Risk-Based Packet Routing for Privacy and Compliance-Preserving SDN

Karan K. Budhraja* Abhishek Malvankar† Mehdi Bahrami‡ Chinmay Kundu§ Ashish Kundu¶ Mukesh Singhal‖

*University of Maryland, Baltimore County, MD, USA Email: karanb1@umbc.edu
†IBM Watson Health, Yorktown Heights, NY, USA Email: asmalvan@us.ibm.com
‡Fujitsu Laboratories of America, Sunnyvale, California, USA Email: mbahrami@us.fujitsu.com
§Researcher, India Email: ckkundu@gmail.com
¶IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA Email: akundu@us.ibm.com
‖Cloud Lab, University of California Merced, USA Email:msinghal@ucmerced.edu

***Abstract*—Software Defined Networking (SDN) is increasingly being used in data centers as well as enterprise networks. In an environment that has strict compliance requirements, such as HIPAA compliance, a critical role for an SDN controller is to route all data packets while considering data privacy-preservation and compliance-preservation. In this paper, we address this problem by proposing a routing protocol for SDN which is an efficient risk-based swarm routing protocol. The programmable capability of controllers is exploited in order to minimize privacy and compliance risks in data transmission. The proposed routing protocol is based on the Ant Colony Optimization technique and machine learning, while the data for learning is obtained from OVSDB and the OpenvSwitch Database management protocol. We collect a history of packet transfers for training purposes and learn from the training data to efficiently and intelligently route sensitive data packets while it preserves the target compliance. This routing is obtained by intelligent eviction of rules that are downloaded to the switches. We have implemented the proposed schemes based on an RYU controller.**

***Index Terms*—SDN; cloud computing; privacy; security; routing; compliance-preservation.**

I. Introduction

Software-Defined Networking (SDN) [1], [2], [3], [4] is a networking approach that simplifies and optimizes network operations.

Existing work on SDN routing has not addressed the problem of data transmission in the presence of privacy and compliance requirements. Another privacy requirement that needs to be considered in this context is that only a static or specific routing path (a sequence of switches) should not be used transfer high density of packets carrying sensitive data, since it poses a risk of Denial of Service attacks as well as inference attacks. Such a scheme, if used for controlled environment such as HIPAA, could lead to a potential information leakage when attacked.

In this paper, we address data privacy and compliance restriction challenges for transferring sensitive data in SDN environment that preserves privacy and adheres to the target compliance, such as HIPAA. We achieve our goals that by using: i) distributed routing optimization by employing a swarm routing algorithm; ii) definition of privacy, risk and techniques to compute such risks on an SDN network using analytics and machine learning; iii) minimization of risk and route-randomization in order achieve privacy and compliance requirements. The contribution of this work is a novel approach to route privacy-sensitive and compliance-sensitive data in an SDN environment. This is achieved by reducing a notion of privacy risks and using swarm-based (ant) routing [5], [6], [7]. The privacy risk is dynamically computed over the network node graph and is thus holistic in nature.

We perform route-randomization by monitoring the amount of packets transmitted on the forwarding path with help of openflow protocol and utilites such as ovs-vsctl (a utility for querying and configuring ovs-vswitchd) and ovs-ofctl (to administer OpenFlow switches) [8]. Once we have counter metrics, we define risk as a function of number of packets transferred on the same datapath. This number would be calculated by an agglomeration of offline log monitoring models in python sci-kit learn. Monitoring models in this fashion would provide features to an algorithm based on probabilisitic graphical models (which is Ant Colony Optimization in our case, explained in Section IV) for traffic spreading across other nodes to minimize risk.

The organization of the paper is as follows. Section II discusses relevant existing work in this domain and related to our research problem statement. Section III then formalizes the problem that our work intends to provide a solution to. Section IV and Section V describe the details related to the proposed solution. Finally, Section VI evaluates the proposed method on varied associated parameters and Section VII concludes our work with closing remarks and scope for future work and any open questions.

