Information & Security

P. de la Vallée, G. Iosifidis & W. Mees
vol. 53, no. 2 (2022): 273-286
https://doi.org/10.11610/isij.5318

Published since 1998

ISSN 0861-5160 (print), ISSN 1314-2119 (online)

Research Article

Cyber Red Teaming: Overview of Sly, an Orchestration Tool

Paloma de la Vallée [1] (✉), *Georgios Iosifidis,* [2] *Wim Mees* [1]

[1] *Cylab, Royal Military Academy, Brussels, Belgium*
http://www.cylab.be

[2] *RHEA Group, Brussels, Belgium*

ABSTRACT:

The complexity of protecting interconnected IT systems grows with the development of new products and applications. Consequently, the capability of Security Operation Centre personnel to keep abreast of new threats is of utmost importance to ensure the security posture of all organisations. In that regard, hands-on exercises on a cyber range reproducing realistic situations can boost the ability of personnel to react appropriately and adequately to intrusion in a production context. Such exercises are known to improve situation awareness. However, the design and delivery of such trainings impose a heavy workload on cyber experts. Relying on an automation system for the execution of attacks considerably lightens the duties of experts and frees some of their time for less repetitive tasks. This article introduces an orchestrator dedicated to red teaming.

ARTICLE INFO:

RECEIVED: 15 JUNE 2022
REVISED: 29 AUG 2022
ONLINE: 23 SEP 2022

KEYWORDS:

red teaming, automation, orchestration, Sly, cybersecurity training, exercises

Introduction

Thanks to the ongoing development of many new applications, tools and machines, our corporate and private information networks and systems continuously increase in complexity and interconnectivity. However, each new piece of

✉ Corresponding Author: E-mail: p.delavallee@cylab.be

software or device brings novel ways to use and exploit this abundance of intrusion possibilities. Unfortunately, the difficulty of monitoring their security raises accordingly: the defence of our organisation networks and assets requires unfailing alertness to detect malicious acts, while threat actors are constantly trying to evade the security teams monitoring and defensive actions.

Various organisations regularly analyse public information on breaches and successful attacks to evaluate trends in the threat actor behaviour.[1] [2] While the extent and motivation of intrusions vary considerably, it appears that cybercriminals consistently find new creative ways to exploit the ever-growing attack surface of the IT systems. On the other hand, well-known techniques are still being used profitably: on the whole, the cyberactivity increases steadily in volume and diversity.

In spite of cyber awareness campaigns deployed at various level, these reports also demonstrate how the humans are still considered to be the weak link in the security chain.[3] However, successful attack response and recovery is also the feat of humans, an essential factor of the solution. The Security Operation Centre (SOC) teams must manage different key tasks to protect Information Technology (IT) systems. A critical responsibility is the detection of suspicious activity or intrusions on the networks and assets. While this duty is considerably aided by Intrusion Detection Systems (IDSes) and other monitoring tools of increasing advanced capabilities, the actual response to attacks is still very much a human operation. The complexity of response and recovery actions make them ill-adapted to being delegated to some sort of automated response system. The capacity to select the most pertinent measures in face of such a complex dynamic context demands a very specific set of knowledge, skills, and abilities (KSA). Furthermore, the quality of the KSA also sustains situational awareness, that is decisive for an effective attack response.

This paper is structured as follows. The section 2 details the structure and importance of situational awareness. The section 3 discusses automation in the context of red teaming. The section 4 presents an overview of the orchestrator structure and the section 5 an instance of red teaming automation. The section 6 concludes with perspective for future work.

