KYPO Cyber Range: Design and Use Cases

Jan Vykopal[1], Radek Ošlejšek[2], Pavel Čeleda[1], Martin Vizváry[1], Daniel Tovarňák[1]
[1]*Institute of Computer Science, Masaryk University, Brno, Czech Republic*
[2]*Faculty of Informatics, Masaryk University, Brno, Czech Republic*
{vykopal|celeda|vizvary|tovarnak}@ics.muni.cz, oslejsek@fi.muni.cz

Keywords: KYPO, cyber range, cyber attack, system design, cloud computing, network virtualization

Abstract: The physical and cyber worlds are increasingly intertwined and exposed to cyber attacks. The KYPO cyber range provides complex cyber systems and networks in a virtualized, fully controlled and monitored environment. Time-efficient and cost-effective deployment is feasible using cloud resources instead of a dedicated hardware infrastructure. This paper describes the design decisions made during it's development. We prepared a set of use cases to evaluate the proposed design decisions and to demonstrate the key features of the KYPO cyber range. It was especially cyber training sessions and exercises with hundreds of participants which provided invaluable feedback for KYPO platform development.

1 Introduction

Operational cyber environments are not suitable for building a systematic knowledge of new cyber threats and to train responses to them. Therefore, cyber ranges or testbeds are usually built to provide a realistic environment suitable for training security and operations teams. A cyber range provides a place to practice correct and timely responses to cyber attacks. The learners can practice skills such as network defence, attack detection and mitigation, penetration testing, and many others in a realistic environment.

Despite the increasing popularity of cyber exercises (Welch et al., 2002; NATO CCDCOE, 2017), there is very limited public information about platforms used. Due to the specific use of cyber ranges (government, military, industry), many technical details are regarded as sensitive. This paper shall provide an integrated view of the KYPO cyber range (KYPO, 2017), which has been in development since 2013. KYPO was made for researching and developing new security methods, tools and for training security teams and students. It provides a virtualised environment for performing complex cyber attacks against simulated cyber environments.

Apart from the technical aspects, the transdisciplinary features of cyber exercises are equally important. Preparing and carrying out cyber exercise requires substantial time, effort and financial investments (Childers et al., 2010). The major workload is carried out by the organizers, particularly in the exercise preparation phase. The ultimate goal of a cyber range developers is to minimize this workload and to support all phases of an exercise's life cycle. We have designed and executed a cyber defence exercise to validate the KYPO cyber range prototype. The technical part of the exercise relies on the built-in capabilities of KYPO and was used in six runs of a cyber defence exercise for 50 participants. Several lessons were learned which provided important guidance for further KYPO research and development.

This paper is divided into six sections. Section 2 shall provide background information about testbeds and cyber ranges. Section 3 will describe KYPO's architecture design and list the main components of the proposed architecture. Section 4 shall describe the user interface and interactions in the KYPO cyber range. Section 5 will show three selected use cases. Finally, Section 6 will conclude the paper and outline future work on KYPO.

2 Related Work

In this section, we introduce generic testbeds which can be used in cyber security. Then we focus on environments which have been specially developed for cyber security training. While some of these evolved from generic testbeds, others were designed with cyber security in mind. The environments are costly, but versatile large-scale infrastructures with state of the art parameters and features as well as

lightweight alternatives with limited scope, functionality and resources.

The Australian Department of Defence published an extensive survey of state of the art cyber ranges and testbeds (Davis and Magrath, 2013). The survey lists more than 30 platforms which can be used for cyber security education worldwide. This number is based on publicly available, non-classified information. Since the development and operation of some cyber ranges is funded by the military and governments of various countries, there is likely to be other classified cyber ranges. To cover recent advances and innovations, we have done a systematic literature review from 2013 to 2017.

2.1 Generic Testbeds

Emulab/Netbed (White et al., 2002) – this is a cluster testbed providing basic functionality for deploying virtual appliances, configuring flexible network topologies and the emulation of various network characteristics. The network topology must be described in detail by an extension of NS language. Emulab allocates computing resources for the specified network and instantiates it in a dedicated HW infrastructure.

Emulab has been developed since 2000 and there are currently about 30 of its instances or derivates in use or under construction worldwide (Emulab, 2017). It can be considered to be a prototype of an emulation testbed for research into networking and distributed systems. It provides accurate repeatable results in experiments with moderate network load (Siaterlis et al., 2013).

CyberVAN (Cyber Virtual Ad hoc Network, 2017) – this is a cyber experimentation testbed funded by the U.S. Army Research Laboratory and developed by Vencore Labs. CyberVAN enables arbitrary applications to run on Xen-based virtual machines that can be interconnected by arbitrary networks topologies. It employs network simulators such as OPNET, QualNet, ns-2, or ns-3, so the network traffic of emulated hosts travels through the simulated network. As a result, this hybrid emulation enables the simulation of large strategic networks approximating a large ISP network.

2.2 Cyber Ranges

DETER/DeterLab (Mirkovic et al., 2010) – the DETER project was started in 2004 with the goal of advancing cyber security research and education. It is based on Emulab software and has developed new capabilities, namely *i)* an integrated experiment management and control environment SEER (Schwab et al., 2007) with a set of traffic generators and monitoring tools, *ii)* the ability to run a small set of risky experiments in a tightly controlled environment that maximizes research utility and minimizes risk (Wroclawski et al., 2008), and *iii)* the ability to run large-scale experiments through a federation (Faber and Wroclawski, 2009) with other testbeds that run Emulab software, and with facilities that utilize other classes of control software. Lessons learned through the first eight years of operating DETER and an outline of futher work are summarized in (Benzel, 2011).

