@kali:~/Downloads/juice-shop$ NODE_ENV=A7_XSS npm start

> juice-shop@12.1.0 start /home/kali/Downloads/juice-shop
> node app

info: All dependencies in ./package.json are satisfied (OK)
info: Chatbot training data botDefaultTrainingData.json validated (OK)
{ locale: 'en', useNoneFeature: undefined, trainByDomain: undefined }
info: Detected Node.js version v12.19.0 (OK)
info: Detected OS linux (OK)
info: Detected CPU x64 (OK)
info: Configuration A7_XSS validated (OK)
info: Required file index.html is present (OK)
info: Required file styles.css is present (OK)
info: Required file tutorial-es2018.js is present (OK)
info: Required file main-es2018.js is present (OK)
info: Required file polyfills-es2018.js is present (OK)
info: Required file runtime-es2018.js is present (OK)
info: Required file vendor-es2018.js is present (OK)
info: Required file main-es5.js is present (OK)
info: Required file tutorial-es5.js is present (OK)
info: Required file polyfills-es5.js is present (OK)
info: Required file runtime-es5.js is present (OK)
info: Required file vendor-es5.js is present (OK)
info: Port 3000 is available (OK)
info: Challenge: API-only XSS
info: Challenge: CSP Bypass
info: Challenge: Client-side XSS Protection
info: Challenge: DOM XSS
info: Challenge: HTTP-Header XSS
info: Challenge: Reflected XSS
info: Challenge: Server-side XSS Protection
info: Challenge: Video XSS
info: Challenge: Bonus Payload
info: Server listening on port 3000
```

Figura 33: Juice eseguita solo con le sfide Cross Site Scripting

Successivamente, i trainee eseguono la propria macchina virtuale, opportunamente configurata in modo da raggiungere l'applicazione CTFD hostata sul server del trainer.

La macchina virtuale è stata configurata in modo tale che, una volta accesa esegua la registrazione in maniera automatica a CTFD, senza che il trainee debba fare nulla.

Il seguente script, scritto in python e chiamato registertoCTFD.py viene eseguito all'avvio della macchina virtuale ed effettua la registrazione a CTFD (variabile host_trainer) utilizzando una password generata in maniera pseudo casuale e come username l'hostname della macchina virtuale.

```
import requests
import re
import socket
import secrets
import string

host_trainer = "192.168.1.62"
```

```
port = "4000"
hostname = socket.gethostname()
alphabet = string.ascii_letters + string.digits
password = ''.join(secrets.choice(alphabet) for i in range
    (20))
f= open("/tmp/password_juice","w+")
f.write(password)
f.close()


get_register_ctfd = "http://" + host_trainer + ":" + port +
    "/register"
headers_ctfd = {"User-Agent": "Mozilla/5.0 (X11; Linux
    x86_64; rv:78.0) Gecko/20100101 Firefox/78.0", "Accept":
    "text/html,application/xhtml+xml,application/xml;q=0.9,
    image/webp,*/*;q=0.8", "Accept-Language": "en-US,en;q
    =0.5", "Accept-Encoding": "gzip, deflate", "Connection":
    "close", "Upgrade-Insecure-Requests": "1"}
response = requests.get(get_register_ctfd, headers=
    headers_ctfd)
session = response.headers['Set-Cookie'].split("=")[1].
    split(";")[0]
grep = '\\\'csrfNonce\\\': "(.*)"'
result = re.search(grep, response.text)
csrfNonce = result.group(1)


create_user = "http://" + host_trainer + ":" + port + "/
    register"
cookies = {"session": session}

post_data = {"name": hostname, "email": hostname + "@email.
    com", "password": password, "nonce": csrfNonce}
response = requests.post(create_user, headers=headers_ctfd,
    cookies=cookies, data=post_data)
if response.status_code != 200:
    print("Error!")
```

Di conseguenza, l'utente è stato registrato a CTFD senza che debba fare alcun passaggio manuale.

Superata la registrazione, è stato necessario integrare CTFD, ospitato sul server

del trainer, con le varie VM ospitate sui computer dei trainee.