Situation Awareness

Situation awareness (SA) is a theoretical concept that abstracts the way humans perceive and act on their environment. It is a fundamental support of appropriate decision making. In the context of cyber security, SA is hence a crucial element sustaining the efficiency of responses to breaches attempts and intrusions. Endsley defined situation awareness as “the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future.” [4] This definition highlights three levels of situation awareness components: the perception, comprehension and projection.[5]

- *Perception* is the lowest level of awareness, it relates to the consciousness, or knowledge of relevant information about the environment.
- *Comprehension* is the ability to consolidate the perception in view of the circumstances, in light of the operational goals. It articulates the atomic pieces of information of the perception into a complete contextual understanding of the situation at hand.
- Finally, the *projection* is the aptitude to foresee events and system states in the near future based on the perception and comprehension. It takes advantage of mental models created in past experience or training to predict how the present situation might evolve.

The figure 1 shows the different cognitive elements articulated around SA, from the perception of the dynamic environment to decision making and actions.

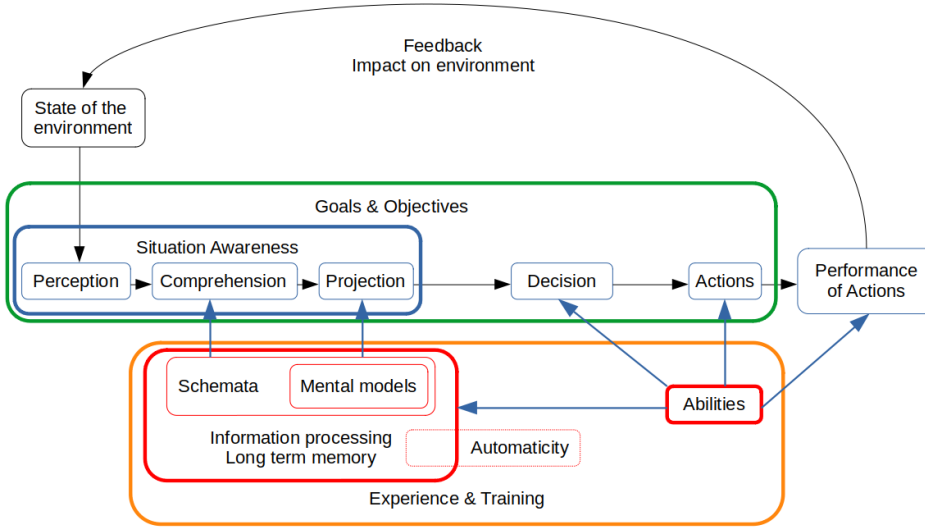

Figure 1: Overall structure of cognitive elements in the context of response to dynamic systems (Adapted from [5]).

It should be stressed that SA is an awareness of the environment at a certain point of time: dynamic systems evolve constantly. The goals and objectives pursued create a context for the situation awareness and help prioritise information pertaining to the environment state, as well as the progressive elaboration of the perception, comprehension and projection. The quality of the SA at all three levels is dependent on the cognitive abilities of the individual. In particular, the capacity to split one's attention towards different environmental cues can play a role in the overall perception of the environment situation details.

Information processing capabilities allow the integration of the information toward comprehension and finally projection. As a person acquires experience through training or exposition to various situation, be it either in real production setting or in the context of trainings, they develop schemata and mental models stored in long term memory. The information collected from the environment is actually mapped onto these schemata that allow to promptly structure the information. Hence, the schemata will lighten the mental workload necessary to convert the set of environmental cues into a coherent situation understanding. There is loss of information when the situation details are mapped one a schema, but the information is then structured in a coherent way that eases and quickens further processing.

The projection level necessitates to predict the near future; this processes is aided by mental models. They are a complex form of schemata characterising the system behavior, its state and its probable evolution. Both the comprehension and the projection level leverage the excellent human pattern-matching capabilities to classify the environmental elements of information on schema and mental models. The efficiency of this process relies on the individual ability to detect key environmental cues, and relate them to key features of the models. This ability is usually lacking in novice individuals: their progression toward experts reflects the integration of experience into these schemata and mental models, as well as the capability to exploit them adequately.

The existence of these structures favour a swift processing of the information and their development can in time lead to automaticity. Automatic processing allows to make decision rapidly and effortlessly, thereby relieving a sizeable part of the mental workload. Automaticity however holds a risk: individuals tend to overlook new environmental cues and might adopt less adequate schemata.