DETER operates DeterLab which is an open facility funded by U.S. sponsors and hosted by the University of Southern California and University of California, Berkeley. It provides hundreds of general-purpose computers and several specialized hosts (e. g., FPGA-based reconfigurable hardware elements) interconnected by a dynamically reconfigurable network. The testbed can be accessed from any machine that runs a web browser and has an SSH client. Experimental nodes are accessed through a single portal node via SSH. Under normal circumstances, no traffic is allowed to leave or enter an experiment except via this SSH tunnel.

National Cyber Range (NCR) (NCR, 2017) – the NCR is a military facility to emulate military and adversary networks for the purposes of realistic cyberspace security testing, supporting training and mission rehearsal exercises (Ferguson et al., 2014). Its development and operation have been funded by the U.S. Department of Defense since 2009 and the target user group are U.S. governmental organizations. The NCR enables operational networks to be represented, and interconnected with military command and control systems, with the ability to restore to a known checkpoint baseline to repeat the test with different variables. The NCR is instrumented with traffic generators and sensors collecting network traffic and data from local and distributed nodes. The NCR has demonstrated the ability to rapidly configure a variety of complex network topologies and scale up to 40,000 nodes including high-fidelity realistic representations of public Internet infrastructure.

Michigan Cyber Range (MCR) (MCR, 2017) – this is an unclassified private cloud operated by Merit, a non-profit organization governed by Michigan's public universities in the USA. The MCR has offered several services in cyber security education, testing and research since 2012.

The MCR Secure Sandbox simulates a real-world networked environment with virtual machines that act as web servers, mail servers, and other types of hosts. Users can add preconfigured virtual machines or build their own virtual machines. Access to the Sandbox is

provided through a web browser or VMware client from any location.

Alphaville is MCR's virtual training environment specifically designed to test teams' cyber security skills. Alphaville consists of information systems and networks that are found in a typical information ecosystem. Learners can develop and exercise their skills in various hands-on formats such as defence and offense exercises.

SimSpace Cyber Range (Lee Rossey, 2015) – a U.S. private company runs this cyber range, which enables the realistic presentation of networks, infrastructure, tools and threats. It is offered as a service hosted in public clouds (Amazon Web Services or Google), at the SimSpace datacenter, or deployed in the customer's infrastructure and premises.

The cyber range provides several types of preconfigured networks containing from 15 to 280 hosts which emulate various environments (generic, military, financial). It is possible to generate traffic emulating enterprise users with host-based agents and run attack scenarios automatically by combining various attacker tasks. All activities can be also monitored at network and scenario level (network traffic, attackers' and defenders' actions, and activities of emulated users at end hosts). The platform is controlled via a web portal that also provides access to the results of an analysis and assessment of monitored activities within the cyber range.

EDURange (EDURange, 2017) – this is a cloud-based framework for designing and instantiating interactive cyber security exercises funded by the U.S. National Science Foundation and developed by Evergreen State College, Olympia, Washington. EDURange is intended for teaching ethical hacking and cyber security analysis skills to undergraduate students. It is an open-source software with a web frontend based on Ruby and backend deploying virtual machines and networks hosted at Amazon Web Services. The exercises are defined by a YAML-based *Scenario Description Language* and can be instantiated by the instructor for a selected group of students. EDURange supports Linux machines which can be accessed via SSH. It also has built-in analytics for host-based actions, namely a history of commands executed by students during the exercise.

2.3 Lightweight Platforms

Avatao (Buttyán et al., 2016; Avatao, 2017) – this is an e-learning platform offering IT security challenges which are created by an open community of security experts and universities. Avatao is developed by an eponymous spin-off company of CrySyS Lab at Budapest University of Technology and Economics, Hungary. It is a cloud-based platform using lightweight containers (such as Docker) instead of a full virtualization. This enables it to start a new challenge in its virtual environment very quickly in comparison with booting full-fledged emulated hosts. Learners and teachers access the challenges via web browser. Hosts and services within the virtual environment are accessed by common network tools and protocols such as Telnet or SSH.

CTF365 (CTF365, 2017) – this is a Romanian commercial security training platform with a focus on security professionals, system administrators and web developers. It is an IaaS where users (organized in teams) can build their own hosts and mimic the real Internet. CTF365 provides a web interface for team management, instantiating virtual machines using predefined images and providing credential to access the machines using VPN and SSH. Each team has to defend and attack the virtual infrastructure at the same time. As a defender, a team has to set up a host which runs common Internet services such as mail, web, DB in 24/7 mode. As an attacker, the team has to discover their competitor's vulnerabilities and submit them to the scoring system of the CTF365 portal.

Hacking-Lab (Security Competence, 2017) – this is an online platform for security training and competitions run by a Swiss private company. It provides more than 300 security challenges and has about 40,000 users. The platform consists of a web portal and a network with vulnerable servers emulated using virtual machines or Docker containers. Each team administers a set of vulnerable applications and has to perform several tasks simultaneously, namely attack the applications of their competitors, keep their own applications secure, and up and running, find and patch vulnerabilities, keep applications up and running, and solve challenges. A Linux-based live CD is provided to ease the use of Hacking-Lab. It contains many hacking tools and is preconfigured for VPN access.

iCTF and InCTF iCTF framework (Vigna et al., 2014) was developed by the University of California, Santa Barbara for hosting their iCTF, the largest capture the flag competition in the world since 2002. The goal of this open-source framework is to provide customizable competitions. The framework creates several virtual machines running vulnerable programs that are accessible over the network. The players' task is to keep these programs functional at all times and patch them so other teams cannot take advantage of the incorporated vulnerabilities. The availability and functionality of these services is constantly tested by a

scorebot. Each service contains *a flag*, a unique string that the competing teams have to steal so that they can demonstrate the successful exploitation of a service. This flag is also updated from time to time by the scorebot.