Per fare ciò, è stato modificato Juice Shop (Figura 34) al fine che, ogni volta che un utente risolva una sfida, oltre che apparire la notifica di avvenuta risoluzione, venga eseguita una chiamata a CTFD inviando, oltre che le credenziali, la risoluzione della sfida all'API corrispondente.

```
lib > JS utils.js > sendNotification > sendNotification > then() callback > then() callback > then() callback > then() callback > chlgs.forEach() callback
219                 new_nonce = r[1];
220                 axios.get("http://" + hostCTFD + ":" + portCTFD + "/api/v1/challenges" ,
221                 {
222                   headers : {
223                     "User-Agent": "Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0",
224                     "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*;q=0.8",
225                     "Accept-Language": "en-US,en;q=0.5",
226                     "Accept-Encoding": "gzip, deflate",
227                     "Content-Type": "application/x-www-form-urlencoded",
228                     "Content-Length": "129",
229                     "Connection": "close",
230                     "Upgrade-Insecure-Requests": "1",
231                     Cookie: session
232                   }
233                 })
234               .then(function (response) {
235                 chlgs = response.data.data
236                 chlgs.forEach(function(value){
237                   if (value.name == challenge.name) {
238                     axios.post("http://" + hostCTFD + ":" + portCTFD + "/api/v1/challenges/attempt", {
239                     challenge_id: value.id,
240                     submission: flag
241                   },
242                   {
243                     headers: {
244                     "User-Agent": "Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0",
245                     "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*;q=0.8",
246                     "Accept-Language": "en-US,en;q=0.5",
247                     "Accept-Encoding": "gzip, deflate",
248                     "Content-Type": "application/json",
249                     "Content-Length": "129",
250                     "Connection": "close",
251                     "Upgrade-Insecure-Requests": "1",
252                     "CSRF-Token": new_nonce,
253                     Cookie: session
254                     }
255                   })
256                     .then(function (response) {
257                       // handle success
258                       console.log("Inviata flag!");
259                     })
```

Figura 34: Richiesta POST inviante le flag a CTFD

In questo modo si è riusciti ad automatizzare, generando in maniera automatica le utenze per CTFD, il punteggio dei trainee su CTFD e il trainer ha la possibilità di valutare in tempo reale l'andamento dell'esercitazione.

Per effettuare un test, si sono eseguite quattro macchine virtuali per quattro ipotetici trainee mentre è stata eseguita CTFD sul server del trainer.