II. Related Work

The concept of privacy preservation routing is important when dealing with sensitive data [9], [10]. The former cited work discusses an attempt to discover the patterns by which a controller installs a flow, and thereby learn from request for a custom installation. This is done from the point of view of an adversary. The latter preserves privacy by using private-key based encryption for each data packet. Neither, however, consider compliance-based networking which is explored in this study.

The congestion control use case of Ant Colony Optimization (ACO) is implemented in [5], [6], [7]. Our work extends this idea by combining the relatively low time-complexity with respect to traditional shortest path problem which is traditionally NP-hard. We use an ant-based approximation solution with privacy and compliance constraints.

III. Problem Definition

Our work focuses on two types of risks to the network traffic. They are as follows.

A. Privacy Exposure

Consider a scenario where the flow of packets through the network uses a fixed route or set of routes, for a given source (s) and destination (d). The switches along those routes may then identify the presence of communication between s and d. Depending on the intelligence capabilities of the switches, they may then be able to learn or gather sensitive information about the data being transmitted. This results in a leakage of privacy for that communication session.

B. Compliance

To discuss compliance risk, consider a setting where a network topology has grown over time. When planning of changes in network topology, this translates to varied hardware components gathered over time. As a result, there may only be a subset of those components that are capable of adhering to a certain role or requirement. For our problem, this means that a subset of switches (and/or controllers) used by the network may be compliant to process data transferred in regulated environment. It therefore becomes a mandate, then, that any data traveling across the network that requires compliance, be constrained to such a selected subset of network components. This may also be considered for a given (s, d) pair, in which case all routes connecting those components must be checked. In other words, network components or routes that do not comply with compliance requirement will pose a greater risk when catering to HIPAA sensitive data. This is not a favorable situation for our problem setting.

IV. Proposed Privacy and Compliance-Preserving Routing Technique

A critical role for an SDN controller is to route data packets by privacy-preserving of all data packets. The proposed routing protocol collects a history of data packets for training purposes and it uses the training data to efficiently and intelligently route data packets. The key role of our swarm intelligence routing protocol is to allow for a small amount (less than traditional networking algorithms because of being a distributed intelligence approach rather than a centralized one) of processing (computation by the network component with respect to time complexity) per data packet. In the future, this can be extended to pushing computation to intelligent switches to share the load of the controller. In order to preserve the users data privacy in our proposed SDN routing protocol, we propose a light-weight data privacy method that allows the protocol efficiently use meta-data of data packets in the training data set to intelligently route the packets.

In this section, we show how our proposed routing protocol preserves the users' data privacy and discuss the various components involved in the proposed solution.

A. Privacy Exposure

In order to preserve privacy packets should be sent across multiple paths. Instead of having a fixed route or set of routes that are used for communication, data packets may be sent probabilistically along paths in a network. While a convergence mechanism may be used to ensure a bound to communication time, the intermittent passage of packets through any given switch will significantly reduce the risk of privacy exposure.

B. Compliance

In order to determine low-risk routes for sending sensitive information, the SDN controller could determine which network components are compliant. A path computation then involving those components could be formulated. This results in minimal risk when handling sensitive data sent across the network. An upper bound (limit) to such routing may also be considered.

V. Risk Analysis

Risk computation provides quantify risks across various network components. This quantification can then be used to make an optimal decision for the routing of sensitive traffic. The following are the key factors in computation and usage of risk quantification.

1. Risk for each router may be considered as categorized based on the discussion in Section III. This results in two types of risks: (i) privacy exposure risk and (ii) compliance risk. The privacy exposure risk may be quantified by evaluating the security of a router and its vulnerability to external attacks. The compliance risk for a network may be quantified by evaluating the number of controls that are not implemented in that network component.For instance the total number of HIPAA controls is 59, this value could be scaled by 59 to result in a value in the range $(0, 1)$.

2. The overall (global) risk for a graph or subgraph of network components, may then be computed and quantified by focusing on the maximum flow of compliant route or node and low privacy exposure data across the network.

Given a graph $G(V, E)$, where V and E are the sets of vertices and edges respectively (for that graph), the controller output after processing (as explained above) will be as follows. The controller may take one of two possible decisions.

