EAI Endorsed Transactions
on Security and Safety
Research Article EAI.EU

Zero-Trust Based Distributed Collaborative Dynamic Access Control Scheme with Deep Multi-Agent Reinforcement Learning

Qiuqing Jin[1,2,*] and Liming Wang[1]

[1]Institute of Information Engineering, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences

Abstract

Vast majority of organizations and companies strongly depend on intranet with access control to achieve security data accessibility and authorized resource sharing across departments and networks. However, traditional boundary defense has difficulty in mitigating the increasing threats and attacks that mostly originated by insiders. Common insider threat solutions decouple the detection and defense, which requires domain knowledge and human intervention to achieve the mitigation after the protection. Moreover, these static methods have no capability to dynamically monitor various anomaly events and take corresponding protective measures. In this paper, we present a Zero-Trust based collaborative dynamic access control scheme to rebuild a security network architecture from the traffic scheduling perspective for insider threats mitigation. This scheme organically combines anomaly detection and mitigation execution by constructing dynamic updating user trust profile as the evidence of access control and collaboratively adjusting mitigation policy with any subtle requirement and environment changes in a scalable distributed way. We make use of the Multi Agent Deep Deterministic Policy Gradient (MADDPG) to optimize the traffic allocation policy for adaptive and automatic collaborative management scheme with the consideration of network security, network environment and user requirement. The performance of the scheme is analyzed through a network simulator, which shows promising results for DRL to be applied in threat mitigation.

Keywords: Zero-Trust, Insider Threats, Dynamic Access Control, Reinforcement Learning.

Received on 12 November 2020, accepted on 13 December 2020, published on 16 December 2020

doi: 10.4108/eai.25-6-2021.170246

* Corresponding author. Email: jinqiuqing@iie.ac.cn

1 Introduction

With the growing prosperity of network, a considerable number of production and business strongly depend on network infrastructures which interconnect servers, hosts and other electronic devices across the organizations and companies. In order to guarantee the network and information security, most companies and enterprises are aimed to construct an absolute safe internal network by maintaining a solid and sophisticated network boundary, such as firewall which integrates numerous security tools and access control rules. However, according to the 2019 Insider Threat report [1], 70% respondents (organizations) observed that insider attacks have become more frequent and 60% experienced one or more insider attacks against their organization over the last 12 months, which strongly proves that insider threat is quite a rampant and challenging problem for cybersecurity researchers. With changes like mobility and big data, “building stronger walls” becomes an expensive farce that will not adequately protect networks. Furthermore, once the attackers have intruded the internal network, they have the access to any resources and achieve the sabotage goals easily.

Traditional insider threats solutions generally decouple the problem as anomaly detection and defensive execution, which disconnect the relation between these two phases and increase extra human costs in eliminating the threat after detection. The sectional defense architecture shows the dependence of cybersecurity

analysts to take a deep insight of anomalies and make a decision according to analysis results. Moreover, traditional methods often take a targeted static offline detection and prevention way which has difficulty in monitoring and tracing any form of intranet user behavior anomalies in real time so as to counter, forewarning and take immediate response against diversified threats and attacks, causing a significant loss to organizations or companies because of the processing delay after the fact.

As we investigate, the ultimate goal of insider threats is actually sabotage and theft of security user access and shared resources. However, as one of the basis pillar of network operation, traffic resources crucially support every service and application in the network, even including the unauthorized request and access. Therefore, manage user behavior and network resources from traffic resources perspective will be an effective and feasible way to mitigate insider threats from the beginning.

Zero trust fundamentally builds networks from the inside out rather than merely overlay existing networks with more and more controls in an attempt to create a semblance of a secure network and better protect valuable information while allowing for free interactions internally. The main idea of zero trust is to abandon the traditional operating mechanism which is designed for bygone network era, have no trust for any requests and de-perimeter the boundary of a trusted network (usually the internal network) and an untrusted network (external networks). This mechanism ensures all resources are accessed securely regardless of location, adopts a least privilege strategy and strictly enforces access control by inspecting and logging all traffic in the network. In essence, zero-trust model is centralized dynamic access control and management of network resources.

With the various uncertainty of network dynamics, it is quite intricate and challenging for balancing the limited network resources and massive request with a considerable resource management policy which can dynamically adjust scheduling scheme according to the network situation. Reinforcement Learning[2] is a quite suitable and adaptive way to simply declare high-level control objectives or reward functions and have networks learn to manage themselves from the large amount of data they have already collected[3]. In this paper, we present a zero-trust based dynamic network resources access control and management scheme which can timely trace the user behavior and achieve dynamic network resources access and scheduling based on the user trust and current network dynamics in a distributed architecture.

