Quality of Interaction among Path Computation Elements for Trust-aware Inter-Provider Cooperation

C. J. Fung*, B. Martini†, M. Gharbaoui‡,F. Paolucci‡,A. Giorgetti‡, and P. Castoldi‡
*Computer Science Department, Virginia Commonwealth University, USA - email: cfung@vcu.edu
† CNIT, Pisa, Italy - email: barbara.martini@cnit.it
‡Scuola Superiore Sant'Anna, Pisa, Italy - email: {m.gharbaoui, fr.paolucci, alessio.giorgetti, castoldi}@sssup.it

***Abstract*—Path Computation Element (PCE) architecture enables effective traffic engineering in multi-domain networks while limiting the exposure of intra-domain information. However, returned path computations might reveal confidential information if artfully correlated by a malicious PCE. Thus, the selection of domains sequence as the result of PCEs cooperation should depend not only on the capability of providing quality paths but also on factors related to expected revenues or perceived risks. In this scenario, cooperation among PCEs could benefit from a trust model by evaluating the quality of the past interactions.**

This work introduces the concepts of Quality of Interaction and trust ranking and elaborates a trust management model including effectiveness and security objectives regulating the cooperation among PCEs. Specifically, the proposed trust model aims at stimulating effective interactions among PCEs as a result of a common interest in contributing to successful and profitable path computations while avoiding misuse of path computation services. The simulation results show that our trust model is effective in detecting malicious PCE thereby tuning the amount of information returned in the path computation replies.

I. Introduction

To enable effective implementation of traffic engineering (TE) in multi-domain, i.e., multi-provider networks, while trying to guarantee an acceptable level of intra-domain information exposure, the Path Computation Element (PCE) architecture has been proposed in IETF [1]–[3]. The PCE architecture includes a PCE for each network domain that elaborates path computation requests issued by PCE clients returning a path that addresses the path requirements specified in the request (e.g., guaranteed bandwidth, delay). In the case of inter-domain scenario, the end-to-end inter-domain path is a concatenation of intra-domain paths resulting from cascaded request-response interactions among PCEs. This allows for the reduction of the amount of internal information to be shared among domains while providing effective traffic engineering in multi-provider environment [2], [4], [5]. However, despite of authentication, authorization and encryption mechanisms, confidentiality issues might arise among PCEs. In fact, returned path computations might reveal confidential information if artfully correlated by a malicious PCE. For instance, multiple requests with the same destination node but with different requested bandwidth might be submitted to a PCE. Instead of establishing the path, the obtained replies including path availability or unavailability can be used to derive possible bandwidth bottlenecks towards a certain destination [6].

On the other hand, PCE-based multi-domain path computation procedure implies a selection among different possible sequence of domains (i.e., sequences of PCEs) that address required QoS [7]. Recently, business and security objectives have been proposed to be considered besides QoS objectives in order to determine to which PCE to forward the path computation requests, or from which PCE to accept the path computation requests. A PCE might not have interest in processing requests from a competitor provider. Definitely, the selection of PCEs, i.e., sequence of domains, should depend not only on the capability of neighbor PCE to provide feasible paths but also on other elements of evaluation related to projected revenues or to risk information [6], [8].

For this purpose, this work presents a trust management model and introduces the concepts of *Quality of Interaction* (QoI) and *trust ranking* regulating the quality of cooperation among PCEs and the reciprocal access to path computation services. Specifically, the proposed trust model aims at stimulating effective PCE-to-PCE interactions as a result of a common interest in contributing to successful path computations while avoiding misuse of path computation services. In this context, a trust management is beneficial to enrich the interaction among PCEs by including business and security objectives. Trust management is a broadly investigated topic for collaborative systems in the information technology area, e.g., peer-to-peer networks [9], intrusion detection networks [10] and multi-agent systems in e-market [11]. In the communication technology area, the most noteworthy research works regard the access control and trust management in multi-domain networks [12], [13] considering also the PCE architecture [14], [15]. However, the aforementioned works are mainly focused on authorization mechanisms and on regulating the grant of access rights. As far as our knowledge, none of them addresses the rating of trustworthiness among PCEs aiming at creating an incentive for PCE cooperation as this work does.