1. This is the case where the controller has found a path or a set of paths across the network, by which sensitive data may be routed. The paths are selected so as to minimize privacy exposure and compliance risks for a given data transmission session.

2. On computation of the risks along various paths in the network, the controller may not find any favorable path on

which data may be transmitted without significant risk in context of privacy exposure and compliance. In this case, the controller acknowledges that the data transmission risk is too high and may decide to not send any sensitive data across the network. No data transmission session is initiated in this setting. This situation can be resolved when more low-risk resources become available at a later time, or if the risk quantification of network components changes.

In order to administer the algorithm on the network across various connected components, the following are steps that can be followed for continuous monitoring and operation.
1. Each SDN controller (C) is allowed to focus on an associated graph $G(V, E)$. This graph may change from time to time based on change in network structure, addition or removal of various network components, or even variations in network traffic and security.
2. In order to mark a certain data transmission for a low-risk requirement, the (s, d) pair may be tagged or the data packets involved may be tagged. This is a way for the controller to ensure that all other network components are able to identify that these data packets must adhere to a low-risk environment.
3. Ant Colony Optimization (ACO) provides a computationally cheap solution to path computations (since it uses approximation) for a given network. The controller may leverage ACO to compute optimized routes which adhere to privacy exposure and compliance constraints associated with the current data transmission. This may be specific to a packet or may be for a given (s, d) pair.

The primitive ACO algorithm only uses pheromones. For our problem setting, we also incorporate security risks for each edge of the graph. These risks may be accumulated over neighbors of several hops along a path in the network. This may also be extended to previous paths used by the network. Similar to pheromone values, quantified risk values also observe a gradual decay in value as time progresses. This allows the use of different paths for a given data flow in the network(which is a favorable feature in risk-associated environments).

In this study, we consider several parameters for analyzing the risk of routing of a data packet. The final output of risk analysis, risk ratio, allows the SDN controller to reconfigure the forwarding table on each switch. In order to avoid the computation overhead, we perform off-line computation, therefore the forwarding table is updated periodically.

A. Risk Parameters

First, we define several parameters for computing risk ratio as follows.
1. Meta-data of a data packet when it is routed to a switch and transferred to a destination address. This parameter including origin address, destination address, size of data packet, priority, in port, out port, and routing path.
2. Timestamp with respect to local timezone of the node or switch. This parameter associates to three nodes at each routing step: i) origin local time; ii) destination local time; iii) current switch local time. The time allows us to assess each data packet or a group of data packets based on the average rate for different times.
3. History of routing path for a specific origin and destination node. This parameter uses a time-stamp parameter to provide a comprehensive evaluation.
4. Data packet content. We randomly evaluate data packet if there is a chance of plain text data transfers through the network which might be performed by software application which is compliance unaware. Evaluation of data packet enables the risk analyzer to assess high risk applications that are not encrypting original data.
5. Type of data packet. This may correspond to different protocols, such as ARP, IP4, and IP6.
6. Data Integrity. We assess a data packet by reviewing minimum of two switches in a routing path. Data packets are randomly selected for data integrity evaluation. At the same time, the data packet may be forwarded to the destination and a copy could be submitted to Integrity Evaluator SDN controller that has all required information for additional review.
7. Type of data packet encryption. Packets may have different levels of security provided by different types of encryption.

B. Risk Computation

Second, we consider ACO to analyze online risk ratio and use a well-known clustering algorithm, $k - means$, to iteratively compute k centroid objects for our n criteria to find the risk ratio based on described parameters in Section V-A. Since this algorithm has more computation overhead than ACO, it is processed on the SDN controller in an off-line mode or it can be processed on an individual node and it modifies forwarding tables of the switches based on computed risk factors.

In the $k - means$ algorithm, let $X = \{x_1, x_2, ..., x_n\}$ be n elements of risk parameter objects, and $S = \{s_1, s_2, ..., s_n\}$ which is the set of our risk level clusters. The goal is clustering n parameters into S clusters of risk ratios.

