UNIVERSITY OF TARTU
Institute of Computer Science
Cyber Security Curriculum

Allar Vallaots

Federation of Cyber Ranges

Master's Thesis (30 ECTS)

Supervisors: Jaan Priisalu, MSc
Uko Valtenberg, MSc
Raimundas Matulevičius, PhD

Tartu 2017

Federation of Cyber Ranges

Abstract:

An essential element of the cyber defence capability is highly skilled and well-trained personnel. Enhancing awareness and education of technicians, operators and decision makers can be done through multinational exercises. It is unthinkable to use an operational production environment to train attack and defence of the IT system. For simulating a life like environment, a cyber range can be used.

There are many emerging and operational cyber ranges in the EU and NATO. To benefit more from available resources, a federated cyber range environment for multinational cyber defence exercises can be built upon the current facilities. Federation can be achieved after agreements between nations and understanding of the technologies and limitations of different national ranges.

This study compares two cyber ranges and looks into possibilities of pooling and sharing of national facilities and to the establishment of a logical federation of interconnected cyber ranges. The thesis gives recommendations on information flow, proof of concept, guidelines and prerequisites to achieve an initial interconnection with pooling and sharing capabilities. Different technologies and operational aspects are discussed and their impact is analysed.

To better understand concepts and assumptions of federation, a test environment with Estonian and Czech national cyber ranges was created. Different aspects of network parameters, virtual machine manipulations, virtualization technologies and open source administration tools were tested. Some surprising and positive outcomes were in the result of the tests, making logical federation technologically easier and more achievable than expected.

The thesis is in English and contains 42 pages of text, 7 chapters, 12 figures and 4 tables.

Keywords:

Cyber Range, NATO, federation, virtualization, multinational cyber defence exercises

CERCS: T330 Military science and technology

Küberharjutusväljakute Ühendamine

Lühikokkuvõte:

Küberkaitse võimekuse aluselemendiks on kõrgete oskustega ja kokku treeninud spetsialistid. Tehnikute, operaatorite ja otsustajate teadlikkust ja oskusi saab treenida läbi rahvusvaheliste õppuste. On mõeldamatu, et kaitse ja rünnakute harjutamiseks kasutatakse toimivat reaalajalist organisatsiooni IT-süsteemi. Päriseluliste süsteemide simuleerimiseks on võimalik kasutada küberharjutusväljakuid.

NATO ja Euroopa Liidu liikmesriikides on mitmed juba toimivad ja käimasolevad arendusprojektid uute küberharjutusväljakute loomiseks. Et olemasolevast ressurssi täies mahus kasutada, tuleks kõik sellised harjutusväljakud rahvusvaheliste õppuste tarbeks ühendada. Ühenduvus on võimalik saavutada alles pärast kokkuleppeid, tehnoloogiate ja erinevate harjutusväljakute kitsenduste arvestamist.

Antud lõputöö vaatleb kahte küberharjutusväljakut ja uurib võimalusi, kuidas on võimalik rahvuslike harjutusväljakute ressursse jagada ja luua ühendatud testide ja õppuste keskkond rahvusvahelisteks küberkaitseõppusteks. Lõputöö annab soovitusi informatsiooni voogudest, testkontseptsioonidest ja eeldustest, kuidas saavutada ühendused ressursside jagamise võimekusega. Vaadeldakse erinevaid tehnoloogiad ja operatsioonilisi aspekte ning hinnatakse nende mõju.

Et paremini mõista harjutusväljakute ühendamist, on üles seatud testkeskkond Eesti ja Tšehhi laborite infrastruktuuride vahel. Testiti erinevaid võrguparameetreid, operatsioone virtuaalmasinatega, virtualiseerimise tehnoloogiad ning keskkonna haldust avatud lähtekoodiga tööriistadega. Testide tulemused olid üllatavad ja positiivsed, muutes ühendatud küberharjutusväljakute kontseptsiooni saavutamise oodatust lihtsamaks.

Magistritöö on kirjutatud inglise keeles ja sisaldab teksti 42 leheküljel, 7 peatükki, 12 joonist ja 4 tabelit.

Võtmesõnad:

Küberharjutusväljak, NATO, ühendamine, virtualiseerimine, rahvusvahelised küberkatise õppused

CERCS: T330 Sõjandus ja militaartehnoloogia

Table of Contents

List of used abbreviations and definitions

ACL – Access List

API - Application Programming Interface

ASOnet – Network provided for Estonian governmental institutions

ASA – Adaptive Security Appliance

ASR - Aggregation Services Router

ATM - Asynchronous Transfer Mode

BGP - Border Gateway Protocol

CCD CoE – Cooperative Cyber Defence Centre of Excellence

CDX – Cyber Defence Exercise

CERIT - Centre for Education, Research and Innovation for ICT

CERIT-SC – Scientific Cloud

CPE – Customer-Provided Equipment

CPU – Central Processing Unit

CR – Cyber Range

CZE – Czech Republic

DNS – Domain Name System

EDF CR – Estonian Defence Forces Cyber Range

EMC – Enterprise storage system provided by Dell

ESX - Industry-leading, purpose-built bare-metal hypervisor provided by VMware

EU – European Union

FTP – File Transfer Protocol

GB – Giga Byte

HA – High Availability

IOS - Internetwork Operating System

IP – Internet Protocol

IPS – Intrusion Prevention System

IPSec – Internet Protocol Security

IT – Information Technology

KB – Kilo Byte

KVM – Kernel-based Virtual Machine

KYPO - Kybernetický polygon (Cyber Exercise & Research Platform)

L2 – Layer 2

L2VPN – Layer 2 Virtual Private Network

LAN – Local Area Network

MB – Mega Byte

MPLS - Multiprotocol Label Switching

NATO – North Atlantic Treaty Organization

NS – NATO Secret

NU – NATO Unclassified

OpenVPN - an open source VPN that is useful for creating Point-to-Point and Site-to-Site connections.

PPTP – Point-to-Point Tunneling Protocol

RAM – Random Access Memory

RDP – Remote Desktop Protocol

SLA – Service Level Agreement

SSH – Secure Shell

SSL – Secure Socket Layer

SSO – Single Sign-On

TB – Tera Byte

TFTP – Trivial File Transfer Protocol

TLS – Transport Layer Security

TCP – Transmission Control Protocol

UCS – Unified Computing System

UDP – User Datagram Protocol

UI – User Interface

vLab – Virtual Lab Manager; automation software for creating virtual exercise environments.

VM – Virtual machine

VMRC – Virtual Machine Remote Console

VNX – Hybrid-Flash storage solution provided by Dell EMC

VPN – Virtual Private Network

VTEP - VXLAN Tunnel End Point

VXLAN - Virtual Extensible Local Area Network

Xen - Open source virtualization platform

1. **Introduction**

An essential element of the cyber defence capability is highly skilled and well-trained personnel. Enhancing awareness and education of technicians, operators and decision makers is urgent. The need for Cyber Ranges (CR) to support training and exercises are both essential and scarce. The scarceness of Cyber Ranges' facilities in support of training and exercises is recognised as an important capability gap that urgently needs resolution in EU's and NATO's institutions [1].

Cyber range functions like a military training ground, facilitating training in "weapons", operations or tactics. Thus, cyber warriors and IT professionals employed by various agencies train, develop and test cyber range technologies to ensure consistent operations and readiness for real world deployment. A cyber range conceptually consist of a research range, a simulation & test range and a training & exercise range as shown in figure 1. This thesis will focus mainly on the training and exercise part, because it is considered the area, where sharing resources will give the highest pay off value in the near future [1].

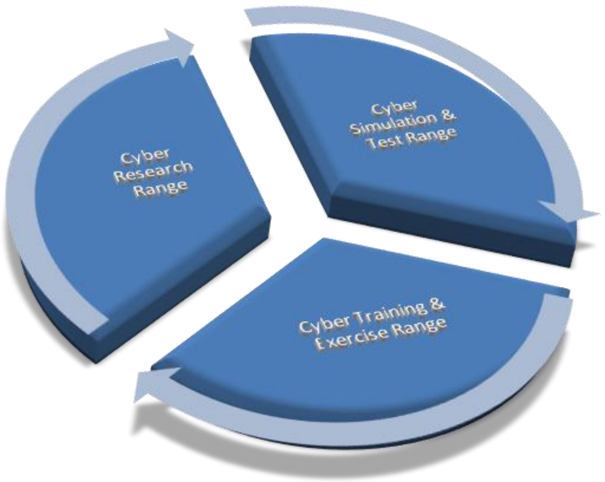

Cyber Research Range
Cyber Simulation & Test Range
Cyber Training & Exercise Range

Figure 1. Cyber Range conceptual parts [1].

Currently NATO, EU and many nations have built or are planning to build a private cloud based training and simulation environment - cyber range. There are many resources and knowledge available with current and emerging ranges. Some of these ranges have unique services and resources like computing power, different storage solutions or custom tools developed specially to fulfil certain tasks. Combining logically those ranges and their services could create a high availability and a big shared resource pool of federated (hybrid) ranges. Interoperability of cyber ranges will have a positive effect on the interoperability of operational cyber defence systems, organisations and processes, thereby improving the effectiveness and efficiency of cyber defence operations and multinational exercises. The focus shall not lie on connecting the ranges on network transportation level but managing and assigning available resources dynamically.

Ambition for the study is to compare two ranges and look into possibilities of pooling and sharing of national facilities and to the establishment of a (logical and physical) federation of interconnected cyber ranges. The thesis will try to give recommendations on information flow, proof of concept tests can be conducted and guidelines and prerequisites are provided to achieve an initial interconnection with pooling and sharing capabilities. Information and descriptions provided by the Estonian range, various institutions and projects mentioned in the thesis come from authors personal experience and involvement as the deputy chief of the Estonian Cyber Range.

Planned thesis will use an empirical study of comparing different range platforms and technologies used. From the technical point of view, there is a need to analyse the underlying infrastructures like hypervisors, networking and specialised tools or automation systems. A quantitative analysis has to be conducted and from that, deduction of prerequisites for federation can be made.

One of the key assumption is that all the cyber ranges are considered “black boxes” and getting all the needed information can only be trust based. For that, cooperation with the Czech Cyber range has been established. The research will not develop new services, it analyses currently available services and comes up with command and control type of resource sharing and pooling for a federated cyber range environment [1].