The key contributions of this paper can be stated as follows:

- A zero-trust based distributed collaborative dynamic access control scheme is proposed through continuous visibility of user behavior, flexible centralized control policy and automatic response workflow. The scheme overturns the traditional boundary security defense by redefining the conventional way of user authentication and authorization and maintaining an automatic response execution policy to adaptively reallocate the traffic resources in the intranet.
- A full-lifecycle insider threat solution is presented by integrating user trust profile construction and self-adaptive traffic control in a distributed architecture, which can dynamically update user trust profile according to user behavior and appropriately adjust scheduling policy with the comprehensive considerations of any subtle changes of network environment, including network security, user requirements and network performance.
- A state transition prediction model and a nested random noise are designed for assisting the agent learning process by constructing a LSTM-based time sequential model to capture the environment transition dynamics and combing the monotonic decreasing log function with random function, respectively.

The remainder of the paper is organized as follows. Section 2 identifies related works. Section 3 provides a detailed description of our integrated access control scheme. A meticulous analysis and effectiveness evaluations of the proposed scheme with respect to several objectives is presented in Section 4. Finally, Section 5 provides discussion and conclusions.

2 Related Work

With the growing dependence of intranet for governments, enterprises and organizations, numerous works have been done on solving insider threats problems [4-14]. Majority of researches take the advantage of sectional defense architecture and show the unbalance attention between anomaly detection and defensive execution. Indeed, most of the schemes, systems, and conceptual frameworks surveyed above focus primarily on detection rather than prevention.

The common detection method can be mainly classified into two groups: feature matching based methods and model constructing based methods. Feature matching based methods[4] generally analyze and extract event feature from the existing attacks data, then compare with the unknown events to diagnosis the anomalies. These methods are apparently expeditious and effective, but has the limitation to detect unknown threats and the stringent requirement of data and manpower. Model constructing based methods[5] commonly construct anomaly detection models through machine learning algorithm, which have the capability to roundly detect anomaly behaviors and events. However, most above methods are static detection methods that rely on the offline analysis and classification of event and behavior data, which have no ability to achieve realtime monitor and control of user behaviors and pay enough attention to any subtle trace of threats and attacks especially in the early phase.

In contrast to the anomaly detection methods, defensive methods[6] tend to be more straightforward and simple by restricting or rejecting user and traffic which satisfies the fixed anomaly rules, that lack of the thorough thinking and long-term arrangement of overall defense scheme. Even they have good performance in specific scenarios, the intrinsic characteristics of pertinence has high requirements to expert knowledge and solution deployment, which leads to difficulty to transfer to other application scenarios while facing the dual challenge of high false positive rate and dynamic defensive demands.

In recent years, reinforcement learning application researches in cybersecurity gradually spring up. Servin et al. [7] present a reinforcement learning based anomaly detector to defend flooding-base DDoS in the simulated network environment with a controllable anomaly injection, rewarding with anomaly detection accuracy. Although the network environment has been physically simulated, lookup table based Q-learning algorithm and manipulated reward rather than extracted from environment observations expose its impracticability. Meanwhile, Xu et al. [8, 9] and Sukhanov et al. [10] also face the above challenges and dilemma, resulting from adopting lookup table based temporal difference algorithm [11] for online intrusion detection to traffic flow. Servin et al. [12] and Malialis et al. [13] improve the detection methods by introducing reinforcement learning algorithm to multi-agent architecture for constructing hierarchical anomaly detection framework. However, their capability still restricted by the limited number of lookup table, thus having difficulty in intrusion detection in continuous state space. Caminero et al. [14] present an adversarial reinforcement learning based intrusion detection algorithm which breaks through the above restriction of discrete state space, models environment dynamics into learning process, adjusts the prediction difficulty of classifier according to prediction results, and gradually improves detection accuracy though the game between agent and classifier.

To summarize the above studies, most applications collaborate with machine learning based classification by playing an auxiliary and augment role. In essence, they are also machine learning based anomaly detection methods for lack of integrated cybersecurity defense scheme, thus having numerous obstructions in practical use.

3 Methods

3.1 Scheme Architecture

With the continuous development and popularity of cloud computing and mobile communication, the ultimate goal of zero trust framework is to protect data across an increasingly fragmented information fabric, which not only provide right-level security across all the access points, but also maintain the seamless user experience and work efficiency of employees. More specifically, zero trust model requires continuous discovery, monitoring, assessment and risk prioritization to maintain continuous device visibility, integrate security tools and information sources to realize self-adaptive access and defense for policy execution, thus superseding traditional manual analysis and application with synergetic and responsive automatic workflow.