Una volta completata la configurazione del server del trainer, i quattro trainee hanno potuto esercitarsi a Juice Shop in contemporanea, ottenendo sia i benefici di una classica esercitazione sia i benefici di una vera e propria CTF.

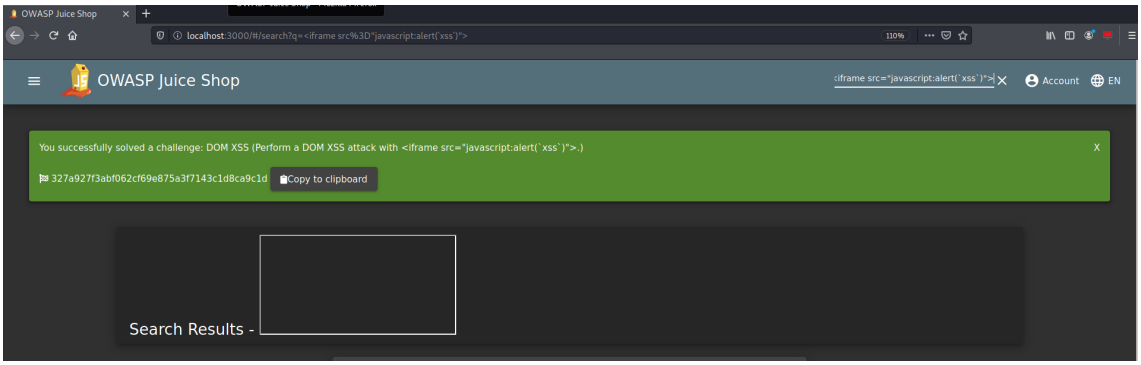


Figura 35: Sfida risolta da parte dell'utente

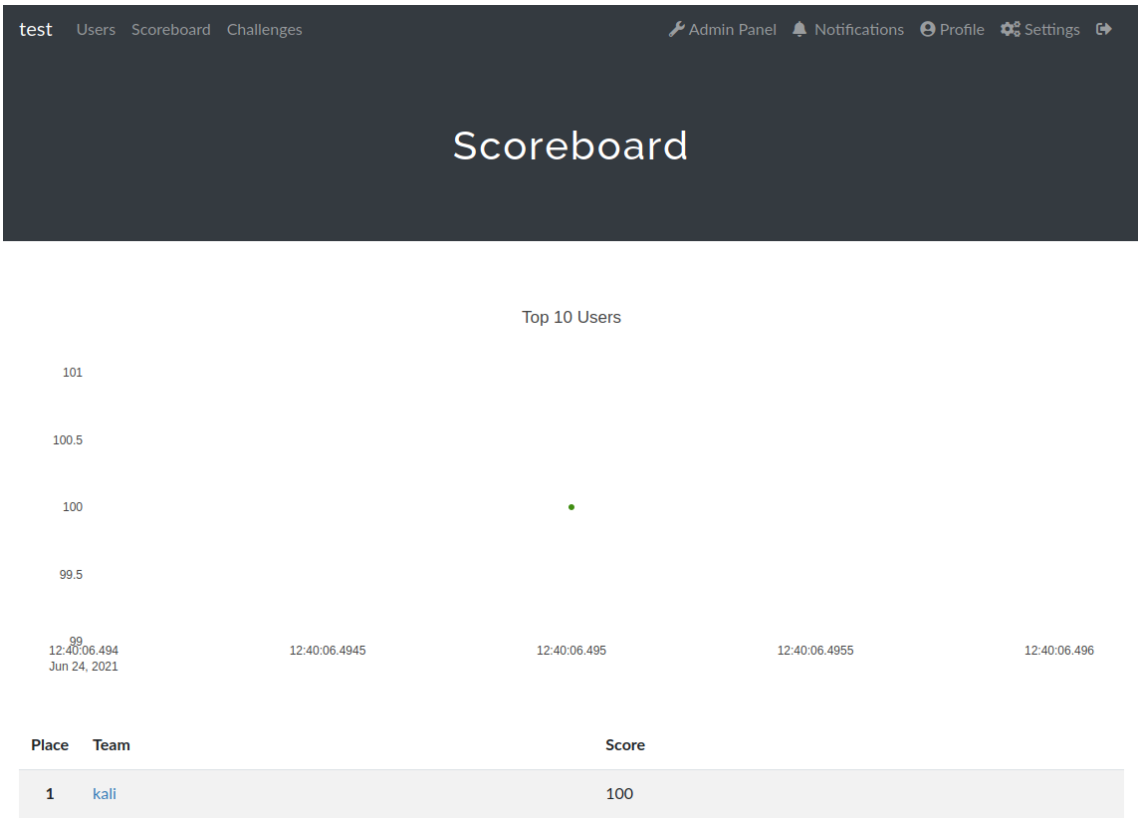


Figura 36: Sfida risolta da parte dell'utente anche su CTFD

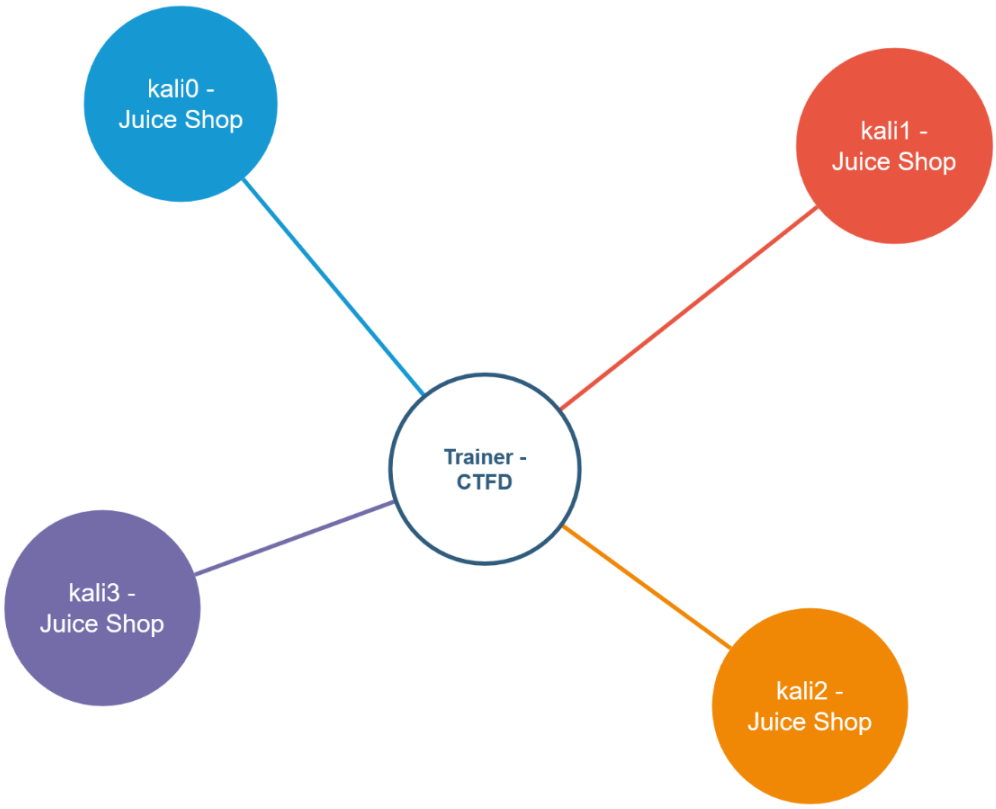
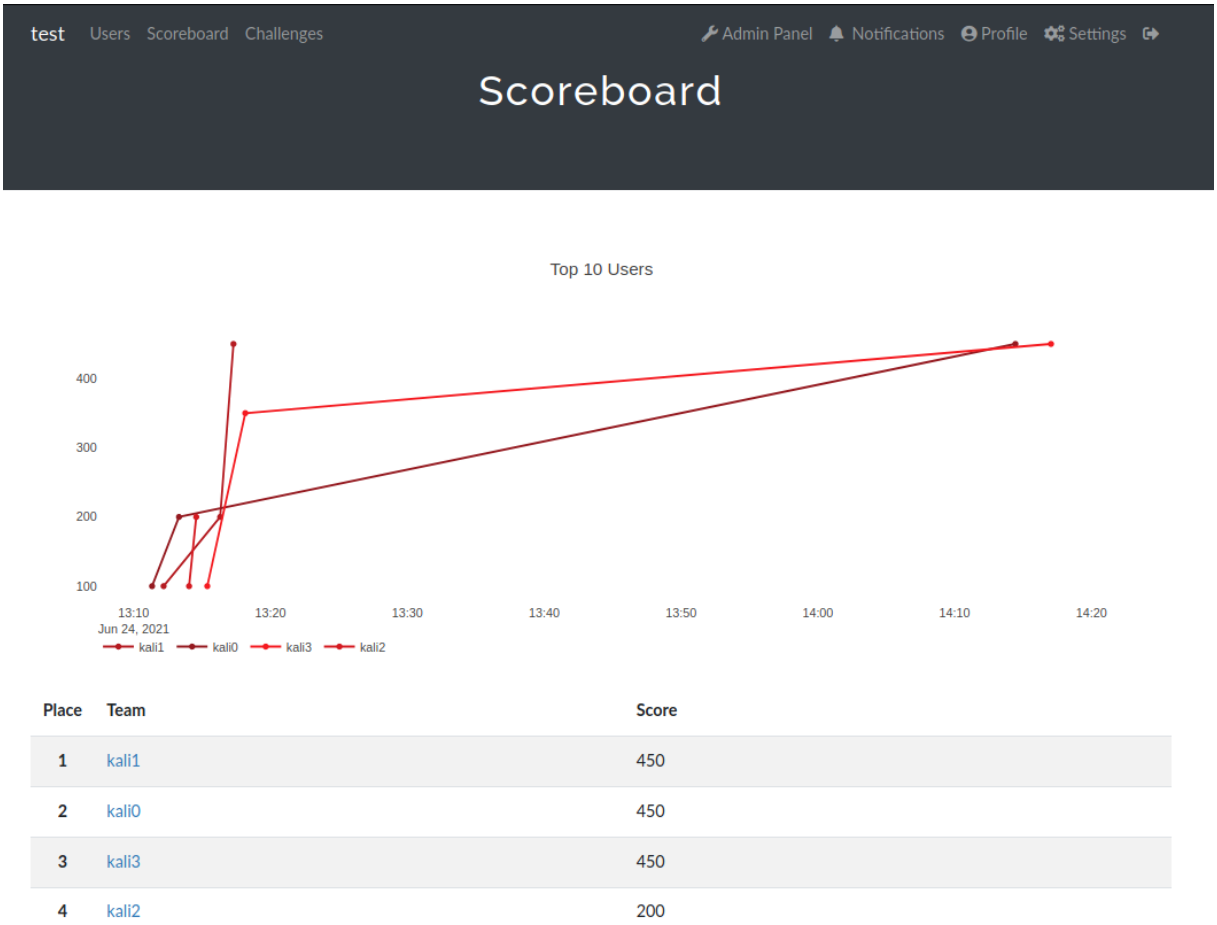


Figura 37: Esercitazione tra 4 Trainee e un Trainer



Place	Team	Score
1	kali1	450
2	kali0	450
3	kali3	450
4	kali2	200

Figura 38: Esempio di esercitazione tra quattro Trainee e la vista del Trainer

Capitolo 5

Conclusioni

In questo documento è stato presentato un approccio Model Driven creato partendo da una piattaforma liberamente scaricabile, modificandola e integrandola con un sistema di scoring in maniera completamente automatizzata, in modo da rendere pressoché nullo il costo in termini di tempo di configurazione per gli utilizzatori.

Questo approccio ha permesso di definire le problematiche presenti, come la mancanza di possibilità di scelta delle sfide e la mancanza di integrazione con altri utenti e di risolverle permettendo l'utilizzo della piattaforma da parte di università e aziende con possibilità di scelta delle sfide sulla base della OWASP Top 10.

5.1 Sviluppi futuri