1.1. Related work

While governmental cyber ranges are emerging in different countries, not much information is available publicly. Works available give a good overview what kind of cyber ranges are already available and what can be expected from future emerging ranges. One of the best and diverse overview of possible resources is described in the Australian Government Department of Defence survey on Cyber Ranges and Testbeds. While this paper gives a good overview of war-gaming, simulation tools and validation environment, it does not focus on red on blue exercises [2].

A more detailed approach on hands-on training and cyber defence exercises is discussed in the 2009 paper of Replicating and Sharing Computer Security Laboratory Environments. This research mainly investigates cyber ranges in the United States Military Academy, the University of New Mexico, Carnegie Mellon University, and the University of Alaska. In addition to the description of the environments, some future thoughts are discussed on resource sharing nationally to meet the needs of students and institutions dealing with high-capability virtual labs and supercomputing [3].

For a more specific overview and possible use cases of the Estonian Cyber Range a 2012 thesis on Cyber Defence training and exercise environment use cases dealing with technology considerations and aspects of exercise environments [4].

1.2. Contributions

The main contribution of this paper is to review previous work and technologies on cyber ranges and data centres from military, academic and private sectors and to discuss resource sharing between cyber ranges for multinational cyber defence exercises. The review discusses general considerations of creating a framework for cyber ranges federation with political, legislative and procedural aspects within NATO and EU institutions.

The remainder of this thesis is structured as follows. Background information about Estonian and Czech ranges and general technological considerations of federation are provided in Section 4. Section 5 describes the test environment set up to test the validity of federation between the two ranges. This also includes test case plans for testing virtual machine (VM) operations in different geographic sites, management with open source and enterprise tools and network parameter manipulations within the federated test environment. The next section analyses the test results and draws conclusions on the possibility of federation in high latency low-bandwidth environments, virtual machine operations, fault tolerance and high availability and the use of open source tools for management. Configurations used in the test environment are presented in the appendix of the thesis.

2. Goal of the study

Combining and connecting logically cyber ranges and their services creates high availability and a big, shared resource pool of different services in a federated (hybrid) exercises and training environment. Technical and strategic components contribute in training of IT specialists and management in a life like high stress scenario. Goal of the thesis is to understand, how to manage and connect available CR resources between nations and how the connected federated environment can be used during multinational exercises. This study does not focus purely on equipment and facilities. To better understand federation, it is important to provide a framework for connecting and sharing available cyber range resources for multinational use of existing and emerging facilities.

2.1. Secondary goals

To achieve the primary goal, some secondary goals shall be met. The first step is the identification of the common functional requirements for cyber ranges. Common functional requirements should address logical and technical network interface specifications for interconnection of CR-s. After understanding the interconnection specification, a common network architecture needs to be designed. This thesis gives some recommendations for network designs and ideas based on the experience of the Estonian Cyber Range.

Connecting Cyber Ranges, should increase availability and fault tolerance of existing Cyber Range facilities for multinational exercises. The thesis will provide some practical tests to understand how availability between different geographical locations is affected.

Tests carried out in this thesis will give answers, how interconnected cyber ranges of member states can form together a bigger range for large scale and complex exercises. They will also point out the emerging problems and bottlenecks what might occur in the future. Interconnection of cyber ranges should increase the capacity, performance of national ranges, thereby creating magnitude and diversity that would not be achievable on a national scale.

This thesis will create a framework for federation, that gives recommendations for defining procedures to achieve interconnection. Legal, organisational and procedural arrangements for interconnection of cyber ranges will be addressed.

Additionally, connecting cyber exercise ranges with similar installations around the world will greatly enhance the ability to train with teams that, despite the universal language of information and communications technology, have different cultural approaches to problem solving, as well as capabilities aimed at different threats. Federation helps to build new relations, strengthens and deepen trust, cooperation across the borders and helping to connect government, military, private and academia sectors.

2.2. Hypothesis

Because cyber ranges are considered black boxes and every connection between them can be specific with certain parameters, some hypothesises prior actual testing can be framed. It can be assumed that with current technologies, federation between different geographic locations is possible. To achieve the final goal, different aspects and technologies should be reviewed. People talking about federation in NATO and EU institutions seemed to have formed some expectations what federation will provide. This thesis will try to research some of these expectations.

1. Operating a remote site from a different geographic location will be affected by the network parameters between datacentres.
2. Open source tools help to make federation management cost effective and less labour intensive.
3. Federation creates fault tolerance and disaster recovery for every connected range.

Analysing those aspects in a multi-site cyber range environment and conducting tests with network parameters and different software solutions will provide future references to carry out further analysis for the federated range environment.

3. Methodology

This research will use a normative intensive empirical study. Empirical study is based on experimentation, systematic observations and measurements. Normative part of the methodology should give an answer how federation could be achieved based on one use case. Intensive approach is used to due federation test is carried out with one counterpart, the Czech cyber range. This way, it is easier to focus on a more specific practical problem. Because of the restricted number of objects, it is possible to study the problem in a more deeply, thus achieving a better understanding of the specific use case [5]. An extensive study could be used with more than two range interconnection simultaneously. Meaning one range is connected to more than two other ranges. To carry out the current study, technical and organizational information from Czech Cyber Range needs to be collected. Underlying infrastructure, hypervisors and software solutions used in this range is of utmost importance to achieve the goal of a federation between current two ranges. The high-level goal of the current study is to attempt to improve the current situation of cyber ranges in the European Union and NATO, to provide flexibility and resources for conducting international Cyber Defence exercises. Steps for achieving the goal within the empirical research method include:

1. Defining the target, which usually is to remove a widespread problem in present activity or in present production and/or to correct an outdated passage in existing theory. Defining the general principles and goals that have to be observed in the work. An essential component of the target is also declaring the point of view that shall be used when making the normative proposals.
2. Stating which facts in the context have to be taken as "given" facts that cannot be modified.
3. Planning how to fulfil the target. This is done preferably as several alternatives, including one, where the present state of things continues as such.
4. Selecting the alternative what is best. This can either be the one, which fulfils best the targets, or the cheapest of the acceptable alternatives.
5. Asking opinions or statements from interested parties.
6. Presenting the proposals to organisations affected and interested, which can either accepts the work or ask for new alternatives [5].

In the first step, the technical information about the KYPO range needs to be collected and analysed. To achieve this, initial communication with different national points of contacts needs to be attained and applicability of the information should be discussed. In a multinational environment, this is one of the slowest tasks to complete. Especially, when to consider NATO and EU command structures. To achieve this, agreements need to be signed by different countries. Luckily, for this project, several EU and NATO countries have ratified an agreement of sharing information on federation.

After the first step, it is possible to start defining principles of achieving initial logical federation. During this phase, the technical limitations based by information provided will be analysed. Initial assumptions of logical federation based on technologies can be made. Technology analysis will provide future limitations and principles for building and implementing a test environment. Within the scope of the logical federation discussions, multiple technologies and alternatives should be considered.

After all the limitations, technologies and alternatives have been considered, plans for the test cases will be planned. This will be based on the best-found variant between two ranges. In the context of this thesis, the variant will be based upon the experience and technologies mainly used by the EDF CR. Also taking into consideration that the test environment will be built principally by one person.

The test environment needs to be set up as life like as possible, to get the most accurate results for defining problems and understanding bottlenecks. Implemented test environment can be used as the basis for future federation. Further developments can be deployed and tested on the same or a duplicated environment. The tests may also point out the unsuitability of the variant deemed as the best and new alternatives can be considered.

A vital step for understanding all the concepts of federation, is the input and feedback from the other interested party. Asking for feedback will not end with the defence process of this thesis.

All the results documented in the current study will be presented to interested NATO and EU bodies (working groups) dealing with future federated networks.

4. Requirements of federation

In order to achieve federation between different national cyber ranges, information gathering is the key objective. Information from all the parties, who plan to connect to the federated environment, is needed and analysed. Understanding the technologies and limitations will ultimately lead to interconnection of ranges and environments. When ranges are connected, processes and documentation for joining the environment needs to be available. General processes of maintaining a range in the environment need to be defined and agreed. Sharing resources within the federated environment should be done per exercise basis. During the planning phase of an exercise, resources should be analysed and allocated. Allocations need to consider, that reassessment of capabilities can happen during exercise phases.

Every national range should have the ability to maintain, update and configure their own environment within the federated cloud. A central authority is needed for providing monitoring, situational awareness and logging tools within the federated cloud. This authority should maintain the documentation of the federated environment in cooperation with national point of contacts. The task of contacting point of contacts in case of emergencies should lie with the focal entity.

Centralized chat services, user portals and authentication mechanism need to be set up and maintained. Additionally, core and aiding systems should to be considered. During exercises, SLA-s between affected parties, need to be determined.

In periods between exercises, the availability of system does not have to be a 100%, but still some level of redundancy in federated infrastructure is required. Acceptable downtime of different services during events may vary. During the development and planning phase 30 minutes of downtime does not have a considerable impact. During multinational exercises, while hundreds of people are connected to the environment, even a 10-minute break cannot be tolerated. To better forecast downtime, every range should present an annual maintenance plan. From this, resources for exercises can be planned. For ad-hoc maintenances, the central authority should notify every member state.

Before multinational exercises, the performance parameters need to be considered. How large events need to be supported? The performance counters for CPUs, memory and storage needs to be allocated for exercises. The location of the training audiences is important and the way they can use allocated resources. Amount of generated and user simulated traffic should be adjustable between different ranges and needs to be lowered if the utilization of the systems exceeds the thresholds. After familiarization period or test runs, the whole environment needs to be reverted. This means that lot of overhead for storage systems is created. For planning exercise resources, mitigation plans and processes should consider the heightened performance necessity of storage systems.

In the planning phase of federation, the networking of different ranges should be considered. After initial set up of the federated environment, the work for future expansion of connected ranges needs to be started. Technical planning, like IP addressing, physical ports of network devices and redundant links should be planned and implemented with redundancy and reserves. While the federated cloud grows, a network topology of endpoints should be considered and agreed. Possible variants for connecting ranges could be either in a star or in a mesh topology. Mesh topology should be preferred, because of redundancy and higher tolerance of failures. The answer for topology questions should be considered case-by-case basis. The cost of implementing different topologies on a case-by-case basis should provide indications of the best variants.