With the explosion of network equipment and users, computer systems and applications have increasing complexity, which exposes the system and network to more vulnerability that easy to be exploited by attackers and make the attacks defense more complicated and difficult. Comparing with the targeted countermeasure of after-thefact defense to threats and attacks, intranet user management and control from the origin can essentially achieve the traceability and threats mitigation. Combining with zero trust model, our access control scheme is designed as a full-lifecycle process by integrating online anomaly detection and intranet self-adaptive traffic control model in a distributed network architecture, which can operate automatically without any human intervention.

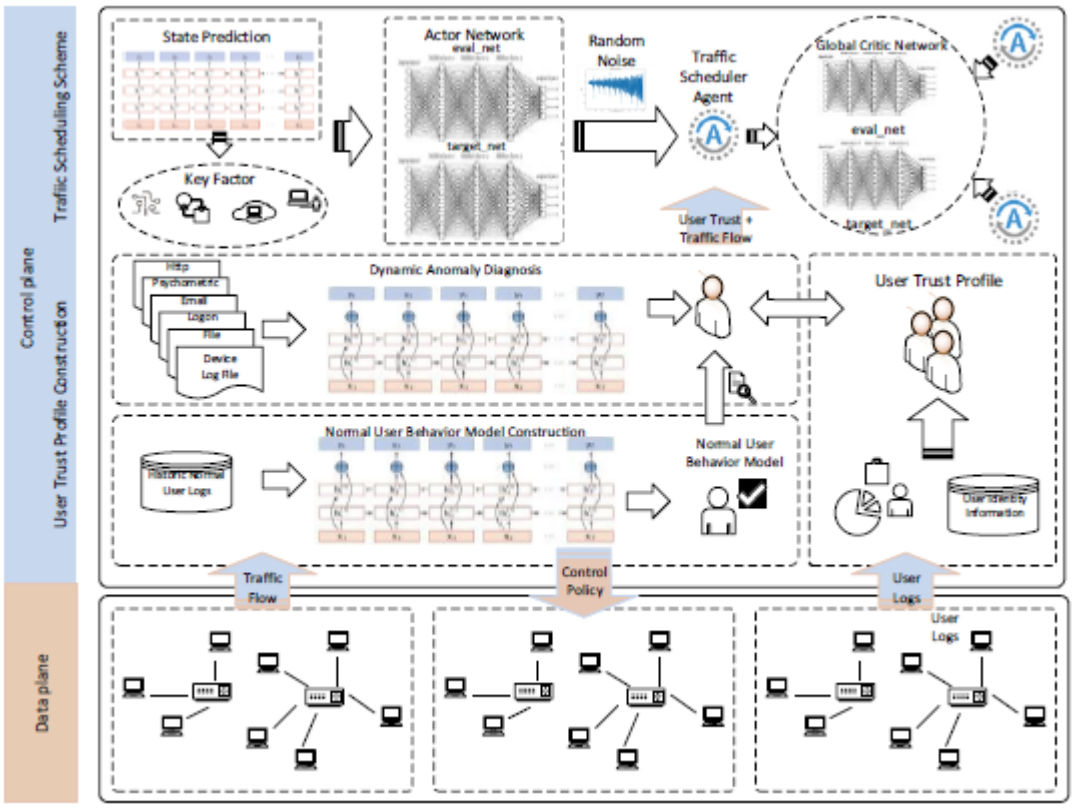

Fig. 1. Architecture description of dynamic access control scheme.

As shown in Figure 1, raw user behavior logs will be collected from data plane, which will be aggregated to feed into online log parser for user trust identification. Combining with the requested traffic volume, user trust will be conveyed to reinforcement learning based controller in the control plane. The controller will comprehensively consider all the key factor in the network, including network security, user requirement, network performance and scheduling policies from other controllers, to generate a traffic scheduling policy that will be executed in the data plane to adjust the current network access control conditions and balance the traffic load and supply. Moreover, user behavior logs will be continuously updated for further user profile construction and next policy adjustment for facilitating the network convergence to the most appropriate access control scheme. The learned scheme will learn to manage itself according to the subtle dynamic changes without any domain knowledge or extra manual instruction.

Most intranets still adopt static certificate authority yet which has no ability to trace and defend insiders, such as authorization logon or remote VPN login. Moreover, most existing resources access and sharing credentials have relied on identity information of users for hierarchical classification, namely department, age, position, salary, relatives with others, etc. This classification belongs to a passive artificially defined grading method which has difficulties to realize dynamic updating and adjustment in real time and has strong dependence on manual operation. Furthermore, it has obstruction to conduct prognosis and timely control to insider threats, which is a great challenge to detection and defense of insider threats.

To address the above difficulties and challenges, we drew inspirations from zero trust model and redefine the mechanism of user certificate authority by taking the online user trust degree as the basis credential of user access, resources sharing and management, thus achieving the timely tracing and monitor for flexible self-adaptive intranet traffic allocation.