To actually make a decision, the individual considers the projection and the goals before he select the estimated most appropriate action. The execution of an action affects the environment as expected or not, and its performance also depends on the abilities of the individual. The environment state is changed due to the action taken, and the whole process of evaluation-decision-action must start again until the goals are considered achieved.

This discussion demonstrate that accurate situation awareness is a fundamental premise for efficient decision making. The quality of SA has been shown to degrade under stress, heavy workload or due to the complexity of the dynamic system. On the other hand, it can be enhanced with exercises and case studies that improve the schemata and mental models. In particular, the theoretical knowledge is better integrated with exercises: studies have shown that the practical dimension of lessons enhance the training efficiency.[6] In the context of cyber security, the training scenarios should be realistic, complex, varied, up- to-date and representative of production situations: cyber ranges offer a flexible and safe environment to design and deploy such exercises. This explains the considerable interest in cyber ranges, recent research bringing advances in various directions as numerous surveys and meta-analyses testify.[7, 8, 9, 10]

Automation in the Context of Red Teaming

The preceding section detailed how experiential training can enhance the situation awareness of cyber personnel by improving his cognitive abilities. The actions taken are consequently bettered, thereby boosting the security posture of the organisation. Exposing personnel to experiential training can also help address overall skill gap observed on the market.[11, 12] Indeed, the ENISA sees education as a pragmatic method to address the difficulty to find qualified personnel observed at the European level.[13] However, to significantly improve personnel KSA, the cyber training sessions must be as realistic as possible, both in terms of the network scenario involved and the attack deployed on the virtual infrastructure. This means that the network scenario and attack need to be complex and elaborate to bring added-value to the trainees. At present, the delivery of a hands-on training scenario encompassing an attack on a cyber range usually relies on a cyber expert to design all the lessons elements and for the actual execution of said attack. However, given the limited number of cyber experts that are available for such training duties, this situation does not scale very well. A system able to take over the role of the cyber expert by executing a predesigned attack independently on a network scenario addresses partially this scalability issue.

Fortunately, while attack simulations on cyber ranges usually require the execution of many tasks, the majority of them can easily be automated. Indeed, offensive activities involve a lot of scanning, configuration, modules exploitation, etc. These tasks actually leverage applications and tools that execute the bulk of the heavy lifting. The human intervention consists in extracting information of the outcome of these tasks, and transform this information in another format fit for ingestion by the next tool. While humans excel at this exercise, in particular expert threat actors, the implementation of an all-encompassing algorithm to this end is not straightforward.

This shows that, in the context of legit offensive actions automation, the low-hanging fruits have been collected. The remaining areas of progress are the highly complex tasks of:

- Eliciting information from non-standardized data.
- Processing this information for storage into a knowledge base.
- Analysis of the knowledge base to select an appropriate action with high probability of success.
- Leveraging the knowledge base to generate workable input in the context of the selected action.
- Advance toward a pre-defined overarching goal.

This has led to a lot of research and publications on penetration testing and red teaming tools that integrate different levels of automation.

Penetration testing

In essence, pentesting evaluates the level of exposition of an organisation's IT system. Whitin a pre-defined scope of operation, a team of cyber experts tries to breach defenses and gain unauthorized access to machines. The fundamental difference with the red teaming exercises considered in this paper is that pentesting happens in a production environment. The cyber team does not have a full knowledge of the network architecture and machine configurations. It is also essential that no lasting damage be caused on the target network: offensive activities cannot be taken unadvisedly. Some pentesting tools, such as CARTT [14] simply aim at easing the operator tasks by presenting the results in a structured way and suggesting next actions. Other applications attempt to generate attack plans.[15, 16, 17] However penetration testing suffers from the inherent difficulty of planning in the context of incomplete World View.[18] As a consequence, recent research focuses on higher forms of artificial intelligence, in particular reinforcement learning, to automate pentesting.[19, 20, 21, 22]

Cyber Red Teaming

When operating on a cyber-range, cyber experts designing a scenario enjoy much more freedom about their actions. They have a complete world view and can deteriorate virtual machines at will.

