Trust Models on Adversarial Distributed Security Agents

Bc. Dita Hollmannová Ing. Sebastián García, Ph.D.

Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Computer Science

Problem introduction

Intrusion prevention systems (IPSs) are used to monitor the network and block attacks. While there may be multiple IPSs working on the same goal in the same network, their efforts are isolated, as they do not communicate.

We propose a peer to peer (P2P) network, where IPSs are peers that can share new detections about IP addresses, and ask others for their data.

The goal is to improve overall network security, as the attack will be known to all IPSs as soon as one of them detects it.

Common threats

A P2P network is open for anyone to join, and malicious peers controlled by the attacker can disrupt it. Trust is a value that describes how *good* the peer is, and it can be used to lower the impact of malicious peers.

Attackers can exploit the P2P trust model (for example) in the following ways:

- If the trust is computed by the network, clever manipulation can assign high trust to malicious peers.
- If the peers block after any negative report, a malicious peer may simply block communication with that IP by falsely reporting it
- If the peers need extensive reports before blocking an IP address, the attacker may not be blocked at all, because the peers wait too long

Basic principles

In computing trust, the following rules apply:

- Trust for a peer is computed from local experience only
- Trust is never shared (not because of secrecy, but because it is not needed)
- To compute trust, IPS detections for the IP address and protocol compliance is used. Access to IPS detections is a huge advantage of our model.

When sharing reports about an IP address, the following rules apply:

- Reports have a score (-1 = malicious, 1 = benign) and a confidence (0 = score is uncertain, 1 = score is certain)
- Reports are sent when detection changes, or when a peer asks
- Only reports generated by the local IPS are sent (reports from others are not forwarded)

Reports from a set of peers are combined to get a *network opinion* about an IP adres, which consists of Ω-score and Ω-confidence. Trust of each of the reporters is used to weight the reports. A prediction is computed based on the local score and confidence, and network values Ω-score and Ω-confidence. This is then compared to the threshold and a blocking decision is made.

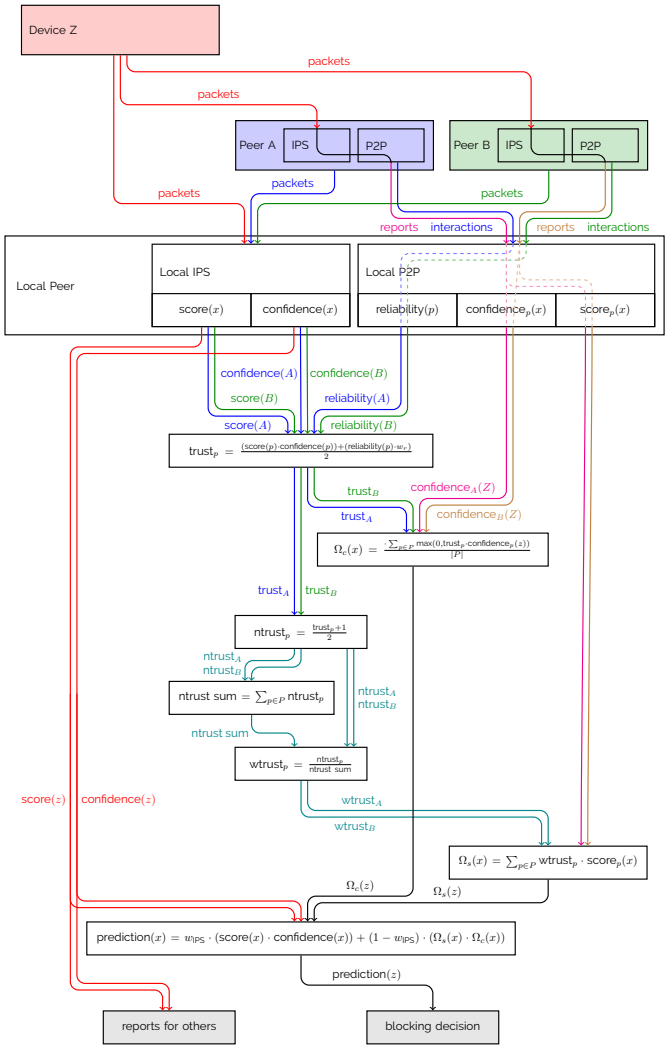
Experiments

To have control over the experiments, we simulate all network communication. Devices can choose to attack or not to attack, and IPSs make a simulated detection based on that. Then, peers in the network can communicate to exchange data.