II. Quality of Interaction and trust-aware PCE collaboration

In multi-domain PCE architecture, the PCEs belonging to different domains cooperate using client-server interactions for computing inter-domain paths. Specifically, a client PCE issues path computation requests to the server PCE of the target domain with specific requirements (i.e., source, destination, bandwidth, latency, etc.) in order to establish a path traversing the target domain. During normal operation, in case of affirmative reply, the PCE client is expected to set-up the computed

978-1-4799-2003-7/14/$31.00 ©2014 IEEE

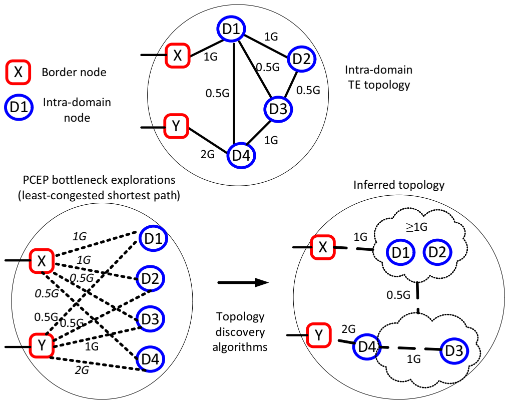

Fig. 1. Example of intra-domain topology information inferred by a malicious PCE without trust-based PCE interactions

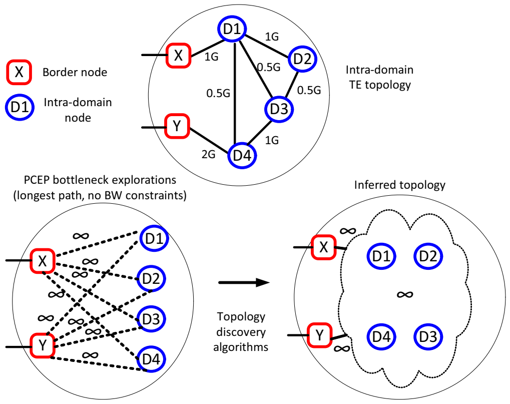

Fig. 2. Example of intra-domain topology information inferred by a malicious PCE with trust-based PCE interations

paths. However, a client PCE is not forced to actually set-up the returned path, therefore it might issue a sequence of bogus, but formally licit, computation requests to the server PCE with the only aim of inferring intra-domain information. For instance, multiple requests with the same destination node and different values of requested bandwidth might be submitted to a PCE. Instead of establishing the path, the obtained replies can be correlated to derive possible bandwidth bottlenecks toward specific destinations [6].

In this scenario, the Quality of Interaction (QoI) concept is introduced as basis for computing a trustworthiness ranking (i.e., the *trust ranking*) among PCEs. Specifically, the QoI is computed considering the outcome of a specific number N of past interactions, e.g., capability of providing feasible paths to client PCEs, number of effective path set-up issued by a client PCE using previously returned paths, rate of arriving requests. The obtained QoI values are then elaborated to provide a trust ranking measure evolving in the time that a server PCE can assign to the several client PCEs and used to provide differentiated path computation services thereby stimulating client PCEs to offer effective interactions.

Considering the scenario depicted in Fig. 1, the target domain has three intra-domain links (i.e., D1-D4, D1-D3 and D2-D3) with only 500 Mb/s of available bandwidth. This intra-domain bottlenecks divide the domain in two areas with limited inter-connectivity. Such information is considered strictly confidential by the domain provider. Despite PCE cooperation employs specific encryption mechanisms (i.e., using *path keys* to encrypt explicit sequence of nodes [16]), if trust-aware cooperation is not implemented, as shown in Fig. 1, a topology discovery mechanism at the client PCE could be able to correlate server PCE replies (i.e., bandwidth availability/unavailability) upon a sequence of bogus requests to infer intra-domain bandwidth bottlenecks [15], [17]. This demonstrates that path keys by itself does not fully preserve intra-domain confidentiality. If trust-aware cooperation is implemented, the trust ranking is used to tune the response of the PCE server. By comparing the trust ranking of the current PCE client to a fixed threshold the following steps are foreseen to preserve intra-domain information:

- When the trust ranking of the client PCE is above the threshold, the server PCE uses an effective path computation algorithm (e.g., shortest-path with bandwidth-guaranteed constraint). In this *normal mode* a reliable path is given back to the PCE client: if the resources are available and a path is returned, the following path set-up will likely to be successful;
- When the trust ranking of the client PCE is below the threshold, the server PCE uses a less effective path computation algorithm (e.g., without bandwidth-guaranteed constraints). This is called the *hiding mode* where a non reliable path is given back to the PCE client while masking the actual available bandwidth toward each destination: a path is likely to be returned, however, the following path set-up will likely to be failed.

As result of such hiding strategy, the client PCE is induced to believe that infinite bandwidth is available since a path returned anyway (see Fig. 2).

The use of hiding strategies while avoiding the refusal of path computation services achieves a twofold goal. On one hand, confidential information are preserved by making the intra-domain information not discoverable. On the other hand, the PCEs with low trust ranking receive responses that could not reflect the actual network utilization of the target domain, thus implying high latency or even possible blocking of path setup. Therefore, applying this strategy, the general PCE interest is to keep high trust ranking so that server PCEs will response using effective path computation algorithms so that the PCE receives higher service quality.