InCTF (Raj et al., 2016) is a modification of iCTF that uses Docker containers instead of virtual machines. This enhances the overall game experience and simplifies the organization of attack-defence competitions for a larger number of participants. However, it is not possible to monitor network traffic, capture exploits and reverse engineer them to identify new vulnerabilities used in the competition.

3 KYPO Architecture Design

The KYPO cyber range is designed as a modular distributed system. In order to achieve high flexibility, scalability, and cost-effectiveness, the KYPO platform utilizes a cloud environment. Massive virtualization allows us to repeatedly create fully operational virtualized networks with full-fledged operating systems and network devices that closely mimic real world systems. Thanks to its modular architecture, the KYPO is able to run on various cloud computing platforms, e. g., OpenNebula, or OpenStack.

A lot of development effort has been dedicated to user interactions within KYPO since it is planned to be offered as Platform as a Service. It is accessed through web browser in every phase of the life cycle of a virtualized network: from the preparation and configuration artifacts to the resulting deployment, instantiation and operation. It allows the users to stay focused on the desired task whilst not being distracted with effort related to the infrastructure, virtualization, networking, measurement and other important parts of cyber research and cyber exercise activities.

3.1 Platform Requirements

At the beginning of the development of the KYPO platform, many functional and non-functional requirements were defined both by the development team and the project's stakeholders. The requirements were first prioritized using the MSCW method (***M**ust have, **S**hould have, **C**ould have, and **W**ould like, but will not have*). After the prioritization process, we identified the *must have* requirements that were the most likely to influence the high-level architecture of the KYPO platform as a whole. The following selected requirements have strongly influenced our high-level design choices.

Flexibility – the platform should support the instantiation of arbitrary network topologies, ranging from single node networks to multiple connected networks. For the topology nodes, a wide range of operating systems should be supported (including arbitrary software packages). The creation and configuration of such topologies should be as dynamic as possible.

Scalability – the platform should scale well in terms of the number of topology nodes, processing power and other available resources of the individual nodes, network size and bandwidth, the number of sandboxes (isolated virtualized computer networks), and the number of users.

Isolation vs. Interoperability – if required, different topologies and platform users should be isolated from the outside world and each other. On the other hand, integration with (or connection to) external systems should be achieved with reasonable effort.

Cost-Effectiveness – the platform should support deployment on commercial off-the-shelf hardware without the need for a dedicated data center. The operational and maintenance costs should be kept as low as possible.

Built-In Monitoring – the platform should natively provide both real-time and post-mortem access to detailed monitoring data. These data should be related to individual topologies, including flow data and captured packets from the network links, as well as node metrics and logs.

Easy Access – users with a wide range of experience should be able to use the platform. For less experienced users, web-based access to its core functions should be available, e. g., a web-based terminal. Expert users, on the other hand, should be able to interact with the platform via advanced means, e. g., using remote SSH access.

Service-Based Access – since the development effort and maintenance costs of a similar platform are non-trivial for a typical security team or a group of professionals, our goal is to provide transparent access to the platform in the form of a service.

Open Source – the platform should reuse suitable open source projects (if possible) and its release artifacts should be distributed under open source licenses.

3.2 High-Level Architecture

It can be seen that many of the requirements were already created with a cloud computing model in mind. This naturally influenced the KYPO platform high level architecture (Figure 1). The platform is composed of five main components – *infrastructure management driver*, *sandbox management*, *sandbox data*

store, *monitoring management*, and the *platform management portal* serving as the main user interaction point. These components interact together in order to build and manage *sandboxes* residing in the underlying cloud *computing infrastructure*. In the following paragraphs, we will individually describe each component. Since the user interface (platform management portal) is very complex it is thoroughly described in Section 4.

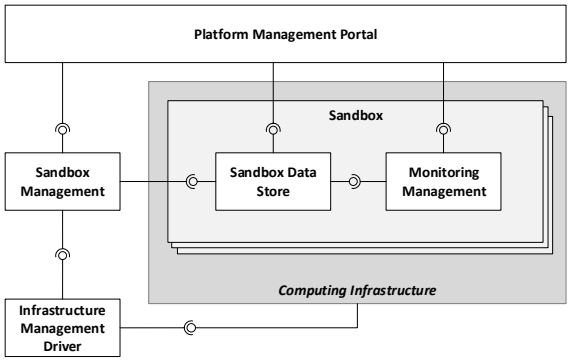

Figure 1: KYPO platform high-level architecture overview.

3.2.1 Infrastructure Management Driver

The infrastructure management driver is used to control the computing infrastructure. A computing infrastructure consists, in general, of housing facilities, physical machines, network devices, and other hardware and related configuration artifacts. It forms the raw computing resources such as storage, operating memory, and processing power. KYPO is designed to run on public cloud computing infrastructure so that *sandboxes* can be built without the need of dedicated infrastructure.

The infrastructure management driver is the only component of the architecture which directly access the low level computing infrastructure. Therefore, the support of multiple cloud providers is isolated to this single component. API provided by the driver offers services which enable the management of virtual machines and networks in a unified way. At present, the KYPO runs on OpenNebula cloud and the adaptation to OpenStack is under development.

3.2.2 Sandbox Management Component

The sandbox management component is used to create and control sandboxes in the underlying computing infrastructure. During the deployment of a sandbox, it orchestrates the infrastructure via infrastructure management driver in order to configure virtual machines and networking.

Advanced networking is one of the most important features of the KYPO platform. KYPO uses cloud networking as an overlay infrastructure. The underlying cloud infrastructure uses IEEE 802.1Q, i.e. Virtual LAN tagging, using Q-in-Q tunneling. Q-in-Q tunneling allows KYPO to configure *sandboxes* networking dynamically. It also does not depend on the L2 and L3 network addressing of the infrastructure, using a separate networking configuration. The *sandbox* networking allows users to configure their own L2/L3 addressing scheme in each LAN.