The environment has to be flexible enough to support multiple hypervisors. The type of primary hypervisor should be bare-metal, instead of a hosted solution in order to get best performance and advanced features. Bare-metal hypervisors (VMware and KVM) are more scalable. Container-based hypervisors could be used for specific tasks, like building a traffic generation botnet from lightweight Linux containers. They are not suitable for running wide range of different operating systems in the same sandbox-like environment. While talking about the general management and monitoring software, the functionality, maturity and stability are essential. A specially built software might be the best variant for using, but then the scalability and flexibility concerns must be incorporated. Additionally, some widely known open source software with multiple hypervisor platform support can be an alternative. Furthermore, the software supporting these operations must be stable and easy to operate. The management software should have a granular set of permissions in order to define which users can access specific objects in a certain way. While conducting multinational exercises, an access to the management system is also needed for the training audience with the minimum operational activities allowed with their virtual machines. Functionality for powering on and off, restarting, reverting to snapshots and accessing the console of a virtual machine.

Every range should provide their first line of perimeter security. Firewalls and IPS systems on the system border preventing malicious traffic from the Internet. During exercises, certain security requirements (for NU and NS) need to be met. Mechanisms for preventing DDoS attacks and white lists must be agreed before every multinational exercise and applied on every nationally provided recourse.

4.1. Estonian Cyber Range

The Estonian Cyber Range project was initiated in 2011, with the main goal of supporting the development of Estonian cyber defence capabilities. It is a government-financed range under military command. Functionalities of the range are not limited to military requirements, but are also targeted to support national and international initiatives, which contribute to the development of cyber defence capabilities, increase of multinational co-operation and enhance long-term cyber security resilience [6].

Estonian Cyber Range is a multinationally usable platform for cyber defence exercises, training and education. EDF CR has unique features, offering interactive team training with “live-fire” exercises, realistic high-stress scenarios and reliable performance assessment. The range is a useful and cost effective solution for cyber defence concept development as well as for testing software enabled solutions. It has been used in multinational exercises Cyber Coalition and Locked Shields and in multiple trainings for the CCD CoE and Tallinn University of Technology. [6]

Server infrastructure is based on the Cisco UCS (Unified Computing System). Three different generations of UCS blade servers are available with the total amount of roughly 1400 CPU cores and 12 TB of RAM.

The data storage is built on the EMC VNX and XtremIO platforms and is connected with the cyber range data centre via multiple 8Gbit/s fiber-channel links. VNX provides slower spinning drives for low resource demanding operations with the total capacity of 140TB. XtremIO provides all SSD ultra-fast storage unit what enables to concentrate the exercise data with deduplication with the usable capacity of 30TB of raw storage. With de-duplication can go up to 250TB.

There are several network connection possibilities for the Estonian Cyber Range. For everyday operation, range uses the Estonian national ASOnet connections. For exercises, there is available a secondary internet connection provided by Telia (the biggest ISP in Estonia). The network solution provides the range with one 1Gbit/s and multiple 10Gbit/s Internet connections. The network devices are duplicated. All the interconnections between the devices are made via duplicated 10Gbit/s connectors.

The first line of defence in the Cyber Range is provided by Cisco firewalls and SourceFire IPS solution. According to licencing, these firewalls provide a secure VPN connectivity up to 500 end-users and unlimited site-to-site tunnels. For fault tolerance, VPN concentrators are configured in active-passive mode. The Cyber Range is remotely accessible all over of the world, as long as the client has Internet connectivity.

Automation of the Cyber Range is provided by vLab Manager automation software, which enables to configure and design exercise environments. It is a universal tool for managing resources and visualizing workflows. The software supports VMware platform. In the future, additions of other known virtualization platforms will be implemented. (Hyper-V, Xen, VirtualBox, and KVM).

Monitoring of the Cyber Range environment is conducted with Zabbix and Observium. For Internet traffic simulation, Range has used the iXia Breaking Point device and other alternative solutions.

Log collection and analysis is conducted by VMware vCentre Log Insight software and open source tools. For log collection and visualization of the gaming environment, the Cyber Range uses the CDX Situational Awareness v2012 software, which gives an overview of the situation in practically real time.

Virtualization is mainly achieved with VMware enterprise solutions, due to its vendor support and easy management and configuration profile. On physical server hardware ESX hosts are installed, which are then connected to a vCenter 6.5 instance.

4.2. KYPO Czech Range

KYPO is the largest academic Cyber Range in the Czech Republic. The platform is fully cloud-based and supports multiple use cases of research and training. To validate KYPO, a Czech national cyber exercise, the Cyber Czech, is conducted on the environment [7].

Computing infrastructure includes housing facilities, physical machines, network devices, other hardware and related configuration artefacts. It forms the raw computing resources such as storage, operating memory, and processing power. The KYPO platform currently runs on top of the computing infrastructure provided by courtesy of CERIT‘s Scientific Cloud. The SC has more than 3500CPU cores and 3.5PB of storage space. It consists of four separate computing clusters. Cluster Zapat consists of 112 nodes with 1792 CPUs with the following specification for every node:

- 2x 8-core Intel E5-2670 2.6GHz,
- 128 GB RAM,
- 2x 600 GB 15k,
- 1x Infiniband 40 Gbit/s, 2x Ethernet 1 Gbit/s. [8]

Cluster Zigur, comprises of 32 nodes with the total of 252 CPUs and with the following specification for every node:

- 2x 4-core Intel E5-2643 3.3GHz,
- 128 GB RAM,
- 2x 600 GB 15k,
- 1x Infiniband 40 Gbit/s, 2x Ethernet 1 Gbit/s [8].

The Zevura cluster contains 8 servers HP ProLiant DL980 G7, each of them in configuration

- 8 Intel Xeon E7-2860 processors (10 CPU cores, 2.26 GHz),
- 512 GB RAM,
- 20x 900GB hard drives to store temporary data (/scratch), configured in RAID-10, thus having 8 TB capacity [8].

The Zegox cluster consists of 48 nodes with the total of 576 CPUs with the following specification for every node:

- 2x Intel E5-2620 (6 cores),
- 96 GB RAM,
- 2x 600GB hard drives [8].

The resources for the KYPO range are allocated by the SC case-by-case basis. Allocated resources are isolated testbeds for conducting tests and exercises.

The OpenNebula platform allows for the management of heterogeneous computing resources (usually virtualized) in order to implement the Infrastructure as a Service model. OpenNebula is managed by CERIT-SC and it is transparently used by the upper layers of KYPO to provide and create virtual machines and configure networking. The monitoring API is responsible for monitoring management, i.e. it provides fine-grained control over network links and hosts monitoring configuration (starting, stopping, and attributes manipulation). The essential component of the exercise is the simulated network, which serves as a virtual battlefield. Figure 2 shows the logical topology of the network [7].

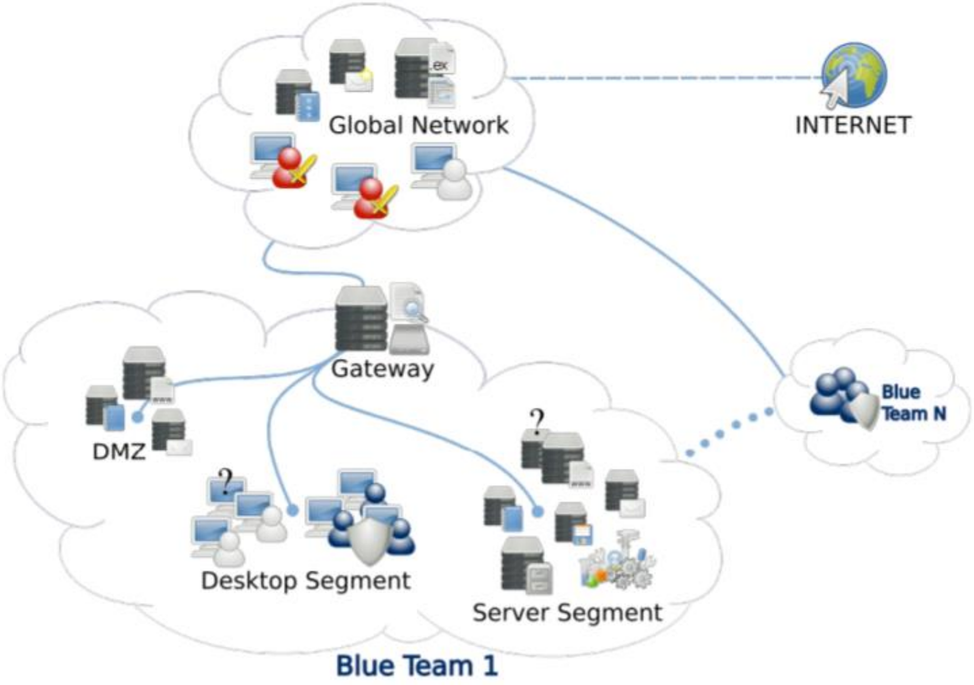


Figure 2. KYPO Range Blue team segment [7].

The exercise network is segmented into two main types of subnetworks:

- The global network – hosting attackers and common network infrastructure, such as DNS and email; this network simulates the global Internet.
- The network of a Blue team – representing the defended network with all critical (and vulnerable) services; this subnetwork is further segmented into a demilitarized zone, desktops and servers.

4.3. Considerations and technologies for federation

To start federation discussions between nations, the statuses of CR-s need to be considered, i.e. entities who operate the CR in each country. Different rules for usage apply for military, civilian agency or a private subject/academia. Cooperation between military and private bodies is always complicated and certain restraints in interconnection and cooperation is needed. Besides technical challenges and requirements, there is also legislative, political and procedural level in such a cooperation. Every state has different *modus operandi*, ranges are operated by different bodies, thus a challenge is created. For some entities, the information shared can be sensitive, creating an unwillingness for information exchange. To overcome this, certain level of legal and political processes need to be set. For arrangements of terms and conditions for multinational usage of cyber ranges, an ad-hoc working group has been brought to life in the European Defence Agency.

4.3.1. VPN technologies

There are multiple options for federation: layer 1 physical interconnection, layer 2 and layer 3 logical federation. Physical interconnection of ranges into a federation is considered most beneficial but challenging and too costly only for support of (multi)national and complex exercises. Layer 2 and 3 interconnections are cheaper to achieve and will potentially grow to an ad-hoc federation of physically interconnected national cyber ranges exchanging real-time data.