In general, research and training activities on cyber ranges are frequently the object of repetitions. For instance, developing an IDS will require to test its capacities against malicious traffic frequently: the ability to delegate the generation of such traffic to an automated system is a great advantage. The Lincoln Laboratory has developed such an automated system to generate red traffic, the Lincoln Laboratory Attack Framework, an element of their Cyber range advanced tools.[23] Similarly, APIT is a tool aimed specifically at generating malicious traffic to test IDSes.[24]

However, depending on the context, it might be necessary to deploy a specific, complex attack. In that case, the attack needs to be encoded in an appropriate way, to be consumed by an advanced automation system. The National Cyber Range (NCR) [25, 26] implements a very complex framework, with the necessary elements to encompass the whole experimental chain. SVED [27, 28] also has this attack design and execution capacity. Caldera is an open-source post-compromise tool developed by MITRE, with adversary emulation capabilities.[29]

Automated Red Teaming Use Cases

The ability to automate complex attacks is a great asset in the context of hands-on cyber range exercises. Other use cases can leverage such an automation profitably. IDS development and testing has made use of attack automation for some time. The IDSes capability to detect attacks is evaluated at a lower overall cost.

Furthermore, if no other actions are taken on the cyber range at the time of the attack progress, automation allows to produce exact, known forensic traces

in the scenario. The attack can be repeated exactly, in a fixed time frame, with predicted outcomes. The specific characteristic to be precisely reproducible in a considerable advantage of automation. In particular, in an examination setting, all students can be submitted to the same fair evaluation since they are exposed to the same pre-designed situation. Similarly, various IT products security, or human skills, can be evaluated in a certification setting. The automated attack could be used to confirm that products or assets can withstand certain breach attempts. A comparable procedure could be used to certifiy the skills and abilities of personnel.

This section demonstrates that automation solutions have been developed successfully in different contexts. The automation of red teaming systems is less well explored than the pentesting, the later being much more frequent commercially. Complex designs usually necessitate on human supervision at some level. The latest systems support increasingly complex relationships between tasks

Orchestrator architecture

This section presents an overview of the architecture of the Sly red teaming automation system. Different elements compose this system: a Central Management system, a set of databases, a collection of workers, a communication system and an execution capability on the Attacker.

In practice, an orchestrator manages the different actions needed to carry out the attack in a timely manner. It runs on a specific virtual machine (VM) deployed in the network scenario where the attack will be effectuated. Where appropriate, it communicates instructions to attacker machines, typically Kali VMs.

Representation and Execution of the Attack

The attack is abstracted as an acyclic oriented and conditional graph of nodes, each of which is a task. Child-parents relationships bind the different nodes, and the children are launched depending on a required status of the parents. At regular intervals, the orchestrator controls whether the tasks failed, succeeded, or have simply been evaluated before deciding to launch a child.

All nodes, or tasks, are python functions that are carried out by Celery workers running on the Orchestrator VM. Celery is an asynchronous task queue system based on distributed message passing. By default, Celery starts as many workers as there are CPUs on the VM: parallel processing of tasks is hence easily handled. Redis serves as messaging substrate and Tasks database. It keeps track of the tasks IDs and links these IDs to the tasks status and their outcome after completion. Therefore, the output of a task remains accessible at all times to evaluate its success.

Execution on Attacker virtual machines

The orchestrator is framework-agnostic. As mentioned earlier, the core of the orchestrator work is carried out on its VM. However, where relevant, the orchestrator can also induce execution of commands on external VMs. All command line instructions can be completed on external machines over appropriate connections. This opens a wealth of possibilities such as launching a Virtual Private Network (VPN) connection, adapting configuration files, changing the routing tables, installing new software, etc.

