Relazione finale di Tirocinio

Corso di Laurea in Informatica

OGGETTO: Studio sull'efficienza di tool per l'analisi statica degli smart contract in Ethereum

Svolto presso: Alma Mater Studiorum - Università di Bologna

Svolto da:
Melania Ghelli
matr. 766608

Tutor accademico:
Prof. Ugo Dal Lago

1 Struttura ospitante e contesto del tirocinio

Il tirocinio è stato svolto presso l'Università di Bologna.
Dopo aver concordato con il docente Dal Lago l'argomento della tesi, abbiamo deciso di includere un'attività pratica nel progetto. Il lavoro svolto durante il tirocinio dunque è stato pensato come una base per la redazione del mio elaborato finale.

2 Attività svolta

L'obiettivo principale di questo tirocinio era quello di confrontare dei software per l'analisi statica.
Il progetto di tesi prevede uno studio delle attuali possibilità di analisi degli smart contract sulla piattaforma di Ethereum. Nello specifico ci siamo focalizzati sull'analisi mirata alla stima dei consumi di questi programmi in termini di gas.
In questo contesto, l'attività di tirocino ha previsto in gran parte una ricerca dei software attualmente disponibili che offrano la possibilità di condurre questo specifico tipo di analisi. Questa parte del lavoro è stata condotta principalmente attraverso la lettura di articoli scientifici, per avere un quadro generale delle ricerche svolte su questo tema.
Una volta individuato un certo numero di programmi volti ad analizzare smart contract, li ho confrontati fra di loro per capire quanti e quali di questi offrissero la possibilità di dare un bound esplicito ai consumi di gas. Durante questa fase si è reso necessario, talvolta, interfacciarsi direttamente con gli sviluppatori. Questo mi ha dato la possibilità di capire meglio i loro programmi, ma soprattutto di verificare se fra le funzionalità di questi ci fosse l'analisi del gas.
L'attività finale ha previsto alcuni test, che sono stati condotti sullo stesso insieme di smart contract utilizzando tool differenti. Questo mi ha permesso di ottenere dei risultati utili, che confrontati fra di loro mi permetteranno di fare alcune considerazioni sulla qualità dell'analisi di gas che è possibile fare.

3 Software utilizzati

Di seguito un breve riepilogo dei software con cui mi sono interfacciata durante l'attività di ricerca.
Sono tutti programmi pensati per fare analisi statica di smart contract per la piattaforma Ethereum. Molti di questi sono ancora in fase di sperimentazione, dunque non hanno una documentazione esaustiva. Per questa ragione si è reso necessario testarli prima di stabilire quali fossero utili ai fini della ricerca e quali no.
Tra quelli scartati:

EtherTrust è un software semantico che permette di analizzare smart contract per la verifica delle proprietà di sicurezza. Questo controllo astrae completamente dal gas, dunque non viene fatta alcuna stima dei suoi consumi;

MadMax è uno strumento di analisi statica utilizzato per rilevare vulnerabilità legate al gas all'interno dei programmi. Nel fare questo non compie alcuna operazione di calcolo della complessità, dunque non produce un output dove i consumi vengano quantificati. Riesce soltanto ad individuare e segnalare bug nel codice ;

EthIR svolge un analisi del bytecode generato a partire dal sorgente scritto in Solidity. L'analisi condotta dal framework produce in output una rappresentazione semplificata del bytecode, rendendolo più leggibile ma soprattutto più adatto per svolgere ulteriori analisi. In questo senso è stato utile per conoscere GASTAP, un software basato su EthIR che però di occupa di stimare i soli consumi di gas.

I software con i quali è stato possibile approfondire le ricerche invece sono:

GASTAP framework disponibile tramite un'interfaccia web, permette di dedurre degli upper bound ai consumi di gas per ciascuna funzione che compone lo smart contract preso in input. La precisione con cui conduce quest'analisi è pressoché simile a quella del compilatore di Solidity, solc. In alcuni casi GASTAP riesce ad essere anche più preciso di solc, in quanto riesce a dare upperbound finiti in casi in cui invece l'altro programma fallisce;

solc è il compilatore ufficiale di Solidity, il linguaggio di programmazione ad alto livello messo a disposizione da Ethereum per implementare gli smart contract. Ha una modalità di esecuzione che fornisce un upper bound ai consumi di gas per ciascuno dei metodi del programma preso in input. In tutti i casi testati produce un output, ma spesso il bound dato è infinito;

KEVM produce una semantica per il bytecode della EVM. In questo senso è uno strumento simile ad EthIR. Gli sviluppatori di KEVM però mettono a disposizione un'estensione del tool che permette di analizzare e dunque dare una stima dei consumi di gas.

4 Conclusioni

L'attività che ho svolto durante questo tirocinio è stata utile sotto diversi punti di vista.
In primo luogo mi ha permesso di conoscere, se pur in minima parte, il mondo delle crittovalute. Studiando la tecnologia blockchain ho capito il potenziale innovativo che porta con sè, e quali sono le sue possibili applicazioni. Ho trovato stimolante concentrarmi su questi argomenti.
Con questo tirocinio ho poi imparato ad orientarmi tra i lavori di ricercatori e docenti. Questo ha costituito per me un primo approccio alla ricerca, aprendomi lo sguardo su un mondo pieno di opportunità ed informazioni.
L'aspetto più importante resta l'aver messo in pratica le conoscenze acquisite in questi anni per svolgere un lavoro critico ed ottenere del materiale che mi servirà per la redazione della tesi.

Data Firma tirocinante

Firma tutor