III. Trust Model

In this section, we present a math model which can be used to calculate the QoI and the trust ranking between PCEs. Specifically, we use a Bayesian probabilistic model to predict the likelihood of a PCE's future cooperation quality based on its past quality of path computation requests.

A. Quality of Interactions

The QoI is assessed at each server PCE by evaluating the likelihood that the received sequence of a number of path computation requests contains confidentiality attacks as well as the rate at which the computed path are actually established.

Confidentiality attacks are envisioned to be carried out in the form of an opportune sequence of bogus requests, whose replies are correlated by malicious PCE clients. Therefore, it is desirable to target a sequence of requests to evaluate the probability that requests are for the purpose of inferring confidential information. Particularly, we observe requests within an observation window containing the last N requests from the same client PCE ending at the current observation time. After each round, the observation window is shifted forward by an offset of ΔN requests, named the Overlap Rate (OR) to catch possible malicious plot of issued requests.

A QoI value is computed for each observation window based on the level of QoI of the received requests within the window. Particularly, we use an anomaly-based approach to detect suspicious sequence of requests. In this approach, a statistical distribution of normal requests are defined based on historical data of normal interactions. A normality score is computed for each sequence of requests based on their likelihood to be normal. Correspondingly, a QoI value is assigned to the sequence. If the QoI value is zero or very low, then the sequence may be malicious. The higher the QoI, the less likely the sequence is malicious.

Based on our previous works [17], [18], three variables are used to evaluate the normality of a sequence of PCE requests, namely, the *total inter-arrival time* (i.e., the sum of all the PCE requests' inter-arrival time in the observation window, denoted by x); the *average requested bandwidth* (i.e., the average requested bandwidth for all requests in the observation window, denoted by y); and the *stress to destinations* (i.e., the number of requests to each destination during the observation window, denoted by $\vec{z} = \{z_1, z_2, ..., z_d\}$, where d is the number of all destinations and $\sum_{i=1}^{d} z_i = N$). We denote the PDF distributions of the above three parameters for normal traffic by $f(x)$, $g(y)$, and $h(\vec{z})$, respectively.

Note that the variables defined above are independent. We assume that requests with smaller inter-arrival time, larger requested resource bandwidth, and larger deviation from the expected distribution for stress to destination, the less likely the requests sequence is normal and therefore less satisfied the server PCE is. Therefore, given an observation $(x, y, \vec{z})$, we design the QoI of sequence of requests as follows:

$$\begin{aligned} Q(x, y, \vec{z}) &= (Q(x)Q(y)Q(\vec{z}))^{\beta} \\ &= (\textstyle\int_0^x f(u)du \int_y^{\infty} g(v)dv \int_{|\vec{z}-\bar{\vec{z}}|}^{\infty} \psi(w)dw)^{\beta} \end{aligned} \tag{1}$$

where $Q(x, y, \vec{z})$ is the QoI level of an observation triple $(x, y, \vec{z})$. $Q(x)$ is the QoI level on the total inter-arrival time; $Q(y)$ is the QoI level on the requested resource; $Q(\vec{z})$ is the QoI level on the destination stress distribution. $\psi(w)$ is the projected distribution of variable $\vec{z}$ to $w = |\vec{z} - \bar{\vec{z}}|$, where w is the Euclidean distance from the observed stress $\vec{z}$ to expected stress distribution $\bar{\vec{z}}$. $\beta \in [0, 1]$ is the quality score normalization parameter.

B. Bayesian Trust Model

Bayesian statistics provide a theoretical foundation for measuring the uncertainty in a decision that is based on a collection of observations. We demonstrate the distribution of QoI levels of the interactions from each domain and, particularly, using this information to estimate the QoI level of future consultations. For multi-valued QoI levels, Dirichlet distributions can be used for prediction.

A Dirichlet distribution is based on initial beliefs about an unknown event represented by a prior distribution. The initial beliefs combined with collected sample data can be represented by a posterior distribution. The posterior distribution well suits our trust management model since the trust is updated based on the history of interactions.