These commands are launched by interaction with a terminal. For simple tasks such as a reboot, the command is sent over a transient ssh connection. If, however, a terminal must remain open in a lasting manner, a paramiko channel is opened on the involved Linux-based VM.[30] The management of the communication between the Orchestrator VM and the Linux-based VM is handled by RabbitMQ, the instructions for specific terminals being segregated into separate Rabbit queues. This system provides the possibility to maintain multiple stable terminals running in parallel on the same machine.

Central Management of Tasks

Sly is a python program that manages the overall execution of the attack. At present, it can only handle a pre-defined attack graph encoded in a YAML file. Regularly, the Redis Task database is polled to evaluate the status of the tasks, and their outcome is parsed to assess their success. Sly stores all tasks status internally; it exploits this internal repository to evaluate if the conditions to launch children tasks are met. The actual execution of a task is carried out by submitting it to Celery, which will assign the task to an available worker.

The orchestrator also maintains a relational database to keep track of a World View at the network scenario level. It is a abstraction of the information gained during the progress of the attack, related to the target network. Typically, these will pertain to user credentials, services running on machines, IP addresses, etc.

Resilience of the Architecture

The architecture was designed with resilience in mind. To resist to loss of connection, the communication between the different elements is handled in a connectionless manner. The Redis database and the World view allow to resist to a crash in the Central Management system Sly. In such a case, the Central Management system can resume the attack from an appropriate intermediate graph node. Sly will first load the tasks status and the World View, possibly updating data where necessary, before proceeding with the graph execution. Indeed, the real attack is in progress on the Attacker VM and hence impervious to mishaps in the orchestration. For instance, should a meterpreter session be opened be- tween the Attacker VM and a Target VM, this session is insensitive to a crash of Sly and remains available after a re-launch.

Instance of an Attack Scenario

The Sly orchestrator has already been successfully used to automate different attacks. One such simple instance is presented here.

The figure 2 shows a network scenario used to demonstrate the capabilities of Sly. It is deployed on an OpenNebula cyber range.[31]

The red team is represented on the left, and consists of a Kali VM, or Attacker, and the Orchestrator VM.

The target network is represented on the right. It consists of a single Local Area Network (LAN) comprising three virtual machines. The target LAN is a severely scaled down version of a passenger ship's internal network, reduced to two VMs essential for the structure of the attack. It contains a Linux server containing files of interest for passengers, the Primary Target. The network also includes the Secondary Target, a Windows 7 workstation that runs an Electronic Chart Display and Information System (ECDIS) console used by the ship personnel. In this scenario, the console acquires the navigation data from a specific program that generates realistic data locally for simulation purposes. The target LAN is protected by a firewall that only allows outgoing traffic. The firewall accepts incoming connections for the Linux server on port 80. The attacking and target sides can connect via a network mimicking the Internet. Due to the firewall configuration, the red team can only directly scan or access the Linux server port 80.

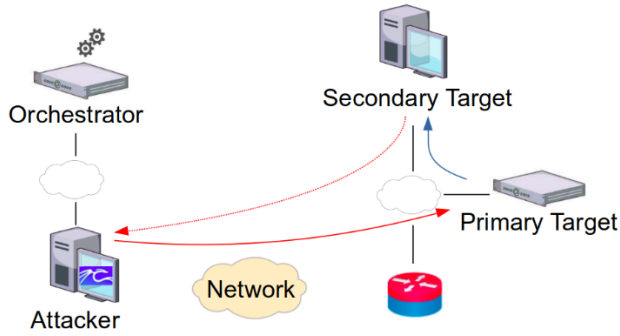

Figure 2: Instance of a network topology and path of the attack.

The figure 3 represents the attack as an oriented graph of tasks, in simplified form. The graph nodes, corresponding to tasks, are organised in columns. The three columns correspond to separate terminals on which the tasks are executed.