Layer 2 virtual private network emulates the behaviour of a LAN across an L2 switched, IP or MPLS-enabled IP network, allowing Ethernet devices to communicate with each other as they would, when connected to a common LAN segment. Point-to-point L2 connections are vital when creating L2VPNs [9, pp LSC-104 - LSC-116].

Building a layer 2 VPN assumes cooperation between the ISP and the customer. ISP provides the layer 2 connectivity. Which after the customer can build a network using data links received from the ISP. The ISP does not require information about the customer's network topology, policies, routing information, point-to-point links, or network point-to-point links. Possible variants and technologies enabling Layer 2 VPNs:

- ATM over MPLS
- Ethernet over MPLS
- Any Transport over MPLS
- Frame Relay over MPLS

To achieve L2 VPN services, some investments need to be made from the customer side of procuring new networking hardware, so that MPLS connections and pseudo-wire connections between service provider and CPE can be initiated. Sending IP networking information through Layer 2, customer deployments require a solution to support AToM with disparate transport at network ends. This solution must have the capability to translate transport on one customer edge device to another transport, for example, Frame relay to Ethernet [9, pp LSC-104 - LSC-116]. Figure 3 shows one possible solution of service provider managed Layer 2 site-to-site setup.

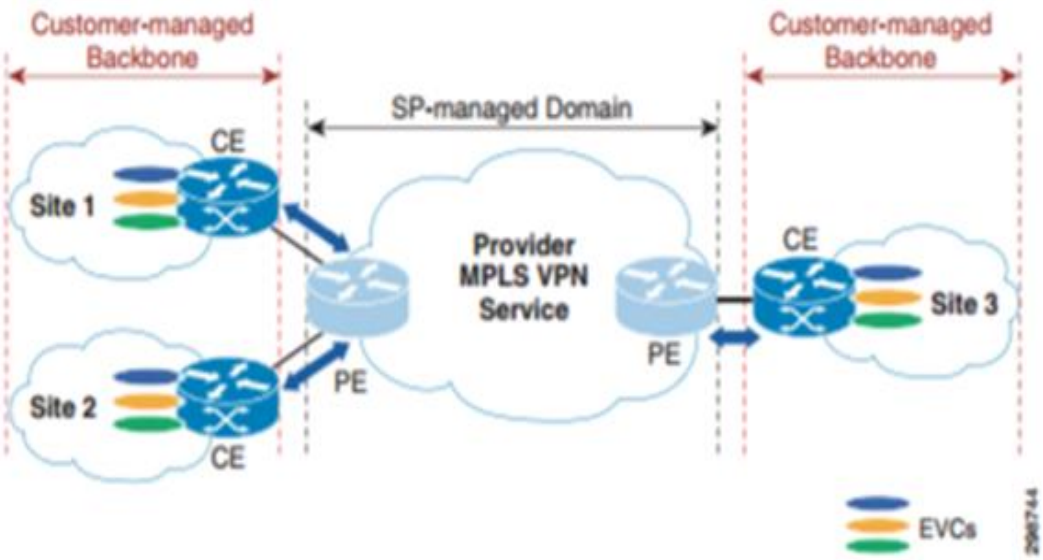

Customer-managed Backbone
Customer-managed Backbone
SP-managed Domain
CE
Site 1
Provider MPLS VPN Service
CE
Site 3
PE
PE
Site 2
CE
EVCs
298744

Figure 3. SP-managed IP VPN Service [10]

Although Layer 2 VPN connects ranges in one big federation environment, some additional control mechanisms for monitoring and managing ranges need to be in place. While planning this kind of connection, an input from every range architecture is needed. There needs to be one common system architect and engineers who will agree upon different IP addressing and VLAN schemes, environment isolations and implementation of monitoring and situational awareness tools.

For the purpose of this thesis, Layer 3 VPN connection is discussed and implemented in the test environment.

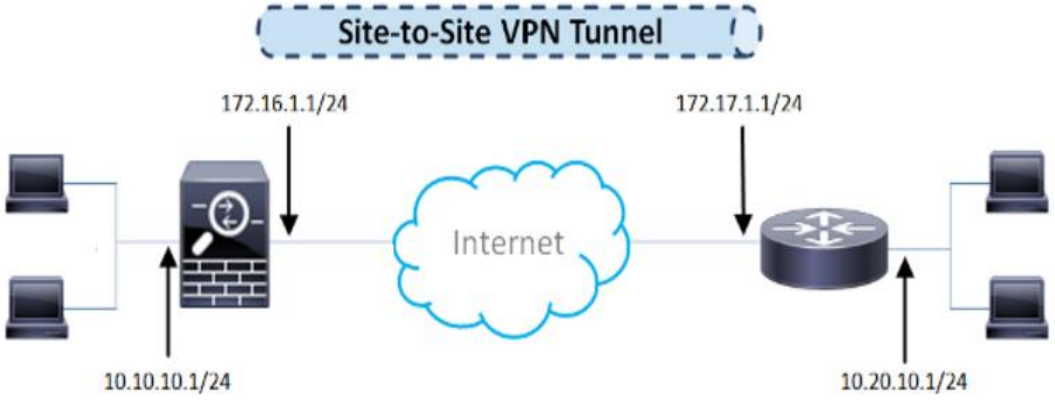


Figure 4. Layer 3 VPN tunnel simplified.

IPSec VPN is a security feature that allow users to create secure communication link (also called VPN Tunnel) between two different networks located at different geographic locations. Figure 4 describes how Layer3 VPN works conceptually. Dedicated Cisco IOS routers or Linux based virtual machines can be used to setup VPN tunnels between two sites. Traffic like data, voice, video, etc. can be securely transmitted through the VPN tunnel [11].

Considerations for using IPSec VPN tunnels.

Advantages:

- Requirement of buying dedicated expensive lease lines from one site to another is eliminated, as public telecommunication lines are used to transmit data.
- The internal IP addresses of both the participating networks and nodes remain hidden from each other and from the external users.
- The entire communication between the source and destination sites remains encrypted, which means that chances of information theft are extremely low [12].

Disadvantages:

- IPSec enabled router or appliance is required at each site to play the role of the VPN server.
- Since encapsulation, decapsulation, encryption and decryption takes place at the routers, these devices may face processing overhead and increased CPU utilization. Because of this, users may experience reduced communication speed.
- The configuration process of IPsec VPN site-to-site is complex and the configurations need to be well documented and agreed upon [12].

To start the IPSec VPN negotiation between two sites, certain networking requirements need to be met:

- Minimum of 384Kbit/s bandwidth between two sites [13].
- At least one publically routable static IP address at the main site. The address should be exposed directly to the Internet.
- NAT should not be used.

IPSec VPNs work best, if both ends of the tunnel have a static IP. It is not effectively possible to manage a VPN where both ends have a dynamic interface.

If one end of the tunnel is behind a NAT-ed address, it is likely, that the site-to-site connection will not work consistently or needs manual reconfigurations with every system initialization. It will also pose problems, with software solutions communication between both sites [14].

Although IPSec could be the best technology for layer 3 site-to-site, multiple alternatives for VPN protocols are also available. Possible alternatives include PPTP, SSL or TLS VPN-s.

For the first phase of federation, the mentioned site-to-site VPN technologies are ideal. They have security built in. Connections between data centres are encrypted. The technology is scalable, allowing to add new sites to the network easily. It is cheap and can be managed within a low latency/bandwidth network, what is crucial in the initial steps of achieving federation.

4.3.2. VXLAN technology

VXLAN is an important requirement for virtualized environments using a Layer 2 physical infrastructure for having the Layer 2 network scale across the entire data centre or even between data centres for efficient allocation of compute, network, and storage resources [15].

Many data centres face the problem of VLAN starvation with only 4094 different usable VLANs. VXLAN protocol allows major improvements in scalability enabling up to 16 million segments, layer 2 Spanning tree protocol and provides built in security [15].

VXLAN uses VTEP devices to map end devices to segments and to perform VXLAN encapsulation and de-encapsulation. The VTEP device uses unique IP address to encapsulate Ethernet frames and transmits the encapsulated packets to the transport network through the IP interface between stateless tunnels of two endpoints showed in figure 5 [16]. By other words, it means it is a prerequisite technology to extend Layer 2 networks across Layer 3 infrastructure making federated range environment easier to interconnect, scale and manage. This technology eliminates some technical limitations between federated parties, such as previously mentioned lack of VLANs and re-usability of IP subnets.

As VXLAN is a broad technology, thus its design and implementation for interconnecting cyber ranges is not in scope of the current thesis, this subject is briefly overviewed for future investigations.

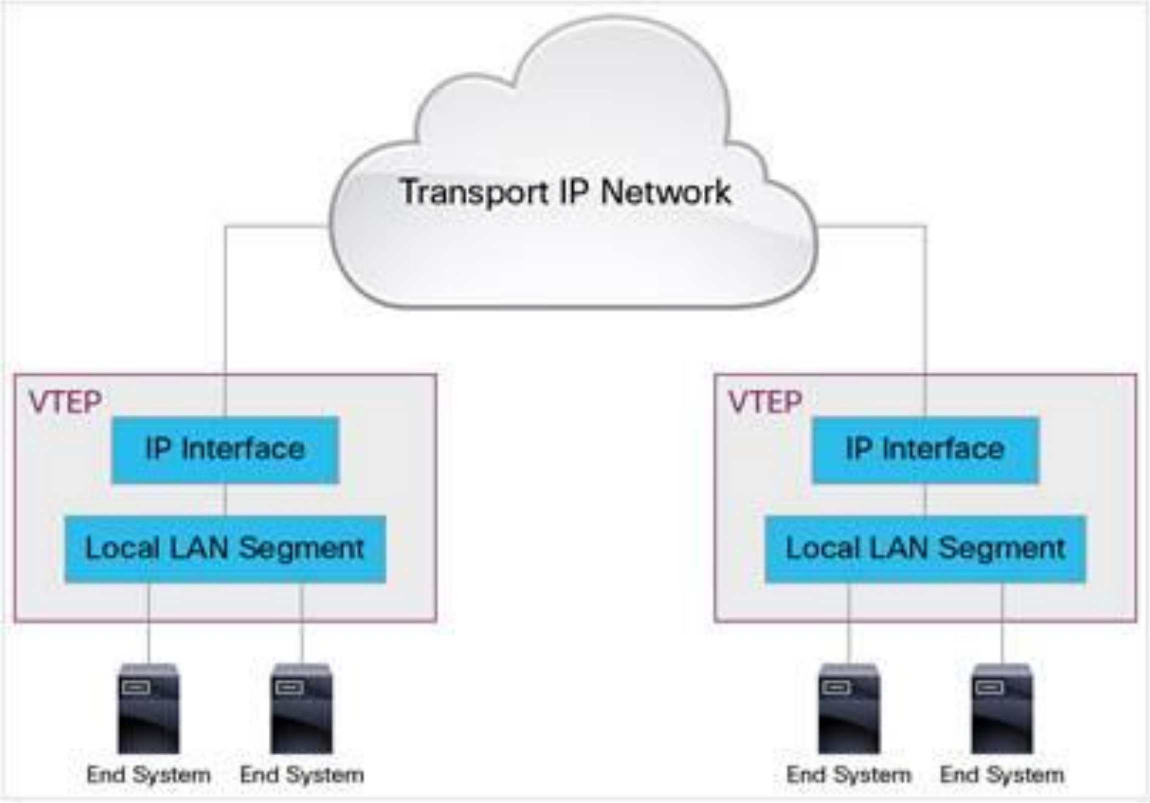


Figure 5. VXLAN VTEP connection overview [16].