3.2 User Trust Profile Construction

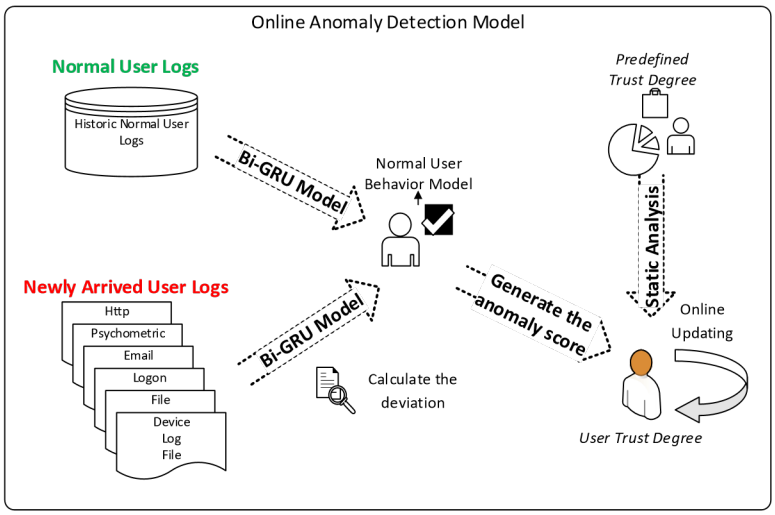

Fig. 2. Online anomaly detection model description.

As the evidence of user history behavior security degree, constructing an accurate user trust profile is the fundamental module in scheme generation. In order to make a convinced access authority measure, we construct an online anomaly detection model by using time sequential modeling methods. As shown in Figure 2, we first use identity information of users to analyze the basic user attack intents, acting as the prior knowledge of user trust profile for adequate user trust identification. Furthermore, we take the advantage of the daily user behavior logs to draw the trace of user behavior for anomaly diagnosis by adopting one of the advanced online network log anomaly detection methods, bidirectional Gated Recurrent Unit based network, to realize an unsupervised sequential language model. This model then produces a user trust which represents the user behavior deviations from normal execution, and keeps continuous online updating according to new log data. During the phase of model training and application, this model has no necessity for feature engineering and choreographed threshold which enable it to become a general detection method and has the capability of transferring to most anomaly detection scenarios with vast online data. This model achieves superior performance in such scenarios comparing with other existing methods, which has been amply proved in our previous work.

3.3 Self-adaptive Traffic Control

Network modeling. In our model, we assume that the total bandwidth resources in the network is $\mathbf{B} = (b_1, b_2,...,b_n)$, and the requested flow comes at each timestep with requested traffic volume and user trust $F = (f_1, f_2,..., f\infty)$, where $f = (p,v)$. All the unprocessed traffic flow composes the global network flow waiting queue, and the agents will transmit a batch of traffic flow at each timestep (as shown in Fig. 3). All the terminology of network model is shown in table 1.

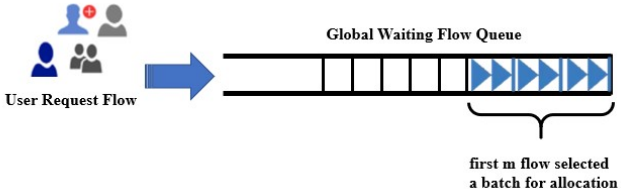

Fig. 3. Global waiting flow queue.

Table 1. Terminology of network model.

Variable	Description
$\mathbf{B} = (b_1, b_2,..., b_n)$	Link bandwidth
$F = (f_1, f_2,..., f_\infty)$, $f = (p, v)$	Network flows
p	User trust
v	Requested traffic volume
$S_t = [O_t, W_t]$	Network State
$O = (o_1, o_2,..., o_n)$	Idle link bandwidth
$W = (f_1^w, f_2^w,..., f_\infty^w)$	Network flows waiting queue
m	Transmission batch size
$\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2,..., \mathbf{a}_m)$	Action
$U_t = (u_1^t, u_2^t,..., u_n^t)$	Link utilization
V_i^t	User preference
T_i^t	Transmission duration
L_t	Network flows waiting queue length

Problem Formulation. The dynamic access control problem addressed in this paper can be formalized as a dynamic resources scheduling problem based on network security in a self-adaptive and full-lifecycle manner. In order to guarantee the rationality of access control operation, we normalize the control policy by user trust profile, which is constructed through user daily behavior log, and each user profile quantizes the anomaly degree of user behaviors that deviate from normal users. Furthermore, we deploy our scheme in a distributed and online framework to pave the way in improving the scalability and automation capability. Our ultimate goal is

to achieve impressive mitigation effect on insider threats problem by overthrowing the traditional way of detection and protection from inside out to tactfully confront and evade the challenges.