The networking in the *sandboxes* is done using one or more Lan Management Nodes (LMN). Each LAN network is managed by one LMN. LMN is a standard Debian system with an Open vSwitch (OvS) multilayer virtual switch (Linux Foundation, 2017). It combines standard Linux routing and OvS packet switching. The intra-LAN communication is done on the L2 layer using OvS as a *learning switch*. The inter-LAN communication is forwarded from switch to standard Linux routing tables.

The notion of KYPO points is used to connect external devices, systems and networks to the KYPO environment. Since the KYPO platform is cloud-based, there is a need for the mechanism to be able to connect systems and devices that do not have a virtualized operating system, i. e. they are hardware-dependent, or location dependent.

We have developed a device which connects such systems – based on a Raspberry Pi platform which automatically connects after its boot via Virtual Private Network (VPN) tunnel to the sandbox in KYPO. This makes the point very easy to use since it has very small proportions and it can be easily delivered and connected anywhere. The connection is secured via the properties of the VPN.

3.2.3 Sandbox Data Store

The sandbox data store manages information related to the topology of a sandbox and provides its generic abstraction. Since the KYPO is partially an overlay environment, it is necessary to bridge the configuration of nodes in the cloud infrastructure and the inner configuration of virtual machines.

Therefore, modules working with sandbox-related data, e. g., the platform management portal or the monitoring management component, do not retrieve information directly from the cloud but utilize the sandbox data store instead.

The store contains information about end nodes, IP addresses, networks, routes, and network properties during the whole lifetime of the sandbox. They are updated by the sandbox management component whenever changes to the sandbox are made. For ex-

ample, when a user deploys a new node or deletes a current node.

3.2.4 Monitoring Management Component

The monitoring management component provides fine-grained control over the configuration of the built-in monitoring and also provides an API that exposes the acquired monitoring data to external consumers (e. g., platform management portal). All the necessary information about the sandbox's topology is read from the sandbox data store, i. e. information about existing network links and nodes. Currently, the platform supports simple network traffic metrics (e. g., packets, and error octets) and there is also support for flow-based monitoring and full-packet capture.

In order to cope with the largely heterogeneous monitoring data that is inherently generated within sandboxes and the KYPO platform itself, we use the normalizer design pattern and the notion of a monitoring bus component implementing this pattern, as described in detail by (Tovarňák and Pitner, 2014). The long-term objective of such a deployment is to render the monitoring architecture within the platform fully event-driven. This is motivated by the growing need for advanced monitoring data corelations both in the terms of real-time and post-mortem analysis.

During the development of the platform, we encountered a problem as to how to differentiate between the monitoring functionality that should be built in, and the functionality that should be, conceptually, a part of a cyber exercise scenario and the resulting sandbox topology. We have determined that a reasonable decisive factor is the intended consumer of the monitoring data and the desired intrusiveness of the monitoring components on the scenario.

For example, in the case of host-based monitoring, there is a need for various monitoring agents to be installed and configured on the end-nodes. If the intended consumer is not part of the scenario, e. g., the monitoring data are used for the purposes of progress tracking or scoring in cyber-exercises, the monitoring agents must be protected from misconfiguration and other manipulation by the participants. This, however, breaks the fourth-wall, so to say, since the participants need to be informed that such misconfiguration is prohibited, including network misconfiguration and so on. This can be sometimes seen as intrusive.

When the intended consumers are the participants themselves, the monitoring components and their configuration should be a part of the scenario. This way it can be misconfigured or stopped altogether. Yet in this case, the monitoring data can be rendered unusable for external consumers, e. g., for the purposes of ex-post analysis.

3.2.5 Platform Management Portal

The Platform Management Portal (PM Portal) mediates access to the platform for the end users by providing them with interactive visual tools. In particular, the PM Portal is designed to cover the following types of interactive services.

Management of cyber exercises – the preparation of cyber exercises is very complex process which requires us to define security scenarios, allocate hardware resources, manage participants, and so on. The PM Portal supports the automation of these tasks by introducing a system of user roles and corresponding interactions.

Collaboration – many security scenarios are based on mutual collaboration where multiple participants share a sandbox and jointly solve required tasks or, on the contrary, compete against each other. The PM Portal supports multiple flexible collaboration modes covering a wide range of scenarios.

Access to sandboxes – the PM Portal enables end users to log into computers allocated in a sandbox via remote desktop web client as an alternative user-friendly access point to the portal-independent command line SSH access.

Interactive visualizations – regardless of whether a user is analyzing a new malware or is learning new defence techniques against attackers, it is always crucial to understand and keep track of progress and current developments inside the sandbox. The PM Portal, therefore, provides specialized visualization and interaction techniques which mediate data and events measured in sandboxes.

4 User Interface and Interactions

The variability of security issues that the KYPO infrastructure is able to emulate places high demands on the realization of the Platform Management Portal and its interactive services. While traditional applications are usually based on clearly defined requirements and use cases that delimit software architecture as well as provided functionality, the design of the PM Portal has to deal with the dynamic character of its usage. This is because the use cases are defined at the user level as part of security scenarios and then user interfaces have to also be either definable or at least highly configurable at the user level.

To assure high accessibility of the services for all types of end users, the PM Portal is designed as a web

application where users are not bothered by the need to install anything on their device (not even browser plugins or extensions such as Java or Flash).

To deal with the dynamic character of the KYPO's use, the PM Portal complies with Java Enterprise Web Portal standards, as defined in JSR 168 and JSR 286. Web portals are designed to aggregate and personalize information through application-specific modules, so-called portlets. Portlets are unified cross-platform pluggable software components that visually appear as windows located on a web page. Once developed, a portlet can usually be reused in many security scenarios. Another key feature of enterprise web portals is their support of inter-portlet communication, synchronization and deployment into web pages and sites. We utilized these features to create complex scenario-specific user interfaces as preconfigured web pages composed of mutually cooperating portlets.

