Ya-wen WANG, Jiang-xing WU, Yun-fei GUO, Hong-chao HU, Wen-yan LIU, Guo-zhen CHENG, 2018. Scientific workflow execution system based on mimic defense in the cloud environment. *Frontiers of Information Technology & Electronic Engineering*, 19(12):1522-1536. https://doi.org/10.1631/FITEE.1800621

Scientific workflow execution system based on mimic defense in the cloud environment

Key words: Scientific workflow; Mimic defense; Cloud security; Intrusion tolerance

Corresponding author: Jiang-xing Wu
E-mail: JiangXing_WU_NDSC@163.com

Motivations

It is necessary to enhance the security of scientific workflow execution in cloud computing environments, the reasons are as follows:

1. The execution duration is long. A practical cloud workflow task often requires several weeks or even months to be finished , which provides sufficient preparation time for attackers.

2. Cloud workflows are computationally intensive, which require high computing accuracy. Any intermediate error will directly lead to a failure of the entire cloud workflow execution.

3. Cloud workflows are often applied in important scientific research fields. Once the execution results are tampered with without being noticed, it will bring incalculable losses.

Main ideas

1. For heterogeneity, the diversity of physical servers, hypervisors, and operating systems are integrated to build a robust system framework.

2. For redundancy, each sub-task of the workflow will be simultaneously executed by multiple executors. Considering efficiency and security, a delayed decision mechanism is proposed to check the results of task execution.

3. For dynamics, a dynamic task scheduling mechanism is devised for switching workflow execution environment and shortening the life cycle of executors, which can confuse the adversaries and purify task executors.

Methods

1. We constructed the framework of the mimic cloud workflow execution system based on the diversity of physical servers, hypervisors, and operating systems.

2. We used heterogeneous task executor cluster and delayed decision mechanism to achieve intrusion tolerant workflow execution.

3. We devised a dynamic workflow execution environment switching strategy, which can confuse the adversaries.

Major results

1. The increase in the number of executors will effectively reduce the probability that the system produces erroneous results, but at the same time, the probability of sub-task re-execution will be increased, which will cost extra resources.