Reinforcement learning is a paradigm in which a decision-making agent improves itself by continuously trial-and-error searching and learning from interaction with environment and accumulated experiences, which reflects by a reward function. We now represent our problem as a discrete-time, continuous state and action space cooperative stochastic game MDP as a tuple <K, S, J, T, R, χ>, where

K is the set of agents, $|\mathbf{K}| = k$.

S is the set of state, and state of each timestep t can be represented as $\mathbf{S}_t = [\mathbf{O}_t, \mathbf{W}_t]$, where $\mathbf{O}_t$ is defined as the rest available link bandwidth and $\mathbf{W}_t$ is defined as the global network flow waiting queue state $W = (f_1^w, f_2^w, ..., f_\infty^w)$ at current.

J is the set of all joint actions $\mathbf{J} = \mathbf{A}_1 \times \mathbf{A}_2 \times \ldots \times \mathbf{A}_k$, each $\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m)$ represent the traffic allocation policy of each agent to each link.

T is the set of conditional transition probabilities $\mathbf{T} : \mathbf{S} \times \mathbf{J} \times \mathbf{S} \rightarrow [0,1]$.

R is the immediate reward function $\mathbf{R} : \mathbf{S} \times \mathbf{J} \rightarrow \mathbf{R}$ calculated by Equ.1, which is based on the accumulation of weighted sum of network link utilization $\mathbf{U}_t = (u_1^t, u_2^t, ..., u_n^t)$, delivered traffic volume weighted by user trust profile (V_i^t), transmission latency (T_i^t) and the length of the waiting flow queue (L_t) at current time t :

$$R(s_t, a_t) = \alpha \sum_{i=0}^{n} u_i^t + \sum_{i=0}^{m} (\beta V_i^t - \gamma T_i^t) - \delta L_t \qquad (1)$$

where α, β, γ, δ are adjustable parameters for controlling the management preference.

$\gamma < 1$ is the discount factor.

In this paper, we use multi-agent deep deterministic policy gradient (MADDPG) to address the inherent problems of traditional reinforcement learning algorithm in the distributed architecture, namely independent learning and policy optimization leads to unstable environment dynamics which results in having difficulty in convergence. While overcoming the requirements of environment dynamics model and special communication, MADDPG has learned optimal policy to execute optimal actions in the cooperation or competition application when just depends on local information. Derive from deep deterministic policy gradient (DDPG) algorithm, MADDPG integrates centralized training, distributed execution, modified experience replay and policy integration optimization to achieve superior performance.

Policy promotion. In order to orient MADDPG to our application, we fine-tuning the algorithm and ameliorate state presentation and random noise by constructing state transition prediction model and designing a new appropriate noise simulation function, respectively.

State transition prediction. In the distributed multi-agent coordination environment, each agent suffers from the ignorant about next actions of other agents for further decision-making due to the independent optimization. Therefore, we draw support from advantages of long-short term memory network in time sequential applications to construct state transition prediction model. This model roughly predicts next environment state according to current state by learning and training model transition dynamics though offline data. Agents will adaptively adjust current policy for converging to optimal according to the above prediction.

In this model, we take the states of all links at each timestep $s_i = (s_i^1, s_i^2, ..., s_i^n)$ as the input and take the states of all timesteps in one episode $\mathbf{S} = (s_1, s_2, ..., s_t)$ as a sequence input. During the training process, the mini-batch is applied for steady and efficient convergence. Every prediction result that trained and learned with cross-entropy loss and back-propagation through time, is directly took as output.

According to the next predictive state from trained state transition prediction model, agents will selectively circumvent the links which are about to surpass or have surpassed the predefined link utilization threshold and reduce the allocation intensity to those links, conversely exploiting the bounteous links for sensible action choosing and execution, which is calculated by Equ.2.

$$\mathbf{A} = \max\{(\mathbf{G} - \frac{\mathbf{P}}{\mathbf{B}}) \times \mathbf{A}, 0\} \qquad (2)$$

where **G** is the predefined network congestion threshold, **P** is the prediction results of each link state.