Let Q be the discrete random variable denoting the discrete QoI level of the PCE request sequence. Q takes values in the set $\mathcal{Q} = \{q_1, q_2, ..., q_k\}$ ($q_i \in [0, 1]$, $q_{i+1} > q_i$) of the supported levels of QoI, which is the QoI (Equation 1) mapped to the closest q_i. Let $\vec{p} = \{p_1, p_2, ..., p_k\}$ ($\sum_{i=1}^{k} p_i = 1$) be the probability distribution vector of Q, i.e. $P\{Q = q_i\} = p_i$. Also, let $\vec{\gamma} = \{\gamma_1, \gamma_2, ..., \gamma_k\}$ denote the vector of cumulative observations and initial beliefs of Q. Then we can model $\vec{p}$ using a posterior Dirichlet distribution as follows:

$$f(\vec{p}|\xi) = Dir(\vec{p}|\vec{\gamma}) = \frac{\Gamma(\sum_{i=1}^{k} \gamma_i)}{\prod_{i=1}^{k} \Gamma(\gamma_i)} \prod_{i=1}^{k} {p_i}^{\gamma_i - 1} \tag{2}$$

where ξ denotes the background knowledge, which is the initial believe and observations. Here ξ is summarized by $\vec{\gamma}$. Let:

$$\gamma_0 = \sum_{i=1}^{k} \gamma_i \tag{3}$$

The expected value of the probability of Q to be q_i given the history of observations $\vec{\gamma}$ is given by:

$$E(p_i|\vec{\gamma}) = \frac{\gamma_i}{\gamma_0} \tag{4}$$

In order to give more weight to recent observations over old ones, we embed a *forgetting factor* $F \in [0, 1]$ in the Dirichlet background knowledge vector $\vec{\gamma}$ as follows:

$$\vec{\gamma}^{(n)} = \sum_{i=1}^{n} F^{i-1} \times \vec{Q^i} + c_0 F^n \vec{Q^0} \tag{5}$$

where n is the number of observations; $\vec{Q^0}$ is the initial beliefs vector. If no additional information is available, all outcomes have an equal probability making $Q_j^0 = 1/k$ for all $j \in \{1, .., k\}$. Parameter $c_0 > 0$ is a priori constant, which puts a weight on the initial beliefs. Vector $\vec{Q^i}$ denotes the QoI level of the i^{th} observation, which is a tuple containing $k - 1$ elements set to zero and only one element set to 1, corresponding to the selected QoI level for that observation. For example, if the QoI of the jth observation is q_2, then $\vec{Q^j} = \{0, 1, ..., 0\}$. Parameter $F \in [0, 1]$ is the forgetting factor. A small F makes old observations quickly forgettable. For the purpose of scalability, the $\vec{\gamma}^{(n)}$ in Equation 5 can be rewritten in terms of $\vec{\gamma}^{(n-1)}$, $\vec{Q^n}$ and Δt_n as follows:

$$\vec{\gamma}^{(n)} = \begin{cases} c_0 \vec{Q}^0 & n = 0 \\ F \times \vec{\gamma}^{(n-1)} + \vec{Q}^n & n > 0 \end{cases} \tag{6}$$

C. Evaluating the PCE trust ranking

After a server PCE u receives path computation requests from PCE client v, it assigns a QoI value to the request sequence according to the likelihood that the requests are normal ones. This QoI value is assigned with one of the QoI levels in the set $\mathcal{Q} = \{q_1, q_2, ..., q_k\}$ that has the closest value. Each QoI level q_i also has a weight w_i.

The distribution will be updated with new QoI observations. Let p_i^{uv} denote the probability that PCE client v sends PCE requests to PCE server u with QoI level q_i. Let $\vec{p}^{uv} = (p_i^{uv})_{i=1...k}$, such that $\sum_{i=1}^{k} p_i^{uv} = 1$. We can interpret $\vec{p}^{uv}$ as the likelihood that the QoI of the next PCE sequence lies in each QoI levels $\mathcal{Q}$. We model $\vec{p}^{uv}$ using Equation 2. Let Y^{uv} be the random variable denoting the weighted average of the probability of each QoI level in $\vec{p}^{uv}$.

$$Y^{uv} = \sum_{i=1}^{k} p_i^{uv} w_i \tag{7}$$

The trust ranking T^{uv} of peer v as noticed by peer u is then calculated as:

$$T^{uv} = E[Y^{uv}] = \sum_{i=1}^{k} w_i E[p_i^{uv}] = \frac{1}{\gamma_0^{uv}} \sum_{i=1}^{k} w_i \gamma_i^{uv} \tag{8}$$

where γ_i^{uv} is the cumulated evidence that v has sent requests to u with QoI level q_i.

IV. Evaluation

In this section, simulations are used to evaluate the effectiveness of the proposed QoI evaluation method and trust ranking algorithm. The interaction of two PCE agents, one PCE server and one PCE client is considered. The client sends normal PCE requests and possibly malicious PCE requests to the server. The server uses the proposed QoI to measure the performance and trust level of the client. More complex attack scenarios have been described in [15], [17] and the effectiveness of the proposed method in such scenarios will be evaluated in future works.