4.3.3. Common IP addressing and gamenet VLAN-s

The current set ups of cyber ranges will be conflicting due to IP addresses and VLAN's used in exercise environments. Layer 3 federation can be achieved only after agreements and network design. While trying to connect environments with the same IP networks, conflicts will occur. In the VMware environment, it is not possible to add ESX hosts with duplicate IP addresses. Duplications might happen with other widely used hypervisors. Troubleshooting hypervisor networking is complicated.

Monitoring of virtual machines and their configurations is also important. While copying or cloning VM-s, some virtualization technologies also copy settings from the initial machine, creating duplicated IP-s in the environment.

A central authority needs to maintain and design the IP addressing scheme. The goal of the scheme should create a consistent structure what will simplify administration. After network design, it should be easy to identify, locate and manage different networks and equipment. The most important consideration, however, is the understanding the network structure of the federation environment. This creates a situation where some entity has all the information where all the environments are located and how they are connected together.

Because there are many cyber ranges within NATO and EU an addressing scheme from the 10.0.0.0/8 network space should be preferred. For instance, a following format for core services can be used:

10.<Country>.<VLAN ID>.X

A different networking scheme is needed for the gamenet. While planning an exercise in the federated environment, Locked Shields exercise networking schemes can be brought as an example. Blue team systems are accommodated on a 10.X.0.0/16 network, where X is the team number. Separate networks for green and red simulated internet should be used.

For site-to-site connections, good configuration practises should be used. The management VLAN needs to function as a privileged network for the purpose of troubleshooting and diagnostics. While it is also used for monitoring purposes, heavy loads of SNMP and NetFlow traffic will be present. This should be isolated and prioritized down not to interfere with the federation critical traffic.

Those are just some considerations, while planning the network for the federated environment. The network design should meet the demands of flexibility, scalability and ease of management of the federated network.

4.4.4. Access lists

To ensure isolation between different ranges and allowing exchange of services within the federated environment, Access Control Lists should be used. ACLs are allowed with both layer 2 and 3 site-to-site connections. They help to limit incoming and outgoing network traffic and restrict the access of users and devices to the network. By nature, ACLs filter traffic as it passes through networking equipment and can permit or deny packets to specified interfaces, VLANs or services. The packets are compared against a list of rules. When the first match is found, the networking device either accepts or rejects the packet. List comparison is finished after the first match, making the order of conditions critical. Considerations for Layer 2 Access Lists:

- Ethernet access lists are supported on Layer 2 interfaces only.
- Ethernet services access lists are not supported over management interfaces.

- Ethernet services access lists are not supported over routed interfaces [9, pp LSC-351 – LSC-364].

Limitations for Layer 3 Access Lists

- Up to 256 Access Control Entries, one ACE per IP Access List.
- Up to 511 ACE on a single IP Access List [17].

It is not foreseen, that ranges will utilize all the maximum values of access lists. Making it an ideal technology for service sharing or isolation within the initial federated environment.

4.4.5. Backup connections

After federation is achieved, multiple connections between sites need to be set up. To avoid connection loss to the other site, at least one parallel VPN connection to the other site needs to be in passive mode. This helps to recover from hardware failure or physical connection issues. In the future, connectivity and VPN tests need to be carried out between all connected parties. Every connection should be viewed as case-by-case.

To ensure a consisting network connection for remote desktops, vMotion and file transfers within the federated cloud, network failover detection and NIC teaming can be used. To achieve better fault tolerance within federated ranges, at least two different dedicated lines between need to be present. The problem is that usually different connections between data centres and ISPs have separate routes and IPs creating detached site-to-site connections. A solution is to create a bonded VPN tunnel, what will seemingly act as one interface. Data centres endpoint interfaces will appear as one IP address, but actually all the traffic is being split [18]. VPN tunnelling is described in figure 6.

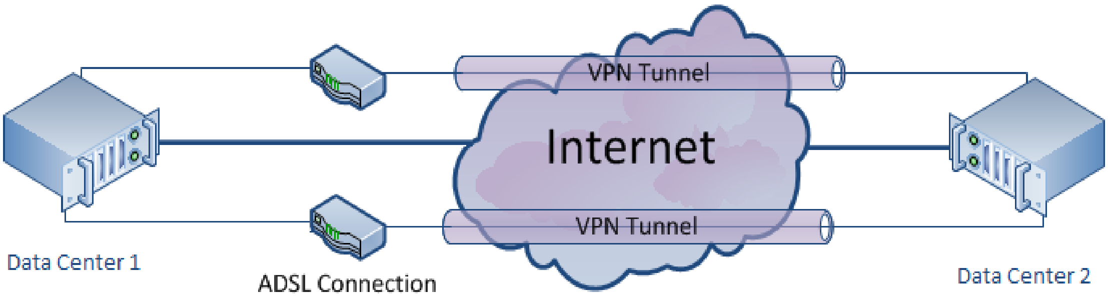
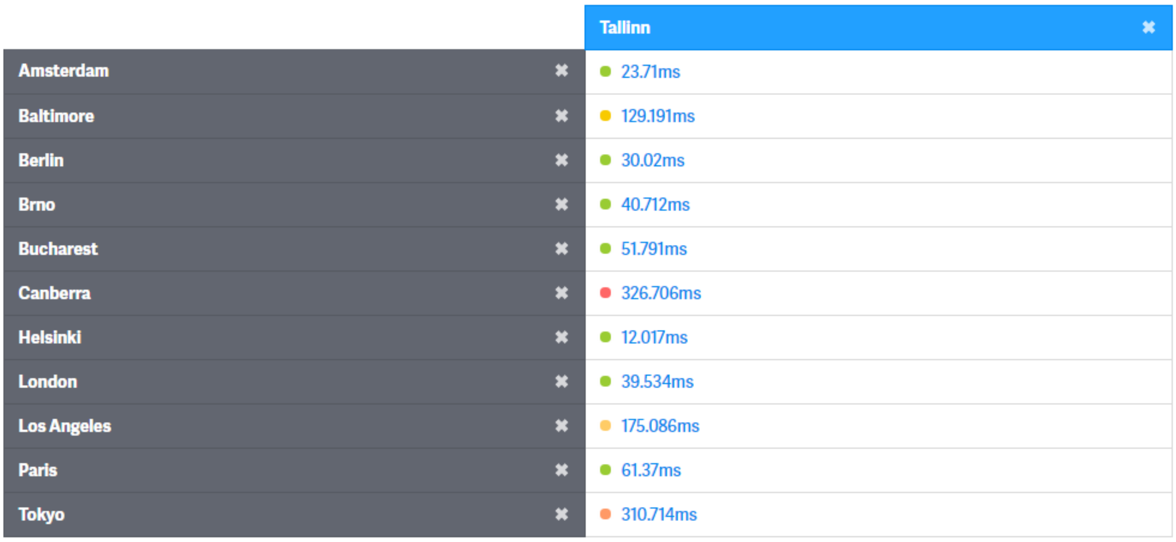


Figure 6. VPN Bonding between datacentres [18].

4.4.6. Bandwidth and latency

Network connection issues, like slow bandwidth and low latency, will be inevitable in the multinational federated cloud. For long distance vMotion, the maximum latency cannot be higher than 150ms between hosts [19]. While looking at the overall statistics of Europe and some partnering nations for NATO, a federated environment with vMotion functionality is plausible. Bandwidth values between NATO and EU partnering nations is shown in figure 7.



	Tallinn ✖
Amsterdam ✖	🟢 23.71ms
Baltimore ✖	🟡 129.191ms
Berlin ✖	🟢 30.02ms
Brno ✖	🟢 40.712ms
Bucharest ✖	🟢 51.791ms
Canberra ✖	🔴 326.706ms
Helsinki ✖	🟢 12.017ms
London ✖	🟢 39.534ms
Los Angeles ✖	🟠 175.086ms
Paris ✖	🟢 61.37ms
Tokyo ✖	🟠 310.714ms

Figure 7. Latency between Tallinn and other cities [20].

A bigger issue arises with dedicated bandwidth between sites. While a plan disaster recovery and high availability emerges in the future in the federated cloud, at least 250 Mbps of dedicated bandwidth needs to be ensured.

The main tool for conducing Cyber defence exercises and training in similar range like environments is a remote desktop interface. To guarantee a RDP session, certain bandwidth and latency requirements need to be met. For VMWare remote console a good user experience is still maintained with the latency less than 200ms. Console is unusable when latency gets more than 350ms [21]. For exercises purposes, here is a need to use not only remote consoles but also remote desktop connections and graphical interfaces. According to Microsoft, the average bandwidth for RDP 6.1 protocol with no themes, no desktop composition and with 32bit high resolution is ~50 KB/s. For different scenarios like browsing the web and using some presentation software, the number goes up to 120 KB/s [22].

4.3.7. Virtualization servers

One of the most complicated tasks of federation is to achieve communication between different virtualization platforms. The main operations like migration, cloning, making templates and snapshots needs to be available. Some platforms like VMware and KVM do not allow those processes between each other. Linux based hypervisors on the other hand are capable of communicating such operations. Additionally, over long distances, those processes might lose their practicality. There has to be a reasonable use case for doing a live migration from one data centre to a remote site. This can be a variant outside of the scope of multinational exercises, to help other ranges to facilitate scheduled maintenances and disaster recovery.