Random noise. It is quite frequently occurs in the training process of reinforcement learning that the actions tend to be stuck in a fixed range, leading to local optimal and slow learning. To increase the randomness so as to expand the coverage of learning process and further balance the exploration and exploitation of the algorithm, we introduce a self-designed random noise to replace the original noise of DDPG, namely Ornstein Uhlenbeck random process:

$$\mathbf{A} = \pi_\theta(S) + \mathbf{N} \qquad (3)$$

The noise **N** consists of nested log function and random function, takes the advantage of monotonic decreasing property of log function for encouraging widespread exploration and weakening data dependency in the preliminary stage of training, gradually reduces the exploration proportion when converges to optimal policy, and introduces the random disturbance to simulate the random noise. Comparing the OU process, this simulated noise is more applicable in our scenario, results from its time sequential property which progressively decreases with the episode i and timestep j , calculated by Equ.4:

$$\mathbf{N} = rand \log(\frac{MAX\_EPI + 1 - i}{14} \times \frac{\log(MAX\_STEP + 1 - j)}{5}) \qquad (4)$$

where MAX_EPI is the maximum of episode, MAX_STEP is the maximum of timestep in every episode. *rand* represents random disturbance which has been limited in a certain range, calculated by Equ.5:

$$rand = \begin{cases} 10 \times random - 3, 0 \le i < 50 \\ 5 \times random - 2, 50 \le i < 200 \\ 2 \times random - 1, 200 \le i < 500 \\ 1 \times random - 0, 500 \le i < 1000 \end{cases} \quad (5)$$

where *random* is random number that ranges in [0,1].

4 Evaluations

Our work is implemented with Tensorflow[15]. In this section, we evaluate and analysis the overall performance of scheme based on the following major criteria:

- Fitting degree of state transition prediction model to environment dynamics.
- Coordination traffic scheduling performance of MADDPG agents.
- Insider threats mitigation performance of the whole scheme.

In this work, we build a network simulator for insider threats scenario and implement three agents to have a cooperative game competition for all intranet traffic resources. All the simulated network setting will be reset after the predefined duration for a new training episode. Each single agent has two hidden layers of 400 and 300 units, learning rate of 10^{-3} and 10^{-4} for the actor and critic respectively, and the minibatch size is 64. The soft replacement is 0.001, the reward discount factor is 0.99 and the replay buffer size is 10^4. As for state transition prediction model, the training data is extracted from intermediate state of network environment that is influenced by trained MADDPG agents. The LSTM based network has a single hidden layer of size 128, the learning rate is 0.001, and the minibatch size is 4. Adam[16] optimizer is applied to learn the neural network parameters in all of networks.

4.1 State Transition Prediction Fitting Degree

To evaluate the fitting performance of state transition prediction model to state transition dynamics in the real world, we record prediction results and real condition of state transition in the learning process at every fixed time interval. Figure 4 illustrates that state transition prediction model rapidly converges to accurate fitting result after short learning phase, and certainly perform random prediction that deviate far from ground truth in early phase. This accurate state transition prediction model successfully captures the real transition pattern and almost simulates the real transition dynamics, which provides stable support and help for MADDPG agents.

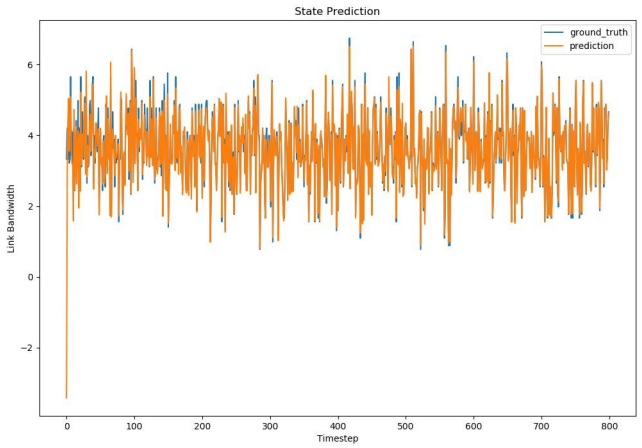

Fig. 4. State Transition Prediction Curve.

4.2 Coordination Traffic Scheduling Performance

As we investigate that, DDPG algorithm has significant capability in most continuous action and state application, including cybersecurity and resources allocation. In order to validate the advancement and practicability of our agents and their improvement policy, we adopt standard DDPG agent to compare with primitive MADDPG agents (without any improvement policy), MADDPG agents with random noise, MADDPG agents with state transition prediction, integrated MADDPG agents with random noise and state transition prediction. All the above agents will be set with the same parameters, trained and validated in the same environment for a fair comparison. Since the comprehensive gain, scheduling efficiency and network performance is the most important capability of our scheme, we take the reward, waiting flow length and utilization satisfaction as evaluation metrics in this subsection.