Sebbene sia pronta per essere utilizzata, la piattaforma sia pronta all'utilizzo, per il futuro si potrebbe valutare un ulteriore modifica, che preveda l'ampliamento della stessa, sia dal punto di vista della configurazione che da quello dell'analisi di dati durante un'esercitazione.

Per quanto concerne l'aspetto della configurazione, essa potrebbe essere implementata all'interno di THREAT- ARREST, promuovendo una gestione completamente automatizzata ed un deploy ancor più semplificato.

Infine, per l'analisi di dati durante un'esercitazione, si potrebbe valutare l'implementazione di un sistema di scoring dinamico, basato sull'analisi del traffico generato, ai dump di memoria e agli strumenti utilizzati dai trainee. In tal modo si potrebbe avere una valutazione completa del comportamento di un trainee e identificare eventuali punti deboli.

Bibliografia

[1] Marco Barros Lourenço (ENISA) Christos Douligeris, Omid Raghimi and Louis Marinos. Enisa threat landscape - the year in review. https://www.enisa.europa.eu/publications/year-in-review. Accessed: 2021-05-10.

[2] FireEye. Highly evasive attacker leverages solarwinds supply chain to compromise multiple global victims with sunburst backdoor. https://www.fireeye.com/blog/threat-research/2020/12/evasive-attacker-leverages-solarwinds-supply-chain-compromises-\with-sunburst-backdoor.html. Accessed: 2021-05-10.

[3] Department of Financial Services. Twitter investigation report. https://www.dfs.ny.gov/Twitter_Report. Accessed: 2021-05-10.