While discussing connections of datacentres over long distances, VMware has introduced some new features with the last updates of 5.5, 6.0 and 6.5 [23]. It is possible to connect multiple vCenter servers through single Single-Sign On service and long distance vMotion technology with upgraded minimum requirements for virtual machine migration.

To connect two vCenter instances, linked mode technology needs to be used. The prerequisite for achieving the desired state is that all vCenters need to be registered in a Linked Mode group on the same SSO server shown in figure 8 [24].

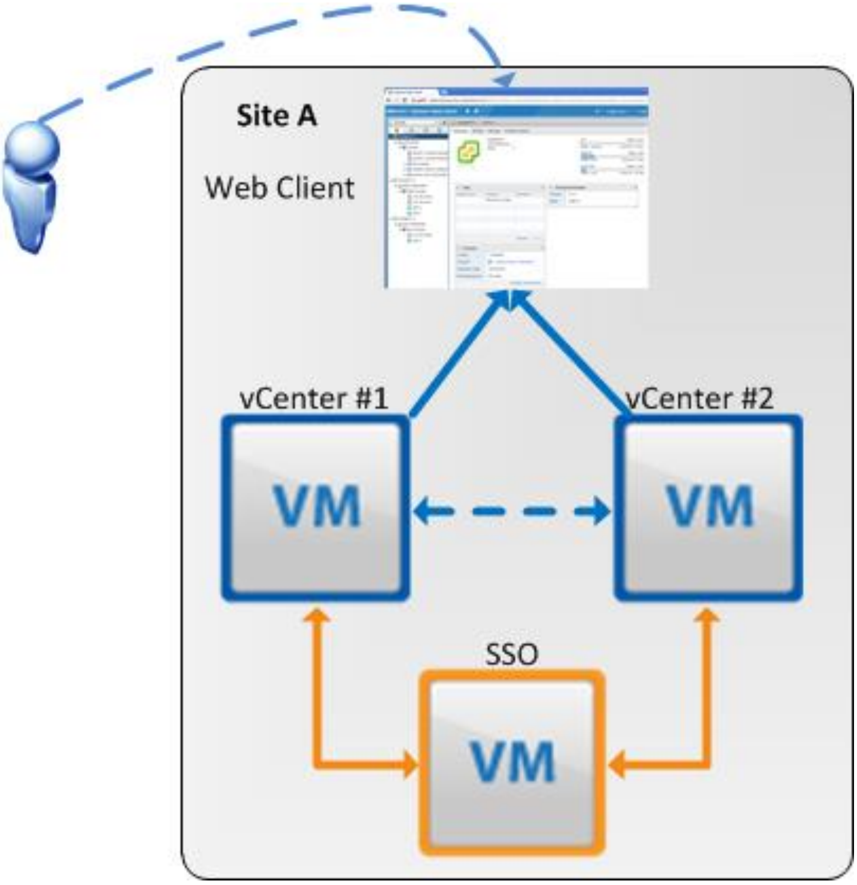


Figure 8. SSO concept [24].

While using this solution, separate data centres might endanger the whole federated cloud environment. Definitely, there will be a latency issue between two sites and the SSO service. Extra effort is also needed from the administrators, because automatic replication of database between SSO sites is not supported. Whenever a change is made to one of the instances, a manual data export/import operation must be performed with a command line tool [25].

Multisite Single-On deployment is designed only for fast access and authentication but does not guarantee a failover mechanism between sites. When SSO on one site fails, then its role is not automatically taken over by the other site. This means that High Availability is not configured [24].

While multisite SSO might be acceptable for smaller enterprises, then in the federated multinational environment, a failure like this might produce a complicated situation, where no one can access the vCenter instance. Next do consider about multi-site SSO, is database availability and synchronisation across different nations. Database and SSO synchronization failure is depicted on figure 9. With the current version of vSphere, database clustering is not supported creating a new single point of failure [24].

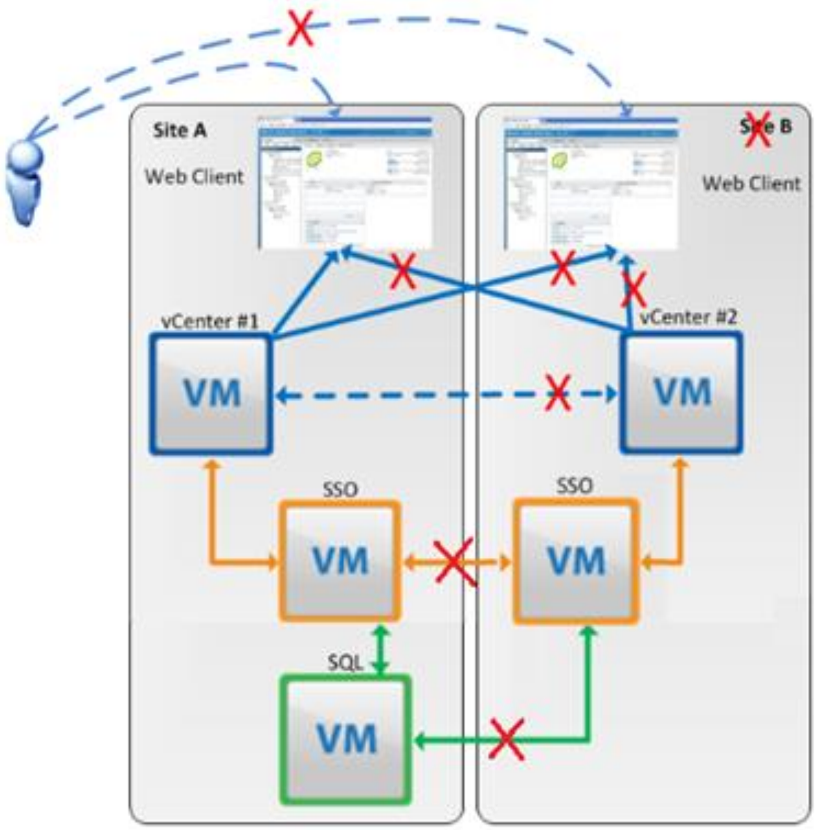


Figure 9. SSO Failure [24].

To make the environment failsafe, a good practice recommends do use separate management interfaces for different sites. This ensures that a failure on one site will not interfere with another's site availability [24].

A more suitable variant for federation would be usage of a remote ESXi instance and connect it to a vCenter server. This solves the problem of responsibility and maintenance of every part connected to the federated cloud. In this case, a central vCenter instance can be configured and set up by one nation. All the other nations will connect to this instance with their ESX servers or resource pools [24].

For the Locked Shields use case, it means that all the core systems of green, red, yellow and white are in the central vCenter and the blue teams can have their own dedicated servers and ESX hosts at their home nations. This solves some latency issues for the teams participating geographically further from Estonia. With this, extra layer of HA exercise environment is added. There is the chance to do cold migrations from different nations to the centric vCenter instance providing additional failsafe during an exercise if needed.

In this type of decentralized environment, new issues need to be considered. While every nation could bring their own computing equipment for the exercise, it creates an uneven competition. Some nations might have better and newer equipment. While exercises get more de-centralized, the tasks of the green and red team might will be more complicated. If something happens to the remote blue team system, the reaction time and fixes to issues will take more time.

Multiple vCenter and remote ESX host solutions tackle the problem of a common tool to manage and monitor the environment. It is trickier, if some other virtualization platforms are used. Even though, there are solutions available that can manage and monitor different platforms. A specially developed tool could be used, but for purposes of this thesis an open source tool, OpenNebula, will be used. OpenNebula supports VMware's vCenter, KVM and XENserver and is easy to set up by everyone.

5. Test environment

To get the maximum out of the thesis a test environment between the Estonian Cyber range and the Czech KYPO lab is set up with multiple services and virtual machines. This environment is small scale and its main purpose is aiding of the proof-of-concept tests. This environment helps to understand the issues what might occur during the actual federation. For the thesis to help to understand what will happen in the federated environment, only VMware type hypervisors are used.

5.1. Description of the test environment

Test environment uses enterprise licenced ESX 6.5 hosts and vCenter 6.5 instance with two ESX hosts on the Estonian side and one ESX host on the Czech side. Hosts are configured with vMotion and two network interfaces.

For testing, the EDF Cyber Range provides two ESX host with eight Intel Xeon X5650 cores and 24GB of RAM. KYPO Range provides one host with four Intel E5-2643 cores and 16GB of RAM.

A virtual Linux based router from the EST side serves as the VPN concentrator with FTP services alongside ESX. Bandwidth and latency manipulations are made in the Estonian router. Virtual router connects to Estonian cyber range ASA, where backup links are maintained.

On the other side, a Debian based virtual router receives and forwards the VPN Layer 3 traffic. A SSL VPN tunnelling protocol is used between sites. VPN configurations reside on this router.

Configuration of the routers can be found in Appendix I. On both sides, some test virtual machines are created.

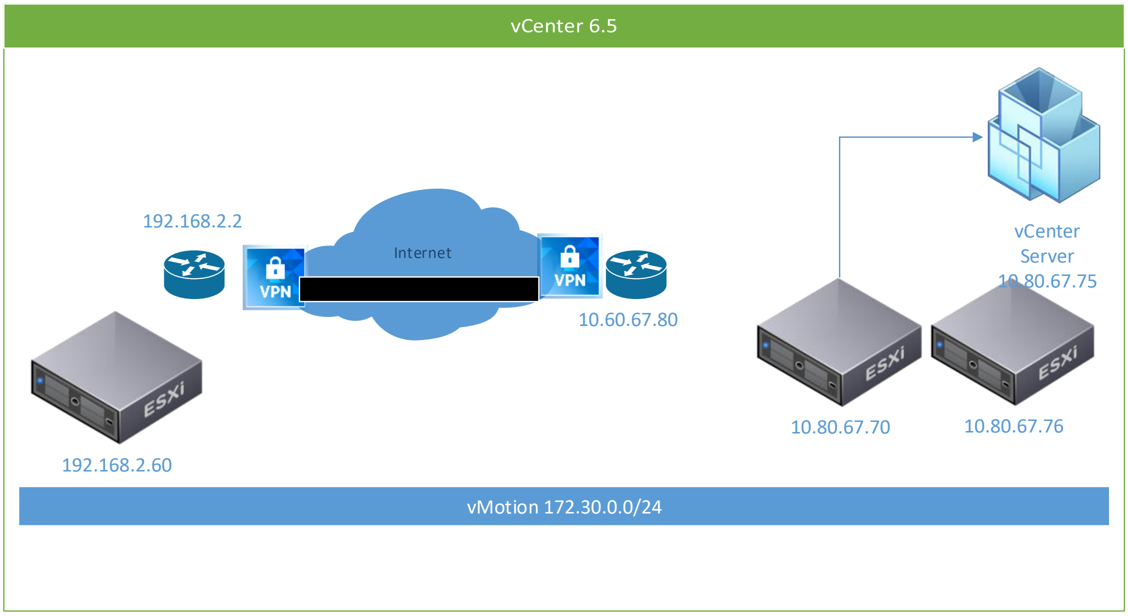


Figure 10. High-level overview of the test environment.

