Towards Automation of Penetration Testing for Web Applications by Deep Reinforcement Learning

Hajime Kuno and Kanta Matsuura(The University of Tokyo)

Introduction

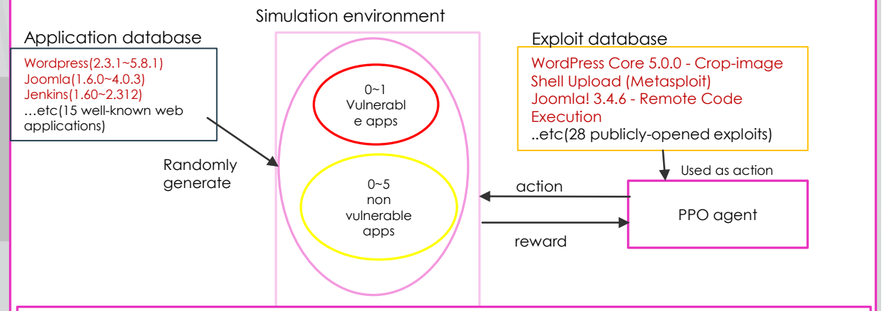
Penetration testing (PT) that assesses vulnerabilities by considering and executing all possible attacks is important in security engineering but very expensive due to the need of experienced professionals.

As a countermeasure, there are attempts[1]–[4] to partially automate and improve the efficiency of PT. Such approaches do not embed ML in PT tools, and would not improve the tools themselves.

In this work, we use **deep reinforcement learning** to automate search and exploit executions for various vulnerabilities existing in Web applications so that a wide variety of PT tools can be integrated in an effective manner with embedded ML.

Experiment 1

Using manually collected application versions from their official webpage and exploits from **exploit-db**, launched simulation environment and **PPO** agents to choose effective exploits from application versions.



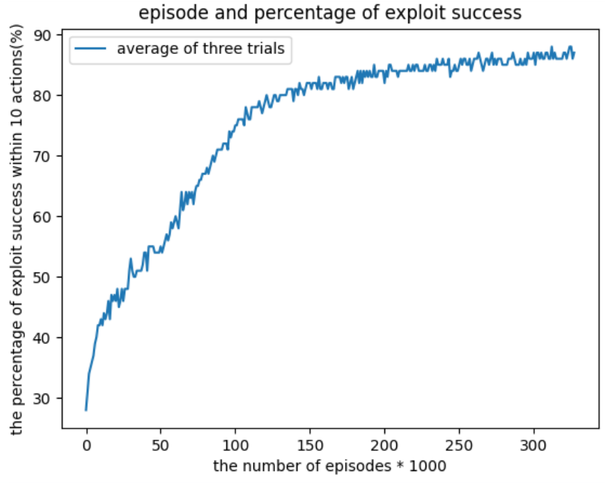
Goal of environment: success exploit actions or give up action within 10 chance of actions.

Actions: 44 type
15 kind of Scan action: Proving corresponding application versions and mapping to observation **reward:** 0(detected information)/ -1(failed)
28 kind of Exploit action:If target version is vulnerable, the episode ends with success.
reward: 1(the agent found exploit appropriate for the target version)/ -1(failed)
Give up action:declare "there is no vulnerability here" and end the episode.
reward:1(if there is really no vulnerable application): -2(left vulnerable application)

States:15 Application Versions (their range is mapped -1 to 1. (ex.4.8.1→[0.4,0.8,0.1]))

Result 1

The result averaged over three trials is shown below.



As a result, the percentage of success reached nearly 90%.
This shows the task can be handled by deep reinforcement learning.

Experiment2

connect the real-life applications and exploits to the learned model of Experiment 1 and demonstrate the applicability of our approach in more realistic environment.

Defined actions:
for wordpress,Jenkins, joomla,phpmyadmin.
Other actions
are defined only the shape, and they definitely fails.

Defined Action	Affected Versions
Wordpress scan	All
Wordpress exploit	5.0.0 and <=4.9.8
Jenkins scan	All
Jenkins exploit	1.60-1658
Joomla scan	All
Joomla exploit	3.0.0-3.4.6
phpmyadmin scan	All
phpmyadmin exploit	4.8.0-1

Environment 1:
When the exploits correct for this environment have wider range of target versions.

Applications	Versions	Vulnerable Action
Wordpress	5.8.1	Not Vulnerable
Jenkins	2.137	Not Vulnerable
Joomla	3.4.3	Joomla exploit
phpmyadmin	5.1.1	Not Vulnerable

Environment 2:
When the exploits correct for this environment have relatively narrower range of target versions.

Applications	Versions	Vulnerable Action
Wordpress	5.8.1	Not Vulnerable
Jenkins	2.137	Not Vulnerable
Joomla	4.0.3	Not Vulnerable
phpmyadmin	4.8.1	phpmyadmin exploit

Result 2

As a result of 30 demonstrations in each of the two states:

- **Environment 1**
Found valid exploits :26
Gave up :4
Could find correct exploits many times.

- **Environment 2**
Time up: 22
Gave up :8
Couldn't find correct exploits, and wasted time to useless exploits.

So the target range of exploit affects the performance of PT.

Future works

・exploits of RCE only in specific versions are not the only means of attack.
→Improvements based on such logic would make our proposed approach more realistic and easier to use.
・The agent should correctly determine that the environment is harmless if the exploit unexpectedly fails because of DBMS, OS ,and other causes
→ adjust the algorithm parameters so that the time series data can be learned correctly.

Reference

[1] https://github.com/13o-bbrbbq/machine learning security/tree/master/ DeepExploit, accessed on October 25, 2021.
[2] https://github.com/rapid7/metasploitframework/wiki, accessed on October 31, 2021.
[3] Mohamed C. Ghanem and Thomas M. Chen, "Reinforcement Learning for Intelligent Penetration Testing", 2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), 2018.
[4] Ovidiu Valea and Ciprian Opris, "Towards Pentesting Automation Using the Metasploit Framework", 2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP), 2020