TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Raido Pahtma
040771 IASM

ITT0020 Tehisintellekti algoritmid

Ründepuud

Kodutöö

Tallinn 2008

Töö eesmärk

Töö eesmärgiks oli koostada CoCoViLa keskkonna jaoks ründepuude pakett artikli „Rational Choice of Security Measures via Multi-Parameter Attack Trees“(Ahto Buldas, Peeter Laud, Jaan Priisalu, Märt Saarepera, Jan Willemson) alusel.

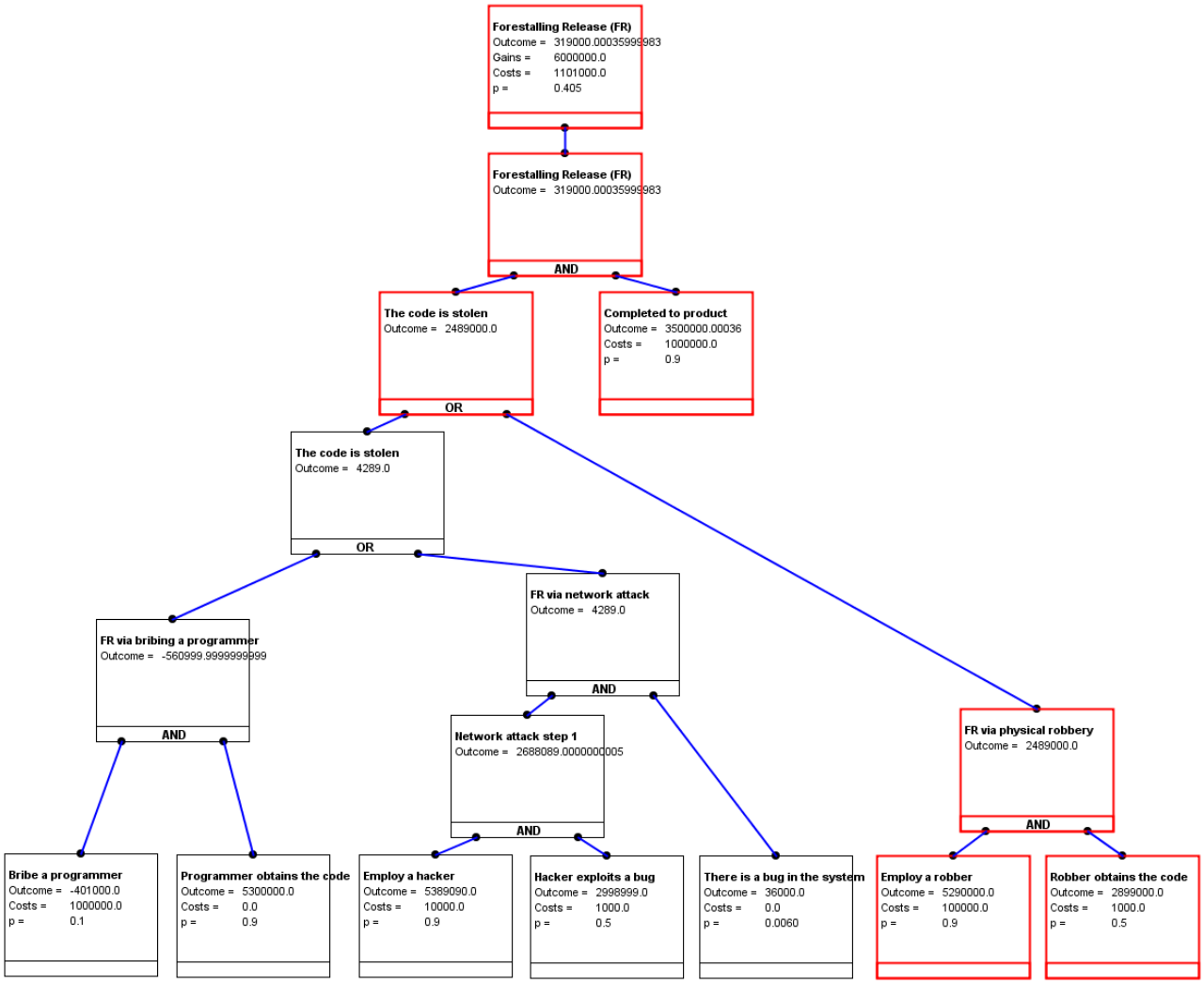
Puu koostamine

Puu koostamiseks saab kasutada nelja erinevat sõlme: ja, või, leht ning juur(root). Sobivalt puu kokku ühendanud, tuleks seadistada kõik lehed, mis kujutavad endast atomaarseid rünnaku samme. Kindlasti on vaja ette anda kulutused, õnnestumise tõenäosus ja tulud. Viimast on mõistlik(kuid mitte kohustuslik) teha juure kaudu, kuna tulud on ühised terve puu peale. Karistuste poole pealt tuleb lehtedele ette anda kas karistuse ja tõenäosuse paarid või kohe keskmised karistused(avPen ja avPenC). Kui sõlmedele kirjeldust mitte ette anda, luuakse neile kirjeldused nende sisendite alusel. Arvutusprotsessi tulemusena leitakse rünnaku tulemus(juure Outcome) ja kui see on suurem nullist, siis värvitakse parima tee koosseisu kuuluvad sõlmed punaseks.

Sõlmede väljad:

- Gains – rünnakust saadavad tulud, globaalne terve puu peale
- Costs – rünnakule tehtavad kulutused
- p – rünnaku õnnestumise tõenäosus
- avPen – keskmine karistus, kui rünnak õnnestus(artiklis π)
- avPenC – keskmine karistus, kui rünnak ebaõnnestus(artiklis $\pi\_$)
- q – karistada saamise tõenäosus, kui rünnak õnnestus
- Penalties – karistuse suurus õnnestunud rünnaku korral
- qC – karistada saamise tõenäosus, kui rünnak ebaõnnestus