Different private networks are used within the range environment. Estonia uses the 10.80.67.0 network and 192.168.2.0 network is used in the KYPO lab. vMotion between sites and hosts are configured to use 172.30.0.0 network. The high-level overview of the test environment is shown on figure 10.

Test machines with the following characteristics are created in the environment:

Ubuntu_TestMachine (16.04 Ubuntu):

- 1 CPU (32-bit)
- 1024MB RAM
- 16 GB Hard Disk
- Network Adapter

Win7_TestMachine (Windows 7)

- 1 CPU (32-bit)
- 2048MB RAM
- 24 GB Hard Disk
- Network Adapter

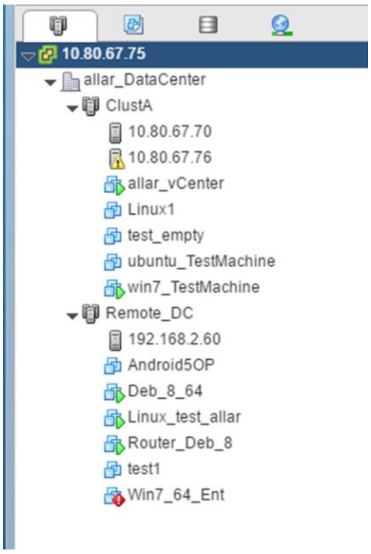
Test_empty (Linux Ubuntu profile)

- 1 CPU (32bit)
- 1024 MB RAM
- 512MB Hard Disk

Additionally, Linux_Test_Allar on the Czech side is created (Linux Ubuntu 16.04 Server 64-bit) and is mainly used for monitoring the connection and latencies between two sites.

Ubuntu and Win7 Test machines will simulate processes within Cyber Range exercise environments, while the empty machine is for just testing if vMotion and other technologies work in the test environment.

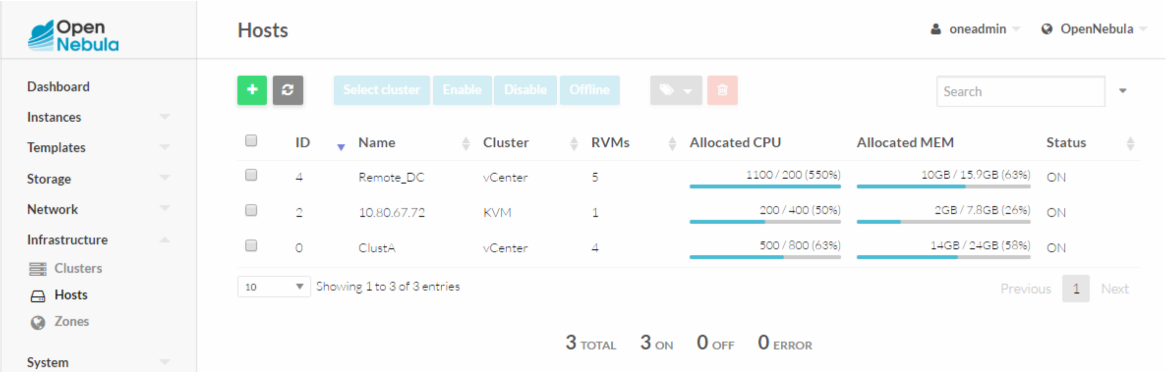
The average latency to the KYPO range is measured ~46 milliseconds and average bandwidth is ~1,1 MB/s. Those figures should support vMotion migrations and RDP sessions. It also gives room for some speed and latency manipulations, to get a better understanding what are the total minimum requirements for federation. Current connection parameters are sufficient to operate the vCenter instance with the graphical web interface shown on figure 11:



- 10.80.67.75
  - allar_DataCenter
    - ClustA
      - 10.80.67.70
      - 10.80.67.76
      - allar_vCenter
      - Linux1
      - test_empty
      - ubuntu_TestMachine
      - win7_TestMachine
    - Remote_DC
      - 192.168.2.60
      - Android5OP
      - Deb_8_64
      - Linux_test_allar
      - Router_Deb_8
      - test1
      - Win7_64_Ent

Figure 11. vCenter view of the test environment

For testing and monitoring purposes of different hypervisors, an OpenNebula instance is set up as show on figure 12. All the vCenter nodes and virtual machines are accessible from it. A KVM based system is attached to the OpenNebula instance to simulate a wider variety of different hypervisor technologies. The OpenNebula instance itself is run on a Linux Ubuntu server virtual machine on the Estonian Cyber Range.



□	ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status
□	4	Remote_DC	vCenter	5	1100 / 200 (550%)	10GB / 15.9GB (63%)	ON
□	2	10.80.67.72	KVM	1	200 / 400 (50%)	2GB / 7.8GB (26%)	ON
□	0	ClustA	vCenter	4	500 / 800 (63%)	14GB / 24GB (58%)	ON

10 ▾ Showing 1 to 3 of 3 entries

Previous 1 Next

3 TOTAL 3 ON 0 OFF 0 ERROR

Figure 12. OpenNebula instance overview

5.2. Test plans

After initial connection is set up, tests designed are to help to understand, how federation works and what future problems might arise. Different use cases will test the environment multiple times and results help to point out bottlenecks and future considerations. Next chapter includes all the outcomes of described tests.

5.2.1. Testing backup links

First step is to simulate a regular interface failure without load between sites. This can be achieved by disabling the main ASA interface connected to the remote site. Average switching time between active-passive interfaces will be the basis for further tests.

The same test needs to be carried out with a small load. For this purpose, a test file (testfile.iso) is uploaded to the Estonian FTP and TFTP server. FTP used the TCP protocol providing a reliable, ordered and error-checked delivery. While TFTP uses a UDP protocol what can work in an environment where there is no guarantee of delivery. (T)FTP file download from the CZ side is initiated, while the main interface is disabled and the backup link is switched on.

A similar test is carried out during a migration of an empty, Linux and Windows test machine.

5.2.2. Virtual machine migration from site-to-site

To see, how the resources from different ranges can be shared in the federated cloud, a test for moving virtual machines from site to site is constructed. The prerequisite for achieving this comes from the technology, where migrated virtual machines need to use the same hypervisor technology. For moving VMs, different bandwidth and latency values will be used. Manipulations will be implemented in the Estonian virtual router. For latency manipulations, the following command on the tunnel interface will be used:

```
tc qdisc add dev <interface> root netem delay XXms
```

Where XX defines the latency forced on the interface.

For bandwidth manipulation, the wondershaper tool in Linux is used. Syntax for bandwidth c <interface> <dl> <ul>

Where <interface> is the interface what will be limited, <dl> is the download limit and <ul> is the upload limit. Speed limits are specified in kilobits per second.

5.3.3. Creating a virtual machine to a remote site

This test determines the usability and applicability of creating a virtual machine on a remote host. For testing purposes, a Linux Ubuntu 16.04 server instance is installed with the following settings:

- 1GHz CPU
- 512 MB RAM
- 3GB Hard Disk

The NICs and installation will be done from the Estonian side. Installation ISO file is accessed through the local datastore from the Czech side. After booting the created Virtual Machine up from the *.iso file default settings for installation are used.

5.3.4. Testing OpenNebula instance

The aim of this test is to see whether OpenNebula is an applicable tool for basic management operations. Decision of using the current open source tool comes from the fact that the KYPO range uses it for their everyday operation. This raises a question if the tool could be the management tool for the federation environment. To get an answer do the previous question some basic operations with virtual machine processes will be initiated. Processes what will be tested on both vCenter and KVM instances are:

a. Power On
b. Power Off
c. Suspend
d. Create a snapshot
e. Reboot
f. Migrate
g. Open remote console/desktop

Discussions on the user interface and the logic of the environment evaluation will follow operational tests.

5.3.5. Bandwidth and latency tests between remote sites

Bandwidth manipulations help to understand, how available bandwidth affects the usage of main exercise tools: remote consoles and desktops. Firstly, the vCenter latency maximums are tested. Next, the baselines of remote console and desktops bandwidth usage is observed. For remote desktops, different use cases like web and directory browsing is tested. Same tests can be carried out with latency manipulations.

To get the maximum out of the test, some manipulations with extreme bandwidth and latency will follow to see how vCenter and virtual machines remote desktop/console connections act under extreme circumstances. Tests with those values help to understand the applicability of the tunnelling and datacentre technologies in military systems in case of emergencies and conflicts. Military systems often use legacy technologies and protocols, making management and sharing information crucial, for instance over satellite links with high round-trip times and low bandwidths.

6. Test results

Current test results are observed between the site-to-site SSL connection between the Estonian and Czech ranges. The results may be wary between different interconnections and are influenced by tunnel negotiation and paths provided by ISPs. Similar tests need to be carried out between all the member states within the federated range environment.

Current tests show that federation is achievable between the two sites. Minimum requirements for future federations can be defined based on current results. Tests carried out also point out some potential bottlenecks, what need to be considered. The biggest prerequisite for federation is the network. To work in a multinational environment with different functionalities of site-to-site federation, high bandwidth and low latency connections need to be available for VPN technologies to assure high availability and fault tolerance of shared services.

Depending on a ranges available work force, enterprise grade software and equipment helps to achieve federation with lesser effort and administration. This kind of equipment might provide some proprietary technologies what cannot be achieved with open source tools. On the other hand, free tools allow more flexibility and customization in every environment.