The algorithm aims to partition n parameters into k clusters based on minimizing the Within-Cluster Sum of Squares (WCSS) which is a summation of distance function of each risk parameters to the risk level clusters. The WCSS for each s_i can be defined as follows:

$$\sum_{x \in S_i} ||x - \mu||^2 \tag{1}$$

If μ be the mean of cluster of s_i, therefore the risk ratio can be defined as follows:

$$arg\,min \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu||^2 \tag{2}$$

This algorithm has to perform two steps.

1) Assign each point to the closest cluster to generate a partition as follows:

$$s_i^{(t)} = \{x_p : ||x_p - m_i^{(t)}||^2 \leq ||x_p - m_j^{(t)}||^2, \forall j = 1, 2, ..., k \tag{3}$$

$m_i^{(t)}$ represents the mean value of i^{th} cluster

2) Compute the centroid of the new cluster as follows:

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j \tag{4}$$

In this paper, we consider five risk levels as follows: i) very low-risk; ii) low-risk; iii) normal; iv) high-risk; v) very high-risk. These risk levels correspond to cluster centers $\{s_1, s_2, s_3, s_4, s_5\}$, respectively. In this scenario the target of our algorithm is finding a lower risk ratio for each route to distributing data packets in SDN environment.

VI. Evaluation

We implement the risk analyzer in python by using scikit-learn library. We use 10 different features or data privacy parameters as input for analyzing the risk ratio with 5 different risk levels as described in Section V-A. Each feature represents a risk parameter to be used in the $k - means$ algorithm.we have implemented our log analysis model is an offline simulation process (using RYU), which however, can be implemented for online log analysis and risk computation. In this experiment, we considered $10,000$ sample log records which are generated randomly as follows:
1. Using an anisotropic distribution for 10 different features when we consider different time-stamp features. We transform the original feature of the time-stamp to the cluster-distance space. Let us consider different timestamps as the main key for this classification algorithm while using the other 9 features as parameters. In this scenario, we can divide each day into three periods as follows: 8:00AM-4:00PM, 4:00PM-1:00AM and 1:00AM-8:00AM. Therefore, we can evaluate all 9 features with respect to this time division. The result of this experiment is shown in Figure 1. Each record of $10,000$ sample records have been labeled with its risk ratio level. In Figure 1, we reduce the 10-dimension input to a 2D as shown points as X-axis and Y-axis. We also label each of the points based on their risk ratio with different colors when 10 features are considered. In this experiment, we found that any security and privacy issue between 4:00PM-1:00AM (the middle cluster represents this set of time-stamps) indicates a very high-risk factor. For instance, when a health-care application is using plain-text without encrypting the original content and using a plain-text protocol.
2. Using different variance for generating random values for 10 different features is shown Figure 2, the risk level for different cluster have been identified for all $10,000$ records; and finally 3. Using unevenly sized blocks to generate random values for 10 different features. As shown in Figure 3, the risk levels for different clusters have been identified for all $10,000$ records.

Considering different subjects for evaluating the risk in a SDN network allows the proposed method to provide a rigorous evaluation for strict compliance requirement network. In addition, implementing this method on each switch even for open switch is not feasible because the method requires heavy computation. However, when we are using a SDN controller we are able to collect data periodically from the switches and compute the risk factors for different data packets based on their origin, destination, time, and etc. The collected data can be reviewed by an external reviewer (risk analyzer) to decide which forwarding table is required modification. Our ACO method will be able to use this risk level tag to evaluate network edge weights and therefore forward data packets based on all considered risk parameters.