A. Simulation settings

Taking into account the typical network parameters (see IV-A), and without loss of generality, we consider the following assumptions on normal PCE requests in our simulation:

- Requests are independent and arrive with exponentially distributed inter-arrival time with average arrival rate λ (i.e., $t \sim \lambda e^{-\lambda t}$) [19].
- Requested bandwidth is uniformly distributed in the range [a, b] [20], [21].
- Requests destination is uniformly distributed among all nodes in the specific domain (i.e., $\vec{\bar{z}} = \{N/d, ..., N/d\}$).

With the aforementioned assumptions the considered features (i.e., x, y, $\vec{z}$) satisfy the following distributions:

- x (i.e., time needed to collect N requests) satisfies $Gamma$ distribution with shape parameter N and inverse scale parameter λ.
- y (i.e., average requested bandwidth of N requests) can be approximated with a normal distribution $\mathcal{N}(m_y, \sigma_y^2)$ where $\mu = \frac{(a+b)}{2}$ and $\sigma^2 = \frac{1}{N}\frac{(b-a)^2}{12}$.
- $\vec{z} = \{z_1, \ldots, z_d\}$ (i.e., the number of requests addressed to each destination) satisfies multinomial distribution with N trials and $p_1 = \ldots = p_d = 1/d$.

The probability distribution of the considered features (i.e., x, y, $\vec{z}$) follow:

$$f(x) = \Gamma(N, \lambda) = \frac{\lambda^N}{(N-1)!} e^{-\lambda x} x^{(N-1)} \tag{9}$$

$$g(y) = \mathcal{N}(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-\mu)^2}{2\sigma^2}} \tag{10}$$

$$h(\vec{z}) = \binom{N}{z_1 \ldots z_d} \prod_{i=1}^{d} p_i^{z_i} = \frac{\binom{N}{z_1 \ldots z_d}}{d^N} \tag{11}$$

Correspondingly, we have

$$\psi(w) = \sum_{\vec{z}:|\vec{z}-\vec{\bar{z}}|=w} h(\vec{z}) = \sum_{\vec{z}:|\vec{z}-\vec{\bar{z}}|=w} \frac{\binom{N}{z_1 \ldots z_d}}{d^N} \tag{12}$$

Therefore the QoI can be written as,

$$Q(x, y, \vec{z}) \quad = \tfrac{1}{2}(\tfrac{1}{(n-1)!}\gamma(n, \lambda x))(1 - erf(\tfrac{y}{\sqrt{2}}))(\textstyle\sum_w^\infty \psi(w))$$

where $\gamma(n, \lambda x) = \int_0^{\lambda x} t^{n-1} e^{-t} dt$ is the lower incomplete gamma function. $erf(\frac{y}{\sqrt{2}}) = \frac{1}{\sqrt{\pi}} \int_{-\frac{y}{\sqrt{2}}}^{\frac{y}{\sqrt{2}}} e^{-u^2} du$ is the error function representing the probability of a random variable with normal distribution of mean 0 and variance $1/2$ falling in the range $[-\frac{y}{\sqrt{2}}, \frac{y}{\sqrt{2}}]$.

The path computation requests submitted to the PCE are characterized by the following statistics:

- Exponentially distributed holding time of path requests with average $1/\mu = 200\ s$ [22]
- Exponentially distributed inter-arrival time of path requests with average $1/\lambda = 0.125\ s$ [19].
- Uniform distribution of path bandwidth range [a, b] with $a = 200\ Mbps$ and $b = 1000\ Mbps$ [20], [21].
- Uniformly distributed destination among all nodes ($d = 14$) in the specific domain.

The other parameters we have used for the trust model in the simulation are: $\beta = 0.05$, $F = 0.9$, $k = 10$, $c_0 = 10$, and $\{w_1, w_2..., w_k\} = \{0.1, 0.2, ..., 1.0\}$ (see Section III).

Based on the simulation setting described above, we simulate $500,000$ normal PCE requests ($10,000$ observation windows) and monitor the distribution of the three PCE parameters, namely, total inter-arrival time, average bandwidth, and destination stress distribution. The PDF and CDF distributions of the three parameters are shown in Figure 3, 4, and 5.