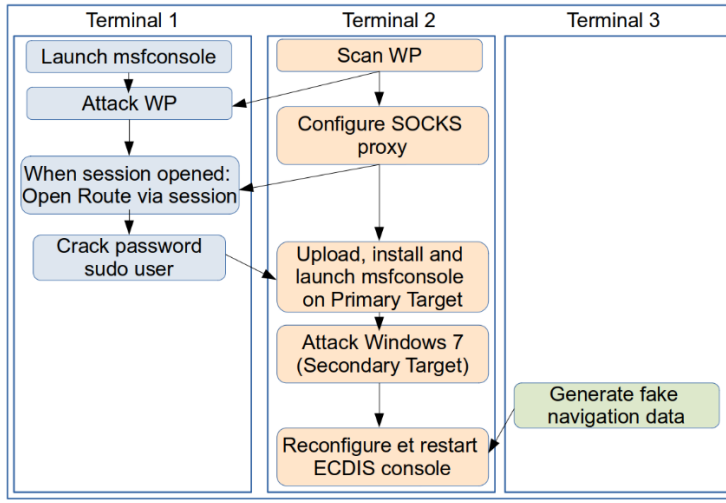

Figure 3: Organisation of the attack in an acyclic oriented graph of tasks.

On the Kali virtual machine, the msfconsole is launched on the terminal 1. On a second terminal of the same VM, the attacker scans the port 80 of the Linux server. The scan discovers a vulnerability sensitive to a metasploit module. Within the msfconsole on the terminal 1, the appropriate module is launched successfully and a meterpreter session is opened between the Attacker and the Primary Target. The second terminal is used to configure a SOCKS proxy, that is then leveraged to route all traffic toward the target network via the newly opened session.

A quick exploration of the Primary Target returns a list of users, one of which is a sudoer. A brute-force password attack is successfully launched against this user. The acquired credentials of the sudoer are used to open a SSH connection on the Primary Target, through which the metasploit installation package is uploaded and run. This allows to launch the msfconsole on the Target and explore its LAN. A scan of the Target network detects the presence of a Windows 7 desktop, the Secondary Target. A specific metasploit module is successfully run against this Operating System, creating a meterpreter session between the Primary Target and the Secondary Target, session which is accessed over the SSH connection mentioned above. Through this session, the presence of an ECDIS system is detected. The Attacker starts to generate fictious navigation data locally. The configuration of the ECDIS console is then corrupted to take its data feed from the Attacker. The Attacker now manipulates the navigation data displayed on the ECDIS console, resulting on a major issue for the ship personnel.

This attack demonstrates the ability of the orchestrator to manage complex tasks inter-dependencies in a timely manner. It also proves the integration of the orchestrator with the Metasploit framework, as well as the possible execution of very varied instructions on Linux-based machines (Kali, Primary Target).

The orchestrator was also deployed as an automated Attacker in the hands-on maritime training [i] delivered on federated cyber ranges in the context of the ECHO Federated Autumn School.[32]

Conclusion and Future Work

This paper presents the added value of an automation system in the context of red teaming engagements deployed in trainings, examinations or certifications. The technical architecture of such an orchestrator is outlined; it has the ability to execute an attack abstracted into a conditional acyclic oriented graph of tasks. At present, the typical use case of the orchestrator is to manage the automation of a pre-defined attack to be carried out on a scenario deployed on a cyber range. The design of the attack is performed by a cyber expert, responsible to develop the network scenario and the associated suite of actions that constitutes the attack.

The exercises managed by the orchestrator could be improved by the integration of additional external tools or frameworks. A common issue in trainings is the production of background traffic to hide the traces of the attack. This issue is usually addressed by having the VMs in the scenario generate standard traffic; typically by installing an application with the ability to mimic normal end-users. In that context, integrating such tools would allow the orchestrator to also handle the generation of legit, background traffic.

In its present state, the orchestrator does not have the ability to make decision as to which action to take next. This limitation could be overcome by other, accompanying systems. For instance, an attack planner could generate the attack graph to execute to reach a certain goal.[33] On the other hand, the orchestrator could be linked to an artificial intelligence that would be in charge of making decisions online, based of the World View being progressively acquired during the attack.