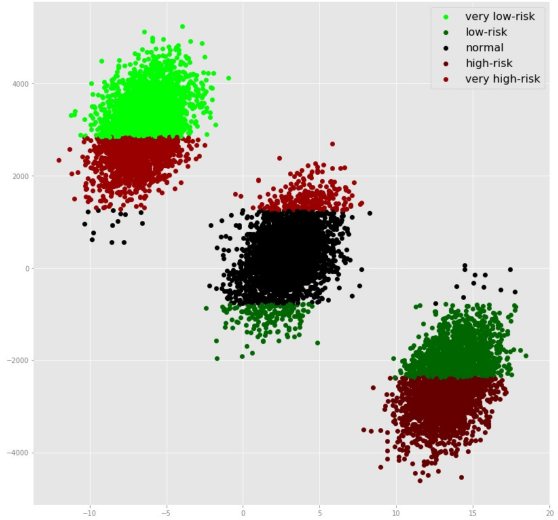

Fig. 1. Risk level for anisotropic distribution of 10 features. Multidimensional feature vectors are reduced to two dimensions.

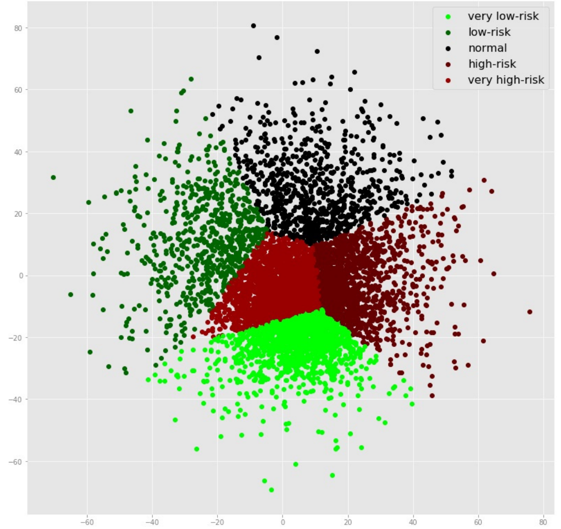

Fig. 2. Risk level for different variances of 10 features. Clusters of risk varied level data are observed to be well separated in this representation.

ACO is tested by evaluating the time durations of a series of random message interactions between hosts. This involves rule computation for new communications, as well as a hard timeout (time after which any rule is forcefully erased, if not erased already) by the controller. These are summarized in Figure 4. Experimental results are averaged over 10 runs. With the addition of risk and privacy constraints, the routing path selected may no longer be the shortest (because routing

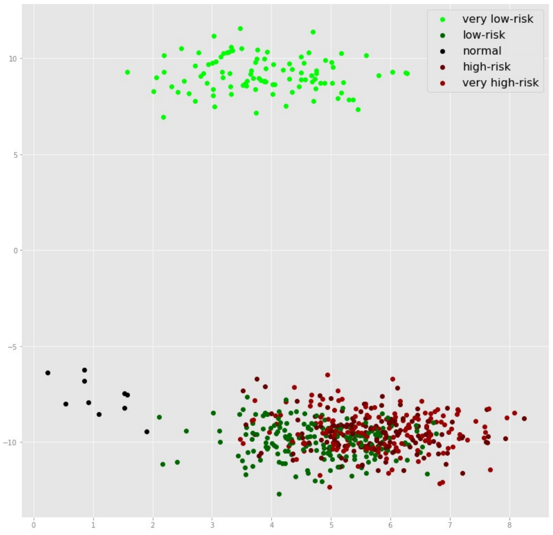

Fig. 3. Risk level for different 10 features when using unevenly sized blocks. Some overlap in clusters data items across risk levels is observed when analyzing all features.

preferences no longer depend solely on physical network distance).While an interaction with the controller (when a flow is not installed) takes significantly longer than bypassing it (when a flow is installed), the use of ACO still allows for lower complexity than other traditional algorithms (as explained in Section II). The time complexity for the controller can further be reduced by decreasing the amount of information being logged or retained, and optimizing the implementation. ACO is an inherently parallel algorithm and may therefore take advantage of parallel compute architectures.