Throughout the experiments, we change attacker strategies and number of malicious peers. We monitor blocking accuracy in one of the peers, the Observer. To allow for comparison across different experiment setups, the network traffic is always the same from the point of view of the Observer.

There are two devices, one malicious and one benign, on which the blocking decisions are evaluated. The malicious device targets the Observer in the second half of the experiment only, therefore the blocking accuracy of a perfect but non-cooperative Observer can never exceed 75%.

Trust computation diagram



Parameters

When making the prediction and the blocking decision, two parameters are used:

- **IPS weight** decides how are the network and IPS data combined. A weight higher than 0.5 means the IPS detection is more significant than the network data.
- **Threshold** Prediction is a value from -1 to 1, where -1 means the IP address is malicious and 1 means it is benign. If the prediction is lower than the threshold, the IP address will be blocked.

Choosing the best parameters

What happens if we set the weight to one? ($w = 1$)

This means we only take the IPS detection into account. Network data is ignored. The accuracy is the same (75%) across all experiments. Optimal threshold is e.g. -0.2.

Threshold	-1.0	-0.8	-0.6	-0.4	-0.2	0.0	0.2	0.4	0.6	0.8	1.0
Accuracy	0.5	0.7	0.725	0.725	0.75	0.75	0.75	0.75	0.75	0.75	0.75

What happens if we set the weight to zero? ($w = 0$)

This means we only take network data into account. Accuracy depends on the network setup:

2A All peers are honest and reporting
2C5 All peers are honest, but only 5 of 9 report the attacker
3A5 There are 4 malicious peers and 5 honest peers
4A1 All peers are malicious

Threshold	-1.0	-0.8	-0.6	-0.4	-0.2	0.0	0.2	0.4	0.6	0.8	1.0
2A	0.5	0.775	0.9	0.95	0.975	1.0	0.925	0.85	0.825	0.775	0.5
2C5	0.5	0.5	0.5	0.5	0.5	1.0	0.925	0.85	0.825	0.775	0.5
3A5	0.5	0.5	0.5	0.5	0.5	0.5	0.975	0.925	0.85	0.775	0.5
4A1	0.5	0.225	0.1	0.025	0.0	0.0	0.0	0.025	0.1	0.225	0.5

Using only network data greatly improves the accuracy in some cases, but if the network is not honest, as is the case in experiment 4A1, the accuracy drops to zero.

Cautious parameters

The goal is to select parameters such that accuracy is improved when peers are honest, and is never worse than the standalone IPS when peers are malicious. This can be done with $w = 0.8, T = -0.1$. In all our experiments, the accuracy never dropped below 75%, however, it was also never higher than that.

Parameters with assumptions about the network

To get better results, we need to make some assumptions about the network. If we assume that less than 50% of peers are malicious, we can choose parameters $w = 0.6, T = 0.2$. These parameters are still resilient in setups where the majority of peers is honest, the accuracy doesn't drop below 75%, but at the same time, in favorable setups, accuracy of 90% can be reached.

Contributions

In this thesis, we have created a protocol for sharing detection data between IPSs in a distributed manner. We have implemented the protocol into an existing free software IPS: the Stratosphere Linux IPS. The implementation is modular and can be easily deployed using a different IPS, different networking layer and different trust model. We have evaluated the protocol and created a testing environment, where future trust models can be tested.

For the full thesis text, please see: https://dspace.cvut.cz/handle/10467/90252.