B. Simulation results

In this section, we simulate the interaction between PCEs and evaluate the proposed trust management model. The number of path computation requests is fixed to $1,500$ and the

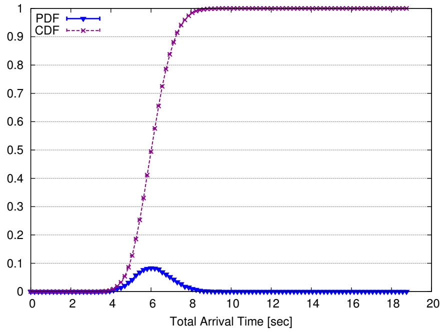

Fig. 3. Cumulative probability distribution on the total inter-arrival time

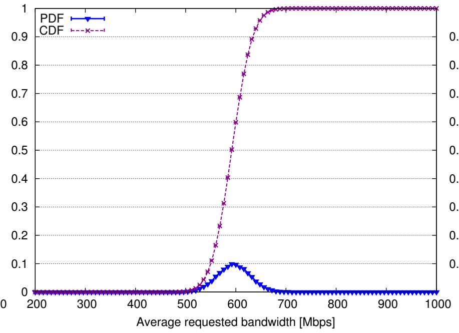

Fig. 4. Cumulative probability distribution on the average bandwidth

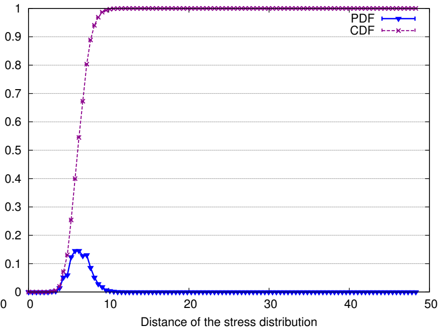

Fig. 5. Cumulative probability distribution on the stress distance to uniform

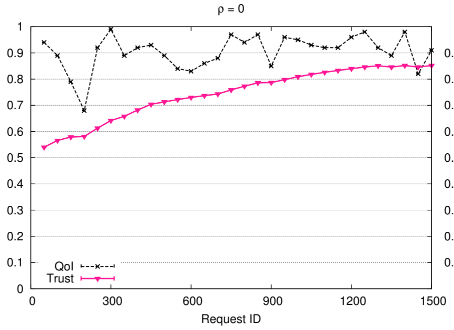

Fig. 6. Pure benign sequence

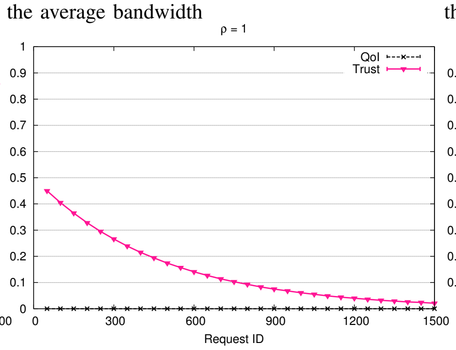

Fig. 7. Pure malicious sequence

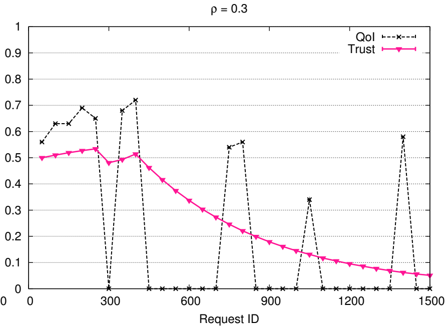

Fig. 8. Smart sequence - OR=0.5

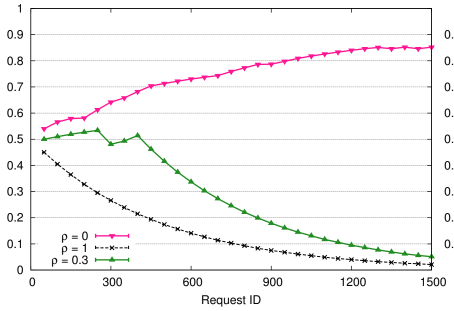

Fig. 9. Trust values under different malice percentage - OR=0.5

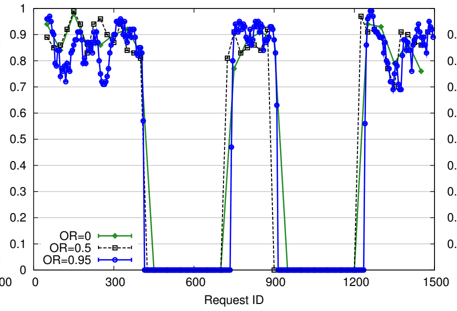

Fig. 10. QoI values under different OR values

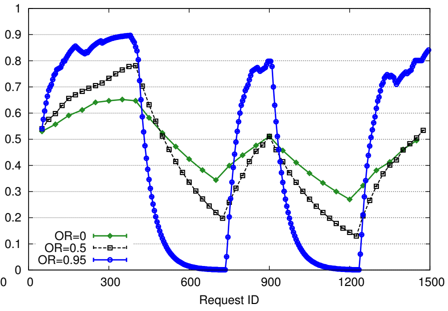

Fig. 11. Trust ranking values under different OR values

number of requests per observation window N is fixed to 50. We use a parameter OR to denote the overlap ration between adjacent observation windows. For example, if $N = 50$ and $\Delta N = 10$, then $OR = 0.2$. In this simulation, the overlap rate between adjacent observation window is set to $OR = 0.5$. Malicious traffic is simulated by a sequence of requests which has large bandwidth demand to the same destination.