4.1 Role-based Access Control

Preparation of a cyber exercise is very complex task comprising scenario definition, allocation of resources, user management, and so on. In order to automatize these processes by means of user interaction, it is necessary to define user roles with clear access rules and responsibilities.

Scenarist devises security scenarios with all necessary details including sandbox definition and the design of web user interfaces for end users engaged in the scenario. At this level, the interfaces are defined as generic templates used to generate per-user web pages in further "scenario execution" phases. Besides the scenario and UI management, scenarists also authorize selected users to become organizers of exercises with adequate responsibilities.

An *organizer* is a well-instructed technically skilled person authorized by a scenarist to plan and prepare cyber exercises or experiments of a particular security scenario. Organizational activities consist of the allocation of sandboxes in the cloud, adjusting information pages, configuring a scoring subsystem and other scenario-specific services, inviting participants, etc. Organizers also delegate selected participants to be supervisors of the exercise.

Participants represent end users engaged in a particular cyber exercise or experiment. They utilize web UIs prepared by scenarists and perform tasks prescribed by the security scenario. If the users are involved in multiple experiments or exercises at the same time, they have to choose a particular one at the beginning of the interaction.

We distinguish between ordinary participants and those having extended supervising privileges. *Ordinary participants* have just one *scenario role* assigned. Scenario roles limit particular participants' access to particular hosts in the sandbox based scenario definition. For instance, an exercise scenario defines the roles of an attacker and a defender. The attacker then has no direct access to the hosts controlled by the defender and vice versa. In contrast, participants with *supervisor* privileges have access to all nodes in the network implicitly. Supervisors also usually utilize specific web forms and visualizations that reflect their specific needs. Another difference can be found in a multi-sandbox collaboration mode. While ordinary participants have access to only a single sandbox, supervisors can access all the sandboxes allocated for a given exercise.

Authentication of all users is based on federated identities. Credentials of users attempting to log into the PM Portal are redirected to a central system for identity management, which integrates many existing identity providers and authenticates users against their external electronic identities. Besides well-known identity providers (such as Facebook or Google) it is easy to integrate other external accounts on demand via a LDAP service. Participants of cyber exercises can, therefore, use their Google or corporate usernames and passwords to access the KYPO infrastructure.

Once authenticated, the authorization of a user is managed directly in the KYPO infrastructure. The PM Portal checks the user against his or her assigned roles and offers the appropriate web pages and portlets for further interaction. The more roles the user has assigned, the broader the user interfaces of the PM Portal are available.

4.2 Collaboration Modes

The combination of flexible web UIs (supported by the PM Portal) and the loose coupling of individual portlets (with sandboxes via remote access) enables us to simulate various collaboration strategies (Eichler et al., 2015). Three basic collaboration modes are depicted in Figure 2. The combination of these modes with other traditional web-browser features (such as multiple browser tabs opened at the same time or multi-display views) provide a very flexible solution covering a wide range of security scenarios.

Individual sandboxes – every participant has their own private sandbox and web user interface. The web UI is defined by a scenarist only once in the form of a template and then the participants have the same set of interactive tools available. Thanks to its cloud-based infrastructure, it is easy to allocate many identical sandboxes for individual users on demand. Neverthe-

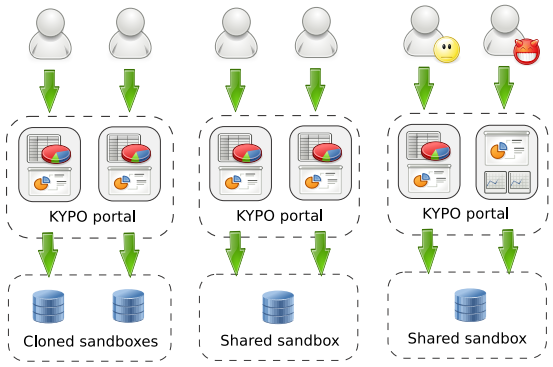

Figure 2: Collaboration modes: individual sandboxes, individual views on shared data and role-based collaboration.

less, sandboxes do not depend on each other. Therefore, participants can complete the same tasks via the same user interface but the state of sandboxes may differ depending on their activities. This collaboration mode is useful mainly for individual training and cyber security experiments.

Individual views on shared data – the participants, each of them sitting at his or her own computer, share a sandbox and the measured data are shared. Participants have the same web UI at their disposal but they use them independently. They can focus on different parts of the network, explore different aspects of the security scenario, return back in time and so on, but they never affect the views of other participants. This collaboration mode is useful mainly for collective learning about security threats or for collaborative forensic analyses.

Role-based collaboration – in this mixed approach, participants are divided into teams with prescribed roles, such as attackers or defenders. Teams have predefined web user interfaces according to their tasks. Teams can share a sandbox which plays the role of a battlefield. This collaboration mode is useful for exercises where multiple teams either cooperate or compete against each other in a single shared sandbox. However, this role-based approach extended with multiple sandboxes enables us to go even further. For example, we can create multiple defending teams, each having its own isolated sandbox, and a single attacking team fighting against them simultaneously.

4.3 Web Front-End and Visualization

The web portal technology used for the implementation of the PM Portal enables us to develop specialized interactive user interfaces, which are narrowly focused on specific goals, but also allow us to combine them easily into complex systems of mutually synchronized views supporting complex workflows. There is no space to describe all the developed user interfaces and interactive visualizations in detail, nor to discuss their combinations leading to the support of various security scenarios. Instead, we present only a few selected portlets that were used most often during various cyber exercises organized by the KYPO team so far.