- PenaltiesC – karistuse suurus ebaõnnestunud rünnaku korral
- Outcome – tulemus, *mängu väärtus*
- Selected – näitab, kas see sõlm on parima tee peal
- Description – sõlme kirjeldus, näidatakse puul

Artiklis käsitletud näide:



Java klassid

Sõlmede visuaalseks esitamiseks vajalik on kirjas AttackTree.xml failis. Sõlmede sisu on jaotatud viide faili:

Base.java
On baasklassiks kõigile teistele, sisaldab ühiseid omadusi ja meetodit, mis võimaldab sõlmede punaseks värvimist.

Leaf.java
Lisab baasklassile eraldi karistused, nende tõenäosused ja keskmiste karistuste arvutamise. Samuti defineerib väljundpordi.

Root.java
Sisaldab meetodit rünnaku edukuse määramiseks(kas tulemus suurem kui 0) ja defineerib sisendpordi.

And.java
Defineerib kaks sisend ja ühe väljundpordi, nende jaoks vajalikud muutujad ja arvutused vastavalt artiklis kirjeldatule. Sisaldab sisenditest lähtuvalt enda nime määramiseks meetodit.

Or.java
Defineerib kaks sisend ja ühe väljundpordi ja nende jaoks vajalikud muutujad. Valiku tegemiseks kasutab vastavaid meetodeid. Sisaldab sisenditest lähtuvalt enda nime määramiseks meetodit.

Sõlmede vahelised pordid järgivad kuju alias port = (Gains, Costs, p, avPen, avPenC, Outcome, Description, Selected), kus Selected ei ole sama, mis XML'is defineeritud väli, vaid boolean(väärtuse null edastamine sõlmede vahel oli problemaatiline ja kõigi muude väärtuste korral joonistatakse XML'is <known></known> vahele jääv alati ekraanile).

Kokkuvõte

Loodud paketiga koostatud ründepuu jõuab artikli algnäite ja tugevdatud turvalisusega näidetega samade tulemusteni. Paketiga on võimalik uurida süsteemi kaitstust ratsionaalselt käituvate ründajate vastu.