Fig. 6 and Fig. 7 show the trend of the Quality of Interest and the trust ranking values as a function of time in the case of pure benign ($\rho = 0$) and pure malicious ($\rho = 1$) sequences, respectively. In particular, each observation window includes 50 path computation requests and the *overlap rate* (OR) is set to 0.5. In such a case, the QoI and trust ranking values are updated each 25 requests. The trends show that the QoI is always in the range $[0.8, 1]$ for the benign sequence and always locked to zero for the malicious sequence. The trust ranking value is initialized to 0.5 for both sequences. It almost linearly increases towards higher values for the benign sequence, to then stabilize at 0.8 which states that the PCE is not malicious. On the contrary, in the malicious sequence, the trust ranking non-linearly decreases towards low values and then stabilizes when it reaches zero. It is worth noting that after receiving a total amount of 400 malicious requests the trust is lower than 0.2.

Pure benign and malicious sequences represent the two extreme cases where the PCE behaves without hiding its intrusion, in an expected way. A smarter malicious PCE might embed a set of malicious requests within a non malicious sequence, in order to extort some information, without being unmasked. The percentage of malicious requests inserted in the benign sequence, called *malice percentage* is equal to 0.3 in Fig. 8 where the QoI and the trust ranking are plotted. We can notice that whereas the QoI oscillates between high and low values, corresponding to the set of benign and malicious requests, respectively, the trust on the contrary takes more time to decrease, remaining almost stable at 0.5 in the first eight windows, however it shows to be an affordable estimator of the percentage of malicious requests within a mixed sequence and tends to zero, thus confirming the malicious behavior of the smart PCE.

In Fig. 9 the trust value plot is depicted under the three values of ρ. We notice that with respect to the pure malicious case, when ρ is greater than zero, the trust ranking takes more time to evaluate the behavior of the smart PCE because of the benign requests present in the sequence. However, thanks to its memory capacity, after a certain time (400 requests) the trust starts decreasing to finally tend to zero.

To evaluate the reactiveness of the trust model, a PCEP sequence with time-variant behavior is submitted, where an alternation between benign requests and malicious requests is forced. This sequence represents the scenario where an attack appears suddenly after a normal period and disappears after a given number of requests which stand for the amount of information extorted by the attacker.

Fig. 10 and Fig. 11 show the satisfaction level and the trust value with different OR values (i.e., 0.95, 0.5, 0 respectively). When OR is high (e.g., OR=0.95) the number of evaluations increases and the trust value is updated more frequently. In this case the reaction to a malicious sequence is practically immediate, since the presence of the first 15 malicious requests are sufficient to drastically decrease the satisfaction level to zero. The trust level, starting from high levels (i.e., 0.9) immediately turns and rapidly decreases, clearly raising the alarm about anomalous requests. With lower values of OR, the behavior is the same but it is more smoothed, meaning that reactivity remains high, however, the updated values are closer and the evaluation has to be considered within a restricted range of trust values.

From these simulations results we can conclude that the trust ranking can effectively track the malicious behavior of a PCE. Varying the overlap rate corresponds to tuning the learning speed of the trust model in order to declare the PCE malicious/benign which is faster for high OR values. However, increasing the OR means increasing the computational rate of the QoI and the trust ranking which makes necessary the need to fix a tradeoff between the detection accuracy and the computational cost.

V. Conclusion

This paper proposed a trust management model for multi-provider communication networks where the path computation is performed through collaborative interaction among PCEs. In this scenario, the paper introduced the concepts of QoI and trust ranking and elaborated a trust management model including effectiveness, security, and business objectives regulating the PCEs cooperation.

The proposed model is evaluated by means of simulations in realistic multi-provider network scenarios. The obtained results showed the effectiveness of the model when a malicious PCE tries to retrieve confidential intra-domain information of a neighbor domain. Other attack scenarios will be defined and evaluated in future works.

References

[1] A. Farrel, J. P. Vasseur, and J. Ash, "A Path Computation Element (PCE)-based architecture," *IETF, RFC 4655*, Aug 2006.

[2] S. Dasgupta, J. De Oliveira, and J. P. Vasseur, "Path-computation-element-based architecture for interdomain MPLS/GMPLS traffic engineering: Overview and performance," *Network, IEEE*, vol. 21, no. 4, pp. 38–45, 2007.