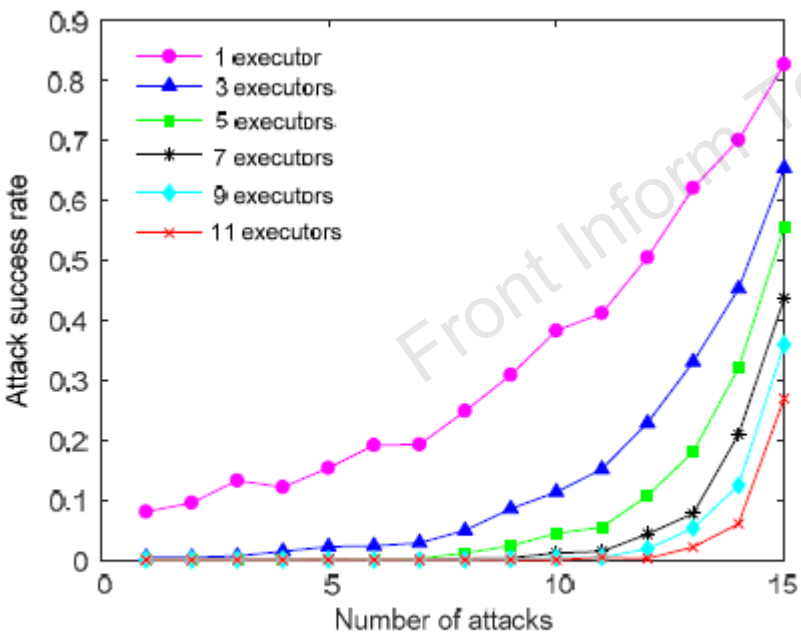

Fig. 5 Relationship between the number of executors and the attack success rate

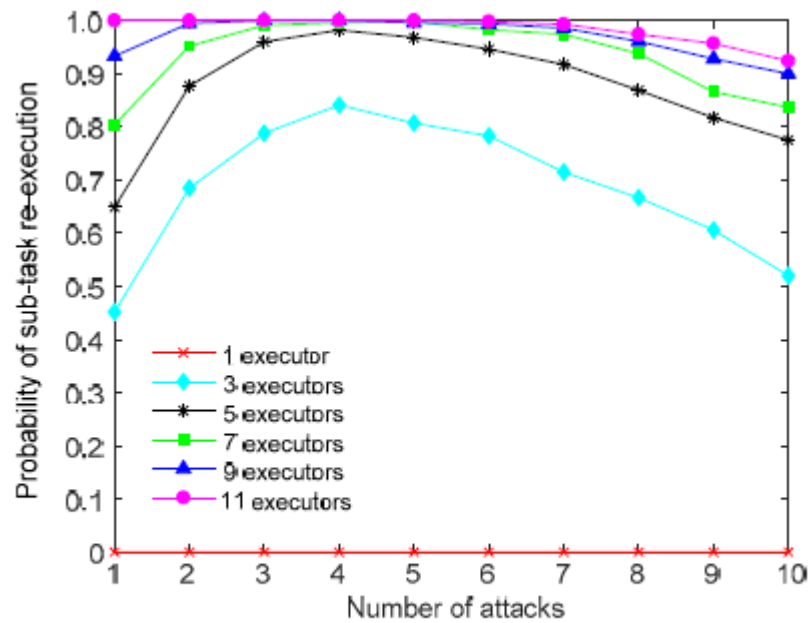

Fig. 6 Relationship between the number of executors and the probability of sub-task re-execution

Major results

2. Dynamic executor generation and recycling strategy can reduce the number of received attacks.

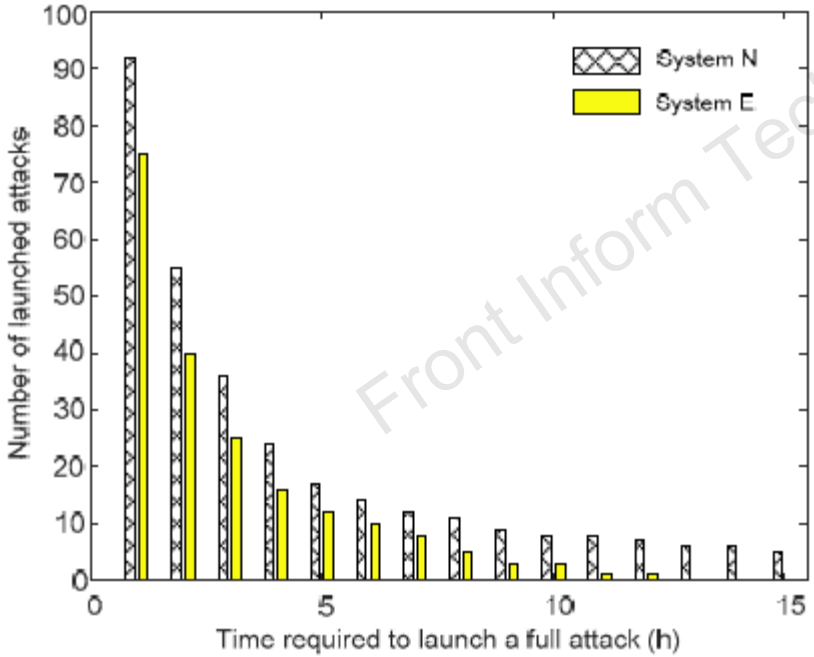

Fig. 9 Comparison of the number of launched full attacks against the two systems

Conclusions

1. The diversity of physical servers, hypervisors, and operating systems was introduced to build the intrusion-tolerant framework.

2. Based on the framework, common vulnerabilities among different operating systems were used for heterogeneity measurement and executor deployment.

3. With the flexible resource management of the cloud computing platform, the elastic executor generation and recycling mechanism was proposed, which can not only shorten the life cycle of executors, but also act as the clean mechanism to keep the pure state of executors.