VII. Conclusion

In this paper, we proposed a novel data privacy preservation method for SDN routing. In this routing algorithm, the SDN controller collects different features, such as: data packet types, network topology, and data packet routing history. We used two methods, first, risk analyzer that provides an offline process to compute the risk ratio and the second, an online method to make real-time decision for routing data packets. First, the SDN controller processes the data packet features by evaluating them based on a well-known machine learning method, $k-means$, and it classifies the data packet based on their level of risks according to HIPPA risk parameters. Second, we used ACO to provide a real-time decision for routing data packets because it provides a much lower time complexity solution than its traditional graph algorithm counterparts. The experimental results show an evaluation on 10 different risk parameters which is considered as high dimensional data. The risk analyzer is combined with ACO to provide a holistic solution to privacy and risk compliant associate to the whole SDN network.

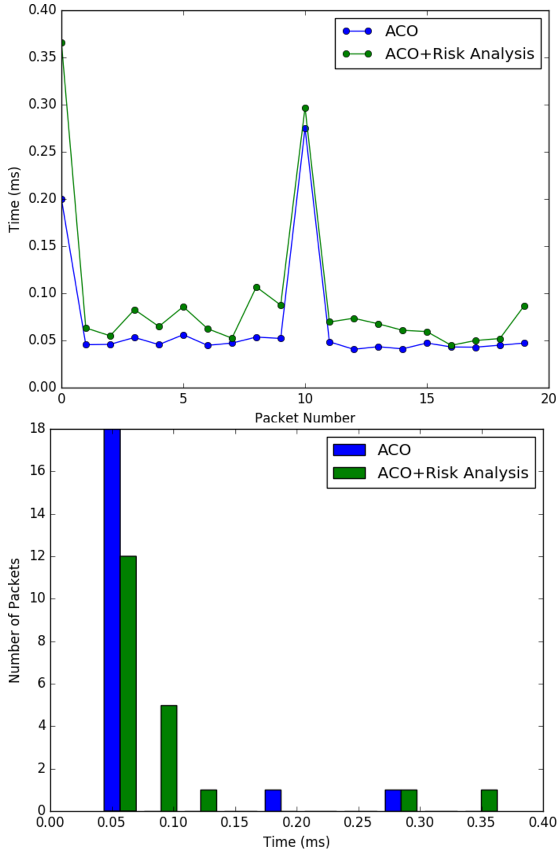

Fig. 4. Delays in communication (after the controller installs a flow) with and without risk analysis.

References

[1] N. McKeown, “Software-defined networking,” *INFOCOM keynote talk*, vol. 17, no. 2, pp. 30–32, 2009.

[2] K. Kirkpatrick, “Software-defined networking,” *Communications of the ACM*, vol. 56, no. 9, pp. 16–19, 2013.

[3] N. Feamster, “Software defined networking,” *Coursera*, 2013.

[4] S. Bailey, D. Bansal, L. Dunbar, D. Hood, Z. L. Kis, B. MackCrane, J. Maguire, D. Malek, D. Meyer, M. Paul *et al.*, “Sdn architecture overview,” *Open Networking Foundation, Ver*, vol. 1, 2013.

[5] O. Dobrijevic, M. Santl, and M. Matijasevic, “Ant colony optimization for qoe-centric flow routing in software-defined networks,” in *Network and Service Management (CNSM), 2015 11th International Conference on*. IEEE, 2015, pp. 274–278.

[6] M. Gunes, U. Sorges, and I. Bouazizi, “Ara-the ant-colony based routing algorithm for manets,” in *Parallel Processing Workshops, 2002. Proceedings. International Conference on*. IEEE, 2002, pp. 79–85.

[7] W. Guo, W. Zhang, and G. Lu, “A comprehensive routing protocol in wireless sensor network based on ant colony algorithm,” in *Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on*, vol. 1. IEEE, 2010, pp. 41–44.

[8] B. Pfaff, “Open vswitch manual,” *Manual, Open vSwitch*.

[9] M. Conti, F. De Gaspari, and L. V. Mancini, “Know your enemy: Stealth configuration-information gathering in sdn,” *arXiv preprint arXiv:1608.04766*, 2016.

[10] Q. Chen, C. Qian, and S. Zhong, “Privacy-preserving cross-domain routing optimization-a cryptographic approach,” in *2015 IEEE 23rd International Conference on Network Protocols (ICNP)*. IEEE, 2015, pp. 356–365.