[3] F. Paolucci, F. Cugini, A. Giorgetti, N. Sambo, and P. Castoldi, "A survey on the path computation element (pce) architecture," *Communications Surveys Tutorials, IEEE*, vol. 15, no. 4, pp. 1819–1841, 2013.

[4] L. Buzzi, M. Bardellini, D. Siracusa, G. Maier, F. Paolucci, F. Cugini, L. Valcarenghi, and P. Castoldi, "Hierarchical border gateway protocol (HBGP) for PCE-based multi-domain traffic engineering," in *Proc. ICC 2010*, 2010, pp. 1–6.

[5] D. Siracusa, S. Grita, G. Maier, A. Pattavina, F. Paolucci, F. Cugini, and P. Castoldi, "Domain sequence protocol (DSP) for PCE-based multi-domain traffic engineering," *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 4, no. 11, pp. 876–884, 2012.

[6] F. Paolucci, M. Gharbaoui, A. Giorgetti, F. Cugini, B. Martini, L. Valcarenghi, and P. Castoldi, "Preserving confidentiality in PCE-based multi-domain networks," *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 3, no. 5, pp. 465–474, 2011.

[7] A. Giorgetti, F. Paolucci, F. Cugini, and P. Castoldi, "Impact of intra-domain information in GMPLS-based WSONs with hierarchical PCE," in *Tech. Dig. OFC/NFOEC*, 2012, pp. 1–3.

[8] N. Djarallah, H. Pouyllau, N. Le Sauze, and R. Douville, "Business-driven PCE for inter-carrier QoS connectivity services," in *Proc. FutureNetw 2011*, 2011, pp. 1–8.

[9] C. Duma, M. Karresand, N. Shahmehri, and G. Caronni, "A trust-aware, p2p-based overlay for intrusion detection," in *Proc. DEXA '06*, 2006, pp. 692–697.

[10] C. J. Fung, J. Zhang, I. Aib, and R. Boutaba, "Dirichlet-based trust management for effective collaborative intrusion detection networks," *IEEE Trans. Netw. Service Manag.*, vol. 8, no. 2, pp. 79 –91, june 2011.

[11] J. Zhang and R. Cohen, "Trusting advice from other buyers in e-marketplaces: the problem of unfair ratings," in *Proc. ICEC '06*, 2006, pp. 225–234.

[12] Y. Demchenko, M. Cristea, and C. de Laat, "XACML policy profile for multidomain network resource provisioning and supporting authorisation infrastructure," in *Proc. IEEE POLICY 2009*, Jul. 2009, pp. 98 –101.

[13] M. Colombo, F. Martinelli, P. Mori, B. Martini, M. Gharbaoui, and P. Castoldi, "Extending resource access in multi-provider networks using trust management," *International Journal of Computer Networks and Communications (IJCNC)*, vol. 3, no. 3, pp. 133–147, may 2011.

[14] S. Polito, S. Zaghloul, M. Chamania, and A. Jukan, "Inter-domain path provisioning with security features: Architecture and signaling performance," *IEEE Trans. Netw. Service Manag.*, vol. 8, no. 3, pp. 219 –233, Sept. 2011.

[15] M. Gharbaoui, F. Paolucci, A. Giorgetti, B. Martini, and P. Castoldi, "Effective statistical detection of smart confidentiality attacks in multi-domain networks," *IEEE Trans. Netw. Service Manag.*, vol. 10, no. 4, pp. 383–397, December 2013.

[16] R. Bradford, J.-P. Vasseur, and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Key-Based Mechanism," *IETF, RFC 5520*, April 2009.

[17] M. Gharbaoui, F. Paolucci, A. Giorgetti, B. Martini, and P. Castoldi, "Effective statistical detection of smart confidentiality attacks in multi-domain networks," in *Proc. IM2013*, 2013.

[18] M. Gharbaoui, F. Paolucci, A. Giorgetti, B. Martini, and P. Castoldi, "Statistical approach for detecting malicious PCE activity in multi-domain networks," in *Proc. HPSR 2012*, Jun. 2012.

[19] A. Giorgetti, F. Cugini, N. Sambo, F. Paolucci, N. Andriolli, and P. Castoldi, "Path state-based update of PCE traffic engineering database in wavelength switched optical networks," *Communications Letters, IEEE*, vol. 14, no. 6, pp. 575–577, 2010.

[20] D. Wang and Guangzhi, "Efficient distributed bandwidth management for MPLS fast reroute," *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, Apr. 2008.

[21] J. Gao and D. Li, "Bod service with VCAT/LCAS and GMPLS signalling," in *Proc. NOMS 2008*, 2008, pp. 207–211.

[22] Cisco and VMware, "Virtual machine mobility with vmware vmotion and cisco data center interconnect technologies," white paper, 2009.