4.3.1 Capture the Flag Games

The PM Portal offers complete support for designing and playing level-based games where users complete cyber security tasks. The administrators' interface enables the game designer to define the network topology, individual tasks, hints with penalties, time limits and other necessary information. There is also support for sandbox allocation and player enrollment. The players' interface guides them through the game and is usually supplemented with an interactive network topology view.

4.3.2 Network Topology

One key visualization of the PM Portal is a general topological view, as shown in Figure 3. Versatility was one of the key requirements for this visualization since the network topology is present in all scenarios. Routers, links, computers and servers are represented in the visualization.

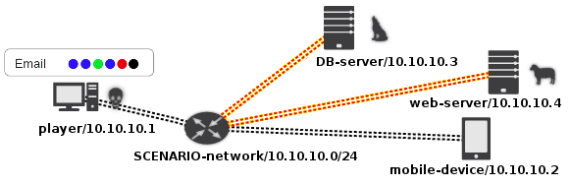

Figure 3: A simple network topology with highlighted roles, network traffic and incoming emails.

The topology visualization shows multiple dynamic data measured in the corresponding sandbox. The small icons close to the nodes represent logical roles, e. g., attacker or victim. Unusual traffic on links is visualized with colors and animations. Nodes can be accompanied by the visualization of user-defined events such as incoming emails. Clicking on a node, a user, if privileged, can access the node's remote desktop via VNC or SPICE client. In this case, a new tab is opened in the browser with the screen of the remote host. This visualization is fully interactive, enabling users to re-organize nodes, collapse, and reveal subnetworks, zoom in and out, and so on.

4.3.3 Time Manager

Data measured in sandboxes and provided by the monitoring management component have the form of a time series. Therefore, many visualizations used in the PM Portal have to cope with time-related data queries in order to show the sandbox's state either at a particular point in time or within a given time span. It would be impractical to deal with time constraints in every particular portlet independently. Instead, we developed a *Time Manager* portlet (Figure 4) which enable users to visually define time restrictions that are propagated to other portlets on the page.

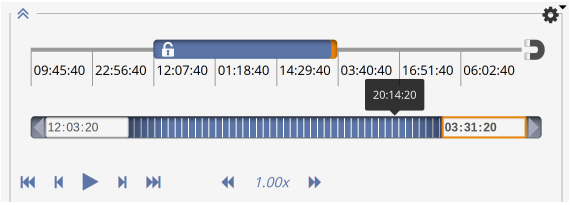

Figure 4: A generic timeline management visualization.

4.3.4 Analytic Graphs

KYPO provides several analytically oriented visualizations and interactions. For instance, measured sandbox data can be transformed into 2D line charts and radar charts or they can be visualized in 3D, as shown in Figure 5. These analytic graphs provide alternative views on multivariate data measured in the sandbox. To help users to identify anomalies, the visualizations are fully interactive, support the reordering of axises simply by their direct manipulation and can switch between two 3D views smoothly by means of animation so that the user keeps track of the investigated part of the graph and never loses context.

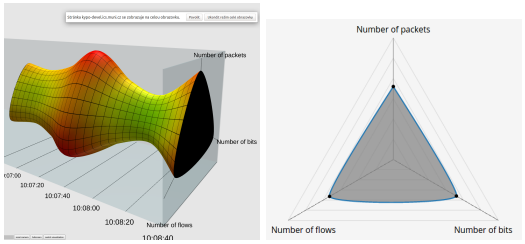

Figure 5: Interconnected analytic graphs.

4.4 Physical Facility

Although the KYPO cyber range is accessible remotely via a web browser, many exercises are organized in a physical KYPO laboratory. Its hardware equipment offers a high variability of display techniques as well as a reconfigurability of inputs and outputs so that it is possible to support variable collaboration strategies and to distribute relevant information across several teams and roles.

The room consists of a training area over which a multimedia control center and a visitors' gallery are located, as shown in Figure 6.

Figure 6: The KYPO laboratory is a versatile room. Its setting can be adjusted to best fit the needs of ongoing exercises.

The training area is equipped with six mobile audio-video tables, each for 3-4 learners. The tables integrate all-in-one touch computers providing access to the KYPO infrastructure. The room is further equipped with four mobile FullHD displays and two UHD/4K displays. These displays can be either connected to individual AV tables or used to display shared information. Nevertheless, the information sharing is primarily managed by two wide central display surfaces; a projection screen and display wall. The projection screen is 5 meters wide and supports FullHD 2D or 3D projection from up to 4 sources at the same time, e. g., the supervisor's screen and working spaces of 3 teams. The display wall consists of a matrix of 5x3 Full HD displays and a multi-touch frame supporting detection of up to 10 simultaneous touches (i. e. true multi-touch).

The distribution of content into display outputs is managed centrally by the coordinator sitting in the control center. Some basic tasks can be also managed directly by the supervisor, who has a table located together with the AV tables in the training area.

5 Cyber Range Use Cases

KYPO can be used for various different applications. During its design and development, we focused on these three main use cases: *i)* cyber research, development, and testing, *ii)* digital forensic analysis,

and *iii)* cyber security education and training. All these use cases have a similar set of requirements on the cyber range, but they differ in scenario-specific tools, the availability of pre-defined content, user interactions and expected knowledge, skills, and effort level of the users. However, the concept of sandboxes and the platform management portal helps us to cope with this fact, i. e. various types of sandboxes with various types of tools can be provided, from an empty sandbox for researchers to a fully populated and configured sandbox for a complex cyber security exercise. In the following text, we describe the differences of the three use cases in a detail, and provide references to research papers employing or benefiting from an application of the KYPO cyber range.

5.1 Cyber Research and Development

The first use case presented here supports research, development, and testing new methods or systems for the detection and mitigation of cyber attacks in network infrastructures of various types.