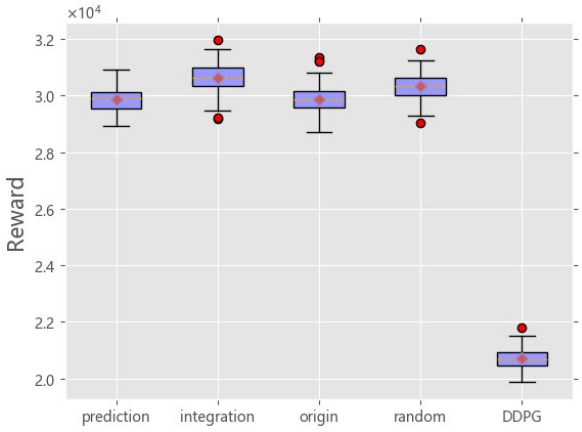

Fig. 5. Model comparison of reward gain.

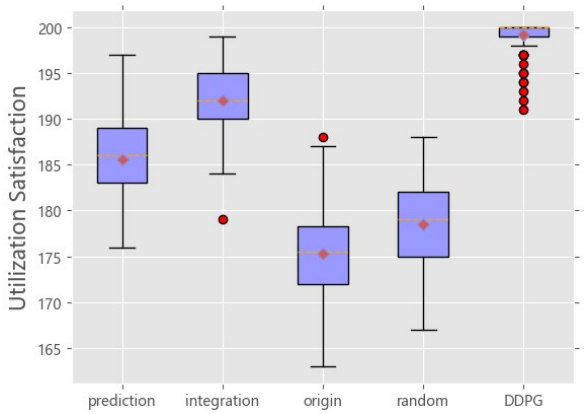

Fig. 6. Model comparison of utilization satisfaction in 200-episode setting.

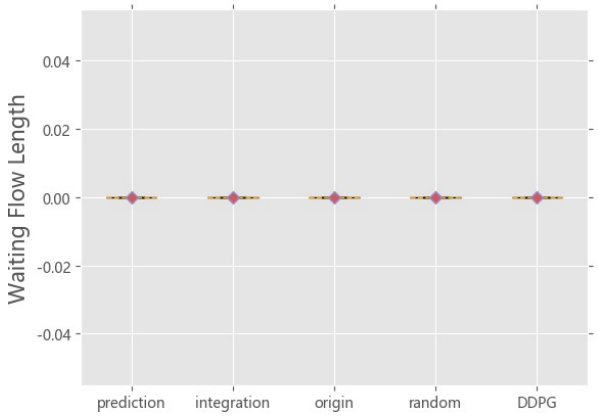

Fig. 7. Model comparison of waiting flow length under the overload network environment.

Figures 5, 6 and 7 show the optimal overall performance of integrated MADDPG agents because of the organic combination of state transition prediction and random noise, followed by MADDPG agents with state transition prediction, MADDPG agents with random noise and primitive MADDPG agents. Although there is not wide difference between all the agents, each agent still has its own advantages in different part except integrated MADDPG agents that have superior performance in every metric. MADDPG agents with state transition prediction shows the superiority in utilization satisfaction by adjusting the network actions to meet the requirement of network congestion threshold based on the prediction of next network state. Due to the excess random factors and anthropogenic factors in our application scenario, it is difficult to make a complete accurate state prediction in an ever-changing environment even the state transition prediction model achieves high performance in training process. Therefore, the improvement provided by prediction model is actually quite limited. On the contrary, MADDPG agents with random noise has the advantage in waiting flow length through sufficient exploration of action, thus accelerating the process to policy optimal and correspondingly gaining higher reward. Integrated MADDPG agents balance the exploitation and exploration by combining the above two improvements, which provides certain promotion and optimization to comprehensive performance.

Furthermore, there is significant performance gap between primitive DDPG agent and all the MADDPG agents, which proves that MADDPG algorithm can perfectly model multi-agent game and network environment to gain the maximum long-term reward with the requirements of network security, user requirements and network scheduling performance. However, as shown in Figure 6, standard DDPG agent slightly oversteps the integrated MADDPG agents in utilization satisfaction, demonstrating the superiority in network resources management. We assume that integrated MADDPG agents promote network scheduling efficiency to gain more accumulated reward by sacrificing the utilization satisfaction to reduce transmission delay for long-term comprehensive considerations.

Moreover, Figure 7 indicates that every agent has high performance in network scheduling efficiency no matter MADDPG agents or DDPG agent, responding to all flow requests and transmission in limited time by leaning to appropriately adjust the allocation policy even in a tough network environment which tends to be congested.

4.3 Insider Threats Mitigation Performance

In order to validate the insider threats mitigation effectiveness of collaborative dynamic access control scheme built through multi-agent cooperative scheduling in a distributed architecture, we sample two groups of representative users from normal users and attackers in the network simulator respectively, then mark and trace their behavior influence to allocation policy execution for taking the deep and meticulous insight of security access control operation. We also take DDPG agent as baseline method to further analyze mitigation performance.

