Auditability model for Intent-based SDN Applications

Nicolas Herbaut , Camilo Correa , Jacques Robin
Centre de Recherche en Informatique
Université Paris 1 Panthon-Sorbonne
Paris, France

Raul Mazo
Lab-STICC
ENSTA Bretagne
Brest, France

I. Conformance Checking Algorithm

- $\mathbb{I}$ The Intents model
- $\mathbb{F}$ The Flows model
- $\mathbb{T}$ Topology
- $\mathbb{T}$ Hosts
- $\mathbb{C}$ The security policiy configuration
- S_b^f: the set of host pairs for which at least one host is blocked and flows are still in place in the topology
- S_c^f: the set of host pairs none of which is blocked by the configuration for which the connectivity flows are missing
- S_b^i: the set of hosts pairs for which at least one host is blocked and intents are not installed
- S_c^i: the set of host pairs none of which is blocked by the configuration for which the intents are not installed.

During the conformance-checking phase, we repeat algorithm 1 for each log and each SCO configuration, to report S_b^f, S_c^f, S_b^i, S_c^i

In algorithm 1, the $K\_shortest\_simple\_path$ function is implemented from [1] with a complexity of $\mathcal{O}(KN^3)$, and generates all the non-looping path from two hosts, starting with the smallest. Assuming that there exist a cache of paths, the path lookup function complexity is assumed $\mathcal{O}(1)$. We can see that the conformance algorithm of flow conformance has $\mathcal{O}(KLN^2)$ with N the number of hosts and devices, K the number of simple paths between 2 hosts and L the average size of the K-shortest path, depending on the topology characteristics. The conformance checking of flow-based security and connectivity is at least cubic $\square$.

Intent conformance, on the other hand, is straightforward, since it involves looking up the intent list composed of host pairs to assure that no intent grant connectivity to a blocked host ($\mathcal{O}(n)$) and making sure that for each non-blocked host pair, there exist an intent ($\mathcal{O}(n^2)$). The conformance checking of intent-based security and connectivity is quadratic $\square$.

```
Data: (𝔽, 𝕀, 𝕋, ℂ, ℍ)
Result: S_b^f, S_c^f, S_b^i, S_c^f
C, C̄, S_b^f, S_c^f, S_b^i, S_c^f ← {} ;
for (h_src, h_dst) ∈ ℍ × ℍ \ {h_src ≠ h_dst} do
    if "block h_src h_dst" ∈ ℂ then
        C̄ ← C̄ ∪ {(h_src, h_dst)} ;
    else
        C ← C ∪ {(h_src, h_dst)} ;
    end
    for p ∈ K_shortest_simple_path(𝕋, 𝔽, h_src, h_dst)
     do
        connected ← true ;
        for (s, d) ∈ p do
            if f_{s,d} ∉ 𝔽 then
                connected ← false ;
                continue outter For;
            end
        end
        if connected ∧ (h_src, h_dst) ∈ C̄ then
            S_b^f ← S_b^f ∪ {(h_src, h_dst)}
        else if ¬ connected ∧ (h_src, h_dst) ∈ C then
            S_c^f ← S_c^f ∪ {(h_src, h_dst)}
    end
end
if 𝕀 ≠ {} then
    for (h_src, h_dst) ∈ ℍ × ℍ \ {h_src ≠ h_dst} do
        if (h_src, h_dst) ∈ 𝕀 ∧ (h_src, h_dst) ∈ ℂ then
            S_b^i ← S_b^i ∪ {(h_src, h_dst)}
        else if (h_src, h_dst) ∉ 𝕀 ∧ (h_src, h_dst) ∈ C̄ then
            S_c^i ← S_c^i ∪ {(h_src, h_dst)}
    end
end
```

Algorithm 1: Conformance checking algorithm

References

[1] J. Y. Yen, " Finding the K Shortest Loopless Paths in a Network ," *Management Science*, vol. 17, no. 11, pp. 712–716, jul 1971.