This discussion demonstrates that the orchestrator described in this paper can be at the core of many future developments to enhance the complexity and realism of the attack deployed in trainings.

Acknowledgements

This work was supported by the ECHO project which has received funding from the European Union's Horizon 2020 research and innovation program under the grant agreement no 830943.

[i] This federated maritime scenario is described in a paper entitled 'Sector-specific training: a federated maritime scenario', submitted for publiction in 2022.

References

1 NTT, "2021 Global Threat Intelligence Report," Technical report, 2021, https://services.global.ntt/en-gb/insights/2021-global-threat-intelligence-report.

2 ENISA, "ENISA Threat Landscape 2021," 2021, Technical report, https://www.enisa.europa.eu/ publications/enisa-threat-landscape-2021.

3 World Economic Forum, "After reading, writing and arithmetic, the 4th'r' of literacy is cyber-risk," 2020, https://www.weforum.org/agenda/2020/12/cyber-risk-cyber-security-education.

4 Mica R. Endsley, "Situation awareness global assessment technique (SAGAT)," *Proceedings of the IEEE 1988 National Aerospace and Electronics Conference, Dayton, OH, USA,* 1988, pp. 789-795.

5 Mica R. Endsley, "Toward a Theory of Situation Awareness in Dynamic Systems," *Human Factors* 37, no. 1 (1995): 32-64.

6 Kirsi Aaltola and Petteri Taitto, "Utilising Experiential and Organizational Learning Theories to Improve Human Performance in Cyber Training," *Information &Security* 43, no. 2 (2019): 123-133.

7 Jon Davis and Shane Magrath, "A survey of cyber ranges and testbeds," Technical Report, ADA594524, Defence Science and Technology, Cyber and Electronic Warfare Division, Australia, 2013.

8 Omar Darwish, Christopher M. Stone, Ola Karajeh, and Belal Alsinglawi, "Survey of Educational Cyber Ranges," *WAINA 2020: Web, Artificial Intelligence and Network Applications*, 2020, pp. 1037-1045.

9 Muhammad Mudassar Yamin, Basel Katt, and Vasileios Gkioulos, "Cyber ranges and security testbeds: Scenarios, functions, tools and architecture," *Computers & Security* 88 (2020), Article ID 101636.

10 Nestoras Chouliaras, George Kittes, Ioanna Kantzavelou, Leandros Maglaras, Grammati Pantziou, and Mohamed Amine Ferrag, "Cyber Ranges and TestBeds for Education, Training, and Research," *Applied Sciences* 11, no. 4 (2021): 1809.

11 ISSA, "The Life and Times of Cybersecurity Professionals 2021," 2021, https://www.issa.org/cybersecurity-skills-crisis-continues-for-fifth-year-p erpetuated-by-lack-of-business-investment/.

12 Fortinet, "2022 Cybersecurity Skills Gap," 2022, https://www.fortinet.com/content/dam/fortinet/assets/reports/report-2022-skills-gap-survey.pdf?utm_source=pr\&utm_campaign=report-2022-skills-gap-survey.

13 ENISA, "Addressing skills shortage and gap through higher education," 2021, https://www.enisa.europa.eu/publications/addressing-skills-shortage-and-gap-through-higher-education/@@download/fullReport.

14 Joseph Plot, Alan Shaffer, and Gurminder Singh, "CARTT: Cyber Automated Red Team Tool," *Proceedings of the 53rd Hawaii International Conference on Systems Sciences*, 2020, pp. 6695-6704.

[15] Dominik Elsbroek, Daniel Kohlsdorf, Dominik Menke, and Lars Meyer, "Fidius: Intelligent support for vulnerability testing," In: *Working Notes for the 2011 IJCAI Workshop on Intelligent Security (SecArt)*, 2011, pp. 58-65.