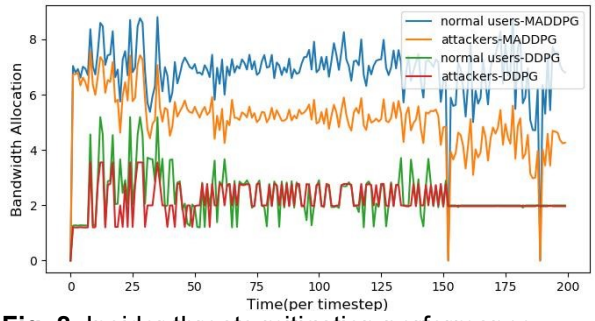

Fig. 8. Insider threats mitigation performance.

Figure 8 plots the bandwidth supply conditions of different users under different agents control and demonstrates that MADDPG agents have significant

performance advantage in insider threats mitigation compared with DDPG agent. Furthermore, all the agents perform varying degrees of resources restraint and gradually widen the disparity of different users, which throttle the traffic supply of attackers(who has suspicious behaviors and lower user trust) and satisfy the request from normal users(who has legitimate routine and higher user trust). Due to the environment complexity and cooperative game of multi-agent, MADDPG agents scheduling policy tends to be polytropic and flexible, which opposed to limited fluctuation of DDPG agent.

5 Conclusion

In this paper, we propose a zero-trust based distributed collaborative dynamic access control scheme to mitigate insider threats problem from traffic resources scheduling perspective. Combing with zero-trust model, this scheme provides self-adaptive, fulllifecycle and automatic insider threats solution through continuous visibility of user behavior, flexible centralized control policy and automatic response workflow in a distributed architecture. Extensive experimental evaluations prove the practicability and effectiveness of the whole scheme and the joint optimization of each component. The scalable self-management security network architecture achieves effective insider threats mitigation without human intervention, breaks through the traditional thinking of sectional defense and provides a new feasible alternative for future insider threats application, which distinguish itself from other solutions with the manual requirements.

References

[1] 2019 Insider Threat Report. https://haystax.com/wp-content/uploads/2019/07/Haystax-Insider-Threat-Report-2019.pdf, last accessed 2020/3/20

[2] R. Sutton, and A. Barto. Reinforcement learning: an introduction. MIT Press. 1998.

[3] S. Chinchali, P. Hu, T. Chu, and et al. “Cellular network traffic scheduling with deep reinforcement learning,” in AAAI. New Orleans, pp. 766-774, 2018.

[4] S. Roy, A. C. König, I. Dvorkin, and et al. “Perfaugur: Robust diagnostics for performance anomalies in cloud services,” in International Conference on Data Engineering. IEEE, pp. 1167-1178, 2015

[5] S. He, J. Zhu, P. He, and et al. “Experience report: System log analysis for anomaly detection,” in International Symposium on Software Reliability Engineering (ISSRE). IEEE, pp. 207-218, 2016.

[6] L. Li and J. Y. Luo. “Research and implementation on access control for Intranet terminal based on policy,” 2009.

[7] A. Servin. “Towards traffic anomaly detection via reinforcement learning and data flow, ” in Department of Computer Science, University of York, United Kingdom, 2007.

[8] X. Xu. “Sequential anomaly detection based on temporal-difference learning: Principles, models and case studies,” in Applied Soft Computing, pp. 859-867, 2010.

[9] X. Xu, T. Xie. “A reinforcement learning approach for host-based intrusion detection using sequences of system calls,” in International Conference on Intelligent Computing. Springer, Berlin, Heidelberg, pp. 995-1003, 2005.

[10] A. V. Sukhanov, S. M. Kovalev, V. Stýskala. “Advanced temporal-difference learning for intrusion detection,” in IFAC-PapersOnLine, pp. 43-48, 2015.

[11] R. S. Sutton. “Learning to predict by the methods of temporal differences,” in Machine learning, pp. 9-44, 1988.

[12] A. Servin, D. Kudenko. “Multi-agent reinforcement learning for intrusion detection,” in Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning. Springer, Berlin, Heidelberg, pp. 211-223, 2005.

[13] K. Malialis. “Distributed Reinforcement Learning for Network Intrusion Response,” in University of York, 2014.

[14] G. Caminero, M. Lopez-Martin, B. Carro. “Adversarial environment reinforcement learning algorithm for intrusion detection,” in Computer Networks, pp. 96-109, 2019.

[15] M. Abadi, P. Barham, and et al. “TensorFlow: a system for large-scale machine learning,” in OSDI, Savannah, GA, USA, 2016, pp. 265-283.

[16] D. P. Kingma, and J. Ba. “Adam: A Method for Stochastic Optimization,” Computer Science, 2014.