[4] Lawrence Abrams. Over 500,000 zoom accounts sold on hacker forums, the dark web. https://www.bleepingcomputer.com/news/security/over-500-000-zoom-accounts-sold-on-hacker-forums-the-dark-web/. Accessed: 2021-05-10.

[5] European Cyber Security Organisation. Understanding cyber ranges: From hype to reality. https://www.ecs-org.eu/documents/uploads/understanding-cyber-ranges-from-hype-to-reality.pdf. Accessed: 2021-05-10.

[6] Shabeer Ahmad, Nicolò Maunero, and Paolo Prinetto. EVA: A Hybrid Cyber Range. In *Proceedings of the Fourth Italian Conference on Cyber Security, Ancona, Italy, February 4th to 7th, 2020*, pages 12–23, 2020.

[7] C. Cowan, S. Arnold, S. Beattie, C. Wright, and J. Viega. Defcon capture the flag: defending vulnerable code from intense attack. In *Proceedings DARPA Information Survivability Conference and Exposition*, volume 1, pages 120–129 vol.1, 2003.

[8] Kai Huotari and Juho Hamari. Defining gamification: A service marketing perspective. In *Proceeding of the 16th International Academic MindTrek Conference*, MindTrek '12, page 17–22, New York, NY, USA, 2012. Association for Computing Machinery.

[9] Jan Vykopal, Valdemar Švábenský, and Ee-Chien Chang. Benefits and pitfalls of using capture the flag games in university courses. *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, Feb 2020.

[10] NICE. The cyber range: A guide. `https://www.nist.gov/system/files/documents/2020/06/25/The%20Cyber%20Range%20-%20A%20Guide%20%28NIST-NICE%29%20%28Draft%29%20-%20062420_1315.pdf`. Accessed: 2021-05-10.

[11] Nestoras Chouliaras, George Kittes, Ioanna Kantzavelou, Leandros Maglaras, Grammati Pantziou, and Mohamed Amine Ferrag. Cyber ranges and testbeds for education, training, and research. *Applied Sciences*, 11(4), 2021.

[12] Barbara Filkins. It security spending trends. `https://www.sans.org/reading-room/whitepapers/leadership/security-spending-trends-36697`. Accessed: 2021-05-20.

[13] Mika Karjalainen, Tero Kokkonen, and Samir Puuska. Pedagogical aspects of cyber security exercises. In *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS PW)*, pages 103–108, 2019.

[14] Niklas Hätty. Representing attacks in a cyber range, 2019.

[15] Jon Davis and S. Magrath. A survey of cyber ranges and testbeds executive, 2013.

[16] Muhammad Yamin, Basel Katt, and Vasileios Gkioulos. Cyber ranges and security testbeds: Scenarios, functions, tools and architecture. *Computers & Security*, 88:101636, 10 2019.

[17] Evangelos C. Chaskos. Cyber-security training: A comparative analysis of cyber-ranges and emerging trends. `https://pergamos.lib.uoa.gr/uoa/dl/frontend/file/lib/default/data/2864976/theFile`. Accessed: 2021-06-20.

[18] JAMK. Report on existing cyber ranges, requirements. `https://cybersec4europe.eu/publications/deliverables/`. Accessed: 2021-06-20.

[19] Elochukwu Ukwandu, Mohamed Amine Ben Farah, Hanan Hindy, David Brosset, Dimitris Kavallieros, Robert Atkinson, Christos Tachtatzis, Miroslav Bures, Ivan Andonovic, and Xavier Bellekens. A review of cyber-ranges and test-beds: Current and future trends. *Sensors*, 20(24), 2020.

[20] D.C. Schmidt. Guest editor's introduction: Model-driven engineering. *Computer*, 39(2):25–31, 2006.

[21] Arie Deursen, Paul Klint, and Joost Visser. Domain-specific languages. *ACM SIGPLAN Notices*, 35:26–36, 06 2000.

[22] ENISA. Ctf events. `https://www.enisa.europa.eu/publications/ctf-events`. Accessed: 2021-06-10.