[16] Jorge Lucangeli Obes, Carlos Sarraute, and Gerardo Richarte, "Attack Planning in the Real World," *SecArt'2010 at AAAI 2010, Atlanta, USA. July 12*, 2010.

[17] Carlos Sarraute, Olivier Buffet, and Joerg Hoffmann, "POMDPs Make Better Hackers: Accounting for Uncertainty in Penetration Testing," *Proceedings of AAAI Conference on Artificial Intelligence*, 2012, pp. 1816-1824.

[18] Joerg Hoffmann, "Simulated Penetration Testing: From 'Dijkstra' to 'Turing Test++'," *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 25, no. 1, 2015.

[19] Dean Richard McKinnel, Tooska Dargahi, Ali Dehghantanha, and Kim-Kwang Raymond Choo, "A systematic literature review and meta-analysis on artificial intelligence in penetration testing and vulnerability assessment," *Computers and Electrical Engineering* 75 (2019): 175–188.

[20] Jonathon Schwartz, "Autonomous Penetration Testing using reinforcement learning," Thesis, Bachelor of Science, School of Information Technology and Electrical Engineering, University of Queensland, Australia, 2018.

[21] Mohamed Ghanem and Thomas Chen, "Reinforcement learning for efficient network penetration testing," *Information* 11, no. 1 (2020).

[22] Sujita Chaudhary, Austin O'Brien, and Shengjie Xu, "Automated Post-Breach Penetration Testing through Reinforcement Learning," *2020 IEEE Conference on Communications and Network Security (CNS), Avignon, France*, 2020, pp. 1-2.

[23] Timothy M. Braje, "Advanced Tools for Cyber Ranges," MIT technical report, 2016, https://www.ll.mit.edu/doc/advanced-tools-cyber-ranges.

[24] Anas Kalam, Mohammed Gadelrab, and Yves Deswarte, "A model-driven approach for experimental evaluation of intrusion detection systems," *Security and C Yves Deswarteommunication Networks* 7, no. 11 (2014): 1955-1973.

[25] Peter Haglich, Robert Grimshaw, Steven Wilder, Marian Nodine, and Bryan Lyles, "Cyber Scientific Test Language," *ISWC 2011: The Semantic Web – ISWC 2011*, 2011, pp. 97-111.

[26] Bernard Ferguson, Anne Tall, and Denise Olsen, "National Cyber Range Overview," *IEEE Military Communications Conference, Baltimore, MD, USA*, 2014, pp. 123-128.

[27] Hannes Holm and Teodor Sommestad, "SVED: Scanning, Vulnerabilities, Exploits and Detection," *MILCOM 2016 - 2016 IEEE Military Communications Conference, Baltimore, MD*, 2016, pp. 976-981.

[28] Niklas Hätty, "Representing attacks in a cyber range," Master thesis, Linköping University, Department of Computer and Information Science, Software and Systems, 2019.

[29] Caldera, 2022, https://github.com/mitre/caldera.

[30] Paramiko, 2022, https://www.paramiko.org/.

[31] OpenNebula, 2022, https://opennebula.io/.

[32] ECHO federated Autumn School, 2022, https://echonetwork.eu/echo-federated-autumnschool/.

[33] Joseph Yuen, "Automated Cyber Red Teaming," Technical Report, Australian Government, Department of Defence, 2015, https://apps.dtic.mil/sti/citations/ADA618584.

About the Authors

Paloma de la Vallée Poussin is a researcher in the Cyber Defence Laboratory, https://www.cylab.be at the Royal Military Academy in Brussels, Belgium. https://orcid.org/0000-0002-0407-1071

Georgios Iosifidis is an engineer with the RHEA Group, Brussels, Belgium, specialising in the field of cyber ranges.

Dr. **Wim Mees** is a Profesor at the Royal Military Academy, Brussels, Belgium and leads its Cyber Defence Laboratory, https://www.cylab.be. His current research interests cover threat intelligence, intrusion detection, cyber range based training and simulation, certification, and cyber situation awareness. https://orcid.org/0000-0002-0696-8093