Sistema di monitoraggio e valutazione per Cyber Ranges

Alessandro Della Torre - 893181

November 28, 2020

Il lavoro in cui sono stato coinvolto riguarda un progetto, finanziato dalla Comunità Europea, con obiettivo la realizzazione di una Cyber Range: con questo termine si intende una piattaforma di training innovativa, nel quale il trainee ha a disposizione diversi scenari virtuali che in genere cercano di ricreare componenti tipici di alcune ambientazioni reali, quali società industriali, di servizi, impianti di produzione di energia o installazioni militari, dove fare pratica sia nella messa in sicurezza del sistema, sia nella ricerca di particolari vulnerabilità. Il progetto era in una fase avanzata riguardo alla realizzazione di ambienti simulati e al dispiegamento di scenari di training insieme alla raccolta delle azioni eseguite dal trainee durante tutto il percorso di apprendimento.
La mia tesi si è concentrata sulla creazione di una architettura volta alla ricezione delle informazioni già raccolte, alla loro elaborazione e salvataggio all'interno di specifici database.

Nelle piattaforme esistenti non viene fatto un monitoring realtime, bensì viene valutato soltanto se l'utente riesce a raggiungere obbiettivi in precedenza definiti. Avere un sistema che riesca a valutare tutte le azioni e gli errori svolti fornisce uno scenario chiaro di quali aspetti non sono chiari al trainee mettendo i trainer nella condizione di fornirgli un aiuto efficace.

Nella fase iniziale si è preferito un approccio meticoloso, che andasse a campionare gli strumenti già presenti all'interno del mercato che potessero assolvere i requisiti definiti precedentemente.

Dopo aver analizzato la tipologia di dati con cui si doveva operare e come questi fossero collegati, si è deciso di utilizzare un database a grafi chiamato Neo4j; la scelta di non utilizzare i database relazionali è dovuta alla maggiore flessibilità che questa soluzione mette a disposizione, insieme ad una maggiore somiglianza con le tipologie di strutture dati all'interno delle specifiche di progetto.
La raccolta delle informazioni all'interno delle macchine virtuali era già stata implementata attraverso applicativi specifici, definiti come agent: questo mi ha messo in condizione di concentrarmi maggiormente sulla parte di elaborazione delle informazioni e sulla loro correttezza rispetto ai metodi di raccolta.

Per quanto riguarda il trasporto delle informazioni inviate dagli agent, all'interno delle macchine client, all'applicazione si è optato per l'utilizzo di RabbitMQ in virtù della sua semplicità e affidabilità, garantita dal protocollo AMQP. All'interno di questa architettura mancava un layer che si occupasse dell'IO con il log manager e anche dell'iterazione con il Database a grafi; dopo un'approfondita ricerca, si è deciso di sviluppare un applicativo ad hoc, scritto in Java, che performasse le seguenti operazioni con i rispettivi layer.

Dalle premesse inizialmente poste si è riusciti a creare un applicativo che interagisce con tutte le componenti citate precedentemente e che assolve le seguenti mansioni: ricezione delle informazioni dalle code, elaborazione, inserimento nel database e ritorno dei grafi di training alle specifiche code di RabbitMQ. Partendo dai dati, è stato creato un database in Neo4j contenente le seguenti informazioni: template di azioni che l'utente dovrebbe svolgere, sottografo realtime di azioni effettivamente svolte dal trainee, nodi di contesto utilizzati per l'identificazione del trainee o degli scenari di training di cui è partecipe.
E' stato implementato un algoritmo di matching che controlla se l'azione compiuta dal trainee è contenuta nel template di azioni, relativo a quello specifico scenario di training, interno al database.
Sono state anche inserite funzionalità aggiuntive, alcune che permettono una gestione migliore dell'applicativo e altre che implementano delle specifiche elencate nell'analisi di progetto.
Una parte fondamentale riguarda la valutazione finale che deve essere assegnata al trainee e deve rispecchiare tutto il lavoro svolto: è stato implementato un sistema che si basa sulle differenze tra il cammino di azioni effettivamente compiute e il template definito precedentemente all'interno del sistema..