In this case, KYPO provides a sandbox and optional monitoring infrastructure for experiments. Users can provide their own virtual images for hosts to be instantiated. Alternatively, they can start with generic virtual hosts available in KYPO which run common operating systems, services and applications (e. g., Ubuntu Server, MS Windows Server 2013, Debian Server with configured DNS server) and install applications used in the experiment.

Network traffic and host based statistics can be monitored and stored within KYPO's infrastructure, where they are immediately available for analysis. Experiments can be evaluated via analytic tools that researchers deploy into the KYPO infrastructure and utilize according to their interests. Researchers can also utilize interactive visualizations of the PM Portal. The network topology visualization (with an indication of network traffic and event-based activities together with 2D and 3D analytic graphs) is especially valuable. The time manager helps to keep track of real-time developments in the sandbox.

This use case is intended for security researchers and experienced network administrators because it requires an advanced level of knowledge in networking, host configuration and some knowledge of virtualization technologies. Researchers need to be experienced in order to assemble or adjust their experiment-specific web UIs, to define their own topologies and other scenario properties, and to properly design multiple sandboxes for comparison studies. Regarding the KYPO user roles, researchers play the role of *scenarist*, *organizer* and *supervisor*.

There are several public papers using KYPO for cyber security research and development ranging from a simulation of a DDoS attack (Jirsík et al., 2014) through to an evaluation of a network defence strategy (Medková et al., 2017) and an analysis of surveillance software (Špaček et al., 2017).

5.2 Digital Forensic Analysis

The second use case partially builds upon the previous one and covers basic forensic analysis, which can be partly automated by tools deployed in the sandbox. In this use case, the users can deploy virtual images of unknown or malicious machines in the predefined sandbox network and run a set of automated dynamic analyses. The sandbox contains an analytic host that provides pre-configured tools and an environment for rudimentary forensic analysis.

This use case supports security incident handlers and forensic analysts in focusing on the subject matter and removes the burden of spending their precious time in the setup of an analytic environment. Since the digital forensic analysis extends the previous use case, the required KYPO user roles are also similar.

5.3 Education and Training

The last use case covers a diverse type of educational hands-on activities, such as security challenges, competitions, capture the flag games, and attack/defence cyber exercises; all of which closely follow the *learning-by-doing* principle.

In our experience, the education and training use case has proven to be the most challenging. On one hand, the KYPO platform needs to provide many additional features, mainly in the terms of user interactions, in order to support both the learners and educators in their roles. On the other hand, there is a considerable amount of customized content that must be created in order to fit a particular educational activity, whilst remaining reusable (e. g., virtual hosts, exercise data stored at hosts).

Some activities, e. g., capture the flag games, are designed to be held without much direct input from the teacher. Instead, the assignments for the learners are implanted into the platform where the game is deployed, including additional instructions and an evaluation of the submitted solutions. The learners typically choose individual tasks or follow the predefined path of the game. Once they find a solution, they submit the requested data to the game platform which immediately provides a response whether the solution is correct or not. If it is, they can proceed further.

In the other cases, it is desirable for the educators to be able to control the flow of the hands-on activity based on automatically acquired status information about the simulated infrastructure in the sandbox and also manually trigger tasks for learners, and evaluate their actions and reports.

Whatever the case, it is desirable to put minimal requirements on the learners' knowledge of the KYPO infrastructure, virtualization technologies and other advanced concepts. As a result, the learners can focus only on the subject of the exercise or training, such as a penetration testing tutorial or a cyber security game.

With regard to KYPO's user roles, learners follow the *scenario roles* assigned to them, and interact with predefined web user interfaces. Instructors have *supervisor* privileges to keep track on learners' activities and to be able to interfere in their activities if necessary. In contrast, substantial preparation effort and technical skills are required from *scenarists* and *organizers* who create the content of exercises, allocate resources and manage the preparation and execution phase.

This use case motivates further research in active learning of cyber security. We evaluated the benefits of design enhancements in generic capture the flag game scheme provided by KYPO platform (Vykopal and Barták, 2016). Next, we introduced methods of distributing learners into teams with respect to their proficiency and the prerequisite skills required by a cyber exercise (Vykopal and Cegan, 2017).

6 Conclusion

Today, KYPO is the largest academic cyber range in the Czech Republic. The platform is fully cloud-based and supports multiple use cases (research, education and training). We organize national cyber exercises and training sessions to validate proposed cyber range components and to continually improve them. We also use KYPO for hands-on security courses to give students realistic experience in cyber security.

Our current work focuses on research into tools for more realistic, economical, and time efficient simulations of real cyber entities. We develop tools to further automate the preparation and execution of cyber experiments. We connect KYPO to other facilities (e. g., ICS and LTE networks) to create a more realistic cyber-physical environment. We aim to execute current and sophisticated cyber attacks in the KYPO infrastructure to provide a research environment for simulation, detection, and mitigation of cyber threats against critical infrastructure.

In addition to the technology based contributions, we would like to contribute to transdisciplinary learning in cyber security to cope with the ever-evolving threat landscape. To make a desirable improvement in the skills of the learners, technical skills must be complemented by communication, strategy and other skills for effective attack detection and response.

Acknowledgements

This research was supported by the Security Research Programme of the Czech Republic 2015-2020 (BV III/1 – VS) granted by the Ministry of the Interior of the Czech Republic under No. VI20162019014 – Simulation, detection, and mitigation of cyber threats endangering critical infrastructure.

Access to the CERIT-SC computing and storage facilities provided by the CERIT-SC Center, provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CERIT Scientific Cloud LM2015085), is greatly appreciated.

REFERENCES

Avatao (Last accessed on Mar 22, 2017). https://avatao.com/.

Benzel, T. (2011). The Science of Cyber Security Experimentation: The DETER Project. In *Proceedings of the 27th Annual Computer Security Applications Conference*, pages 137–148. ACM.

Buttyán, L., Félegyházi, M., and Pék, G. (2016). Mentoring talent in IT security–A case study. In *2016 USENIX Workshop on Advances in Security Education (ASE 16)*.

Childers, N., Boe, B., Cavallaro, L., Cavedon, L., Cova, M., Egele, M., and Vigna, G. (2010). Organizing large scale hacking competitions. In *Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 132–152. Springer.

CTF365 (Last accessed on Mar 22, 2017). Capture The Flag 365. https://ctf365.com/.

Cyber Virtual Ad hoc Network (Last accessed on Mar 22, 2017). http://www.appcomsci.com/research/tools/cybervan.

Davis, J. and Magrath, S. (2013). A survey of cyber ranges and testbeds. Technical report, DTIC Document.

EDURange (Last accessed on Mar 22, 2017). http://www.edurange.org/.

Eichler, Z., Ošlejšek, R., and Toth, D. (2015). *KYPO: A Tool for Collaborative Study of Cyberattacks in Safe Cloud Environment*, pages 190–199. Springer International Publishing, Cham.

Emulab (Last accessed on Mar 22, 2017). A list of Emulab Testbeds. http://wiki.emulab.net/Emulab/wiki/OtherEmulabs.

Faber, T. and Wroclawski, J. (2009). A federated experiment environment for Emulab-based testbeds. In *TRIDENTCOM*, pages 1–10.

Ferguson, B., Tall, A., and Olsen, D. (2014). National cyber range overview. In *2014 IEEE Military Communications Conference*, pages 123–128.

Jirsík, T., Husák, M., Čeleda, P., and Eichler, Z. (2014). Cloud-based security research testbed: A DDoS use case. In *2014 IEEE Network Operations and Management Symposium (NOMS)*.

KYPO (Last accessed on May 16, 2017). https://kypo.cz/.

Lee Rossey (2015). SimSpace Cyber Range. ACSAC 2015 Panel: Cyber Experimentation of the Future (CEF): Catalyzing a New Generation of Experimental Cybersecurity Research.

Linux Foundation (Last accessed on Mar 22, 2017). Open vSwitch. http://openvswitch.org/.

MCR (Last accessed on Mar 22, 2017). The Michigan Cyber Range. https://www.merit.edu/cyberrange/.

Medková, J., Husák, M., Vizváry, M., and Čeleda, P. (2017). Honeypot Testbed for Network Defence Strategy Evaluation. In *Proceedings of the 2017 IFIP/IEEE International Symposium on Integrated Network Management*, pages 887–888. IEEE Computer Society.

Mirkovic, J., Benzel, T. V., Faber, T., Braden, R., Wroclawski, J. T., and Schwab, S. (2010). The DETER Project. In *2010 IEEE International Conference on Technologies for Homeland Security (HST '10)*.

NATO CCDCOE (Last accessed on May 16, 2017). Locked Shields. http://ccdcoe.org/event/cyber-defence-exercises.html.

NCR (Last accessed on Mar 22, 2017). The National Cyber Range. http://www.acq.osd.mil/dte-trmc/docs/Docs/NCR/2015_NCR

Špaček, S., Čeleda, P., Drašar, M., and Vizváry, M. (2017). Analyzing an Off-the-Shelf Surveillance Software: Hacking Team Case Study. *Security and Protection of Information*, (IX).

Raj, A. S., Alangot, B., Prabhu, S., and Achuthan, K. (2016). Scalable and Lightweight CTF Infrastructures Using Application Containers. In *2016 USENIX Workshop on Advances in Security Education (ASE 16)*, Austin, TX. USENIX Association.

Schwab, S., Wilson, B., Ko, C., and Hussain, A. (2007). SEER: A security experimentation environment for DETER. In *Proceedings of the DETER Community Workshop on Cyber Security Experimentation and Test on DETER Community Workshop on Cyber Security Experimentation and Test 2007*. USENIX Association.

Security Competence (Last accessed on Mar 22, 2017). Hacking-Lab. http://www.hacking-lab-ctf.com/technical.html.

Siaterlis, C., Garcia, A. P., and Genge, B. (2013). On the Use of Emulab Testbeds for Scientifically Rigorous Experiments. *IEEE Communications Surveys Tutorials*, 15(2):929–942.

Tovarňák, D. and Pitner, T. (2014). Continuous queries over distributed streams of heterogeneous monitoring data in cloud datacenters. In *2014 9th International Conference on Software Engineering and Applications (ICSOFT-EA)*, pages 470–481.

Vigna, G., Borgolte, K., Corbetta, J., Doupe, A., Fratantonio, Y., Invernizzi, L., Kirat, D., and Shoshitaishvili, Y. (2014). Ten Years of iCTF: The Good, The Bad, and The Ugly. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*.

Vykopal, J. and Barták, M. (2016). On the Design of Security Games: From Frustrating to Engaging Learning. In *2016 USENIX Workshop on Advances in Security Education (ASE 16)*, Austin, TX. USENIX Association.

Vykopal, J. and Cegan, J. (2017). Finding Exercise Equilibrium: How to Support the Game Balance at the Very Beginning? In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '17, pages 719–719, New York, NY, USA. ACM.

Welch, D., Ragsdale, D., and Schepens, W. (2002). Training for information assurance. *Computer*, 35(4):30–37.

White, B., Lepreau, J., Stoller, L., Ricci, R., Guruprasad, S., Newbold, M., Hibler, M., Barb, C., and Joglekar, A. (2002). An Integrated Experimental Environment for Distributed Systems and Networks. pages 255–270, Boston, MA.

Wroclawski, J., Mirkovic, J., Faber, T., and Schwab, S. (2008). A two-constraint approach to risky cybersecurity experiment management. In *Sarnoff Symposium*. Invited paper.