6.1. Backup links

For the first step, a regular interface failure with no load between sites was simulated. After shutting down the main routers interface, the interface in the passive state took over the connection between sites. To get the proper statistical baseline, this test was carried out ten times with. The number of the current failure simulation is enough because here were no big deviations within the observations. On average, a small packet loss of two ICMP packets was observed.

Next, the same test was conducted under a small load. A FTP connection was initiated from the remote site. During the FTP file transfer, the main router interface was disconnected and the backup router became active. FTP file transfer got stuck, thereafter it failed during the switch to the backup link. FTP uses a TCP connection, and by design, packets sent over this type of connection need to be in a certain order and the success of the file transfer depends on the connection. If any data is lost between the transit, the sender will retransmit the data. Seems, it is not the case while changing to the backup tunnel interface.

The same test was carried out with an UDP file transfer over TFTP. By design, UDP should not care if some data is lost. From the remote site, a TFTP connection is initiated and a test file download is started. On the local site, the main tunnel interface is unplugged and the backup interface becomes active. The UDP file transfer fails with a timeout error.

Next, the same procedure for activating backup link is tested while using vCenter and virtual machine migration. Firstly, the empty virtual machine is migrated from the Estonian side to the Czech side and the main tunnel interface is switched off. Migration fails. Similar results with the Linux and Windows test machines.

While running the federated environment, another alternative for a backup connection is needed. There will be no problems with the vCenter and management instances. Management is not influenced by switching to the backup link.

The current set up indicates that building a federated environment with just active/passive routers has no fault tolerance. If the main connection is switched to the backup, all the connections need to be reset and ongoing operations with file transfers and virtual machine migrations will be aborted. In the future, tests with VPN bonding or other technologies need to be tested, a better and more fault tolerant variant for federation is needed.

6.2. Virtual machine migration from site-to-site

Site-to-site migration of a virtual machine needs a good use case. Migrating a virtual machine with a big hard drive and information inside is heavily affected by bandwidth. For testing purposes and future reference, tests with VM migration between geographically different datacenters was conducted.

Because of the slow bandwidth between the two data centres, VM migration takes lots of time. Tests were carried out during the passive times to get the most accurate baseline information. Empty test machine was migrated ten times and the other machines were migrated five times.

In the test environment where a 1,1MB/s guaranteed bandwidth between sites was available, an empty machine was migrated by average with 10 seconds, the 16 GB Ubuntu_Test-Machine machine was migrated with 60±5 minutes and the 24GB win7_TestMachine was migrated with 100±7 minutes. It is important to note that all test machines disks are thin provisioned. For thick provisioning, the times for migration mathematically would be 230 minutes for the Linux and 375 minutes for the windows machine. Mathematical formula to get the time in minutes is following: $minutes = \frac{Hard\ Disk\ Size\ (MB)}{60 * Bandwidth\ \left(\frac{MB}{s}\right)}$. Migration was tested with half the bandwidth and the empty machine was migrated with 13 seconds by average. Under those conditions the Linux machine took 78 minutes and windows machine 130 minutes.

For further bandwidth manipulation, only the empty test machine was moved between the data centres. Other test machine migrations were only started, but cancelled by the user. This was only done to to see whether the migration process for test machines with bigger hard disks starts. Test machine migration was successful still with 5KB/s bandwidth between sites taking ~60 seconds by average. Migration for other machines was cancelled by the system at the bandwidth of 20KB/s.

For latency manipulation, test Windows and Linux machine migrated with the latency less than 350ms. Taking ~80 minutes for Linux and 115 minutes for the Windows machine. Empty test machine migrations failed with the latency more than 2000ms. Migration time for the near failure latency for the empty machine took by average ~50 seconds. For latency and bandwidth manipulation summarization, refer to the bandwidth and latency tests and table 2 in the same chapter.

For using migration in the federated environment a bigger bandwidth allocation and certain latency requirements need to be guaranteed. In the future, tests with higher bandwidth rates need to be carried out and analysed whether long distance vMotion is applicable to use in a federated range environment.

6.3. Creating virtual machine to the remote site

Creating a virtual machine on the remote site is not affected by the fact that two data centres are in different geographic locations. The tasks for giving inputs through the graphical interfaces do not need high bandwidth and it is not affected by latency. Virtual machines can be managed with high latency and low bandwidth, meaning that it is possible to create a VM to a remote site. Using the UI, following parameters for a virtual machine were inserted:

- 1GHz CPU
- 512 MB RAM
- 3GB Hard Disk

Creation of the following test machine was successful. The difficult part started with the actual installation and with opening the remote desktop. The first obstacle was the needed *.iso file. It needs to reside on the remote sites datastore. After it was uploaded to the remote server side and the virtual CD drive was attached, the installation did not take any longer than setting it up in a local server.

By design, vCenter 6.5 has only a thin web client and while accessing the remote desktop console, it does not provide any means of connecting a virtual drive. The drive needs to be connected through the settings menu or manually from the ESX thick client.

6.4. OpenNebula management tests

To understand how OpenNebula works and if it can be used for management, basic operations with virtual machines were tested. Management possibilities with different hypervisor platforms were tested. Firstly, tests with VMware enterprise solution was tested with OpenNebula.

vCenter:

- Power off virtual machine: success.
- Cold migration: is not supported for imported virtual machines.
- Suspend: success.
- Scheduled action: success (start/shutdown).
- Take/revert to snapshot: can be seen also in vCenter.
- Changing virtual machine configurations through OpenNebula: currently not supported for vCenter [26].
- Open remote console: no remote console option by default.

Using OpenNebula with vCenter needs some time to getting used to. After every basic operation initiation, a refresh is needed. By default, some operations are not supported with imported virtual machines. There is an option available to create virtual machines through the OpenNebula instance to the vCenter server. In the use case of federation, OpenNebula should only be used for basic management operations and virtual machine monitoring. For environment management through the OpenNebula instance, a dedicated person for backend administration is needed.

After getting used to the logic, basic operation management is quite intuitive. Most of the functions available by default worked. Doing virtual machine power manipulation is easy and all the commands sent to the virtual machine can be seen in the vCenter event list. Taking and managing snapshots is also easy. Unfortunately, it worked only one way. After snapshot creation in OpenNebula, a new snapshot also appeared in vCenter. It is not true the other way round. This created a problem, after deleting an OpenNebula snapshot from the VMware snapshot list. The snapshot did not disappear from the original instance and reverting to the mentioned snapshot did not work anymore.

Scheduled actions worked good in OpenNebula and are easy to create. Nevertheless, it transpired that the two instances do not communicate with each other. Using two different management tools poses a risk for administrators. The full overview of tasks, and operations is not clear and one side might interfere with the others side processes.

The biggest drawback for OpenNebula is the absence of a remote desktop/console tool by default. To get a remote desktop tool working for the vCenter virtual machines in the OpenNebula instance, configurations from both server sides are needed.

To get a better understanding of the possibilities of OpenNebula, the same operations are tested on an open source Linux-based hypervisor.

KVM:

- Power off function is not supported for imported VMs.
- Power on does not work.
- Migration is not supported for imported VMs.
- Reboot - success.
- Scheduled tasks – success.
- Taking/reverting to snapshots.
- Changing virtual machine configuration – success.

Basic operations like rebooting, scheduled tasks and snapshots worked well. Also a really ease to use configuration wizard can be used to change the virtual machine setting.

On the other hand, some strange things happened with the most important operations. By default, power off operation is disabled by the KVM hypervisor. In addition, some complications rose when trying turning on the machine. OpenNebula instance did not understand if the machine was working or not when the command input came from the hypervisor side.

OpenNebula as a tool is actually quite capable of understanding different hypervisor technologies and has a logical structure to manage everything. Initial set up can be done in roughly 15 minutes. However, for adding all the hypervisors, datastores and clusters, it takes lots of configuration in the back-end side. Those operations, at least with vCenter products, cannot be configured from the front-end. While configuring many parameter through the back-end, mistakes can happen easily. Especially in the virtualization daemon part. Because it is an open source tool, lot of effort and time is needed to get every script and add-on running. If to consider using a tool like such, there needs to be a separate administrator who can manage the environment and communicate the problems with all the partners in the federated range.

6.5. Bandwidth and latency tests

Designed tests were carried out in passive timeframes, to get the most accurate baseline readings for the current connection. No additional traffic interfered with current tests.

Surprisingly, long distance vMotion still worked with a higher latency noted in the official documentation. VMware states, that long distance vMotion needs latency less than 150ms. Migrations for the test virtual machines were successful even with the 300ms latency. To stretch the limits and to see where the vMotion technology fails some high latency values were used. Virtual machines with bigger hard drives failed on 350ms. The empty virtual machines migration failed at 2000ms. At this value, the remote site was still connected through the vCenter and was still accessible. Remote site connection failed at 12000ms. Managing the other data centre within the high latency environment is slow. The initial signs for usability and management regression started at 2000ms, where browsing the datastores and sending commands took notably more time.

vCenter with remote site hypervisors works well also under low bandwidth networks. Bandwidth within the tunnels Estonian side interface was manipulated. The bandwidth was turned to minimum (1KB/s). vCenter and remote site browsing was still working. Virtual machine migration with the empty Linux machine failed at 4KB/s. Surprisingly, the tunnel was still working with such a low bandwidth. Author tested, if the tunnel will be negotiated if to stop the active and start a new tunnel instance with the same parameters. The tunnel interface was negotiated and tunnel between two data centres became active.

vCenter and vMotion usability, minimum and maximum latency and bandwidth values between remote sites are summarized in the table 1 and table 2:

Table 1. vCenter usability values.

vCenter usability	**Latency (ms)**	**bandwidth (KB/s)**
Good	**<150**	**>512**
Usable	**500**	**256**
Slow	**2000**	**128**
Not usable	**>5000**	**1**
Failed	**12000**	**x**

Table 2. vMotion tests.

	Latency (ms)		**Bandwidth (KB/s)**	
vMotion	**Completed**	**Failed**	**Completed**	**Failed**
Test_Empty	**<2000**	**2000**	**>5**	**4**
Ubuntu_Test_Machine	**<350**	**350**	**N/A**	**N/A**
win7_Test_Machine	**<350**	**350**	**N/A**	**N/A**

Next, tests for remote desktops and consoles were carried out. By average, a remote desktop connection without any user inputs between different data centres consumes for windows systems ~60KB/s and ~70KB/s for Linux desktop. Browsing through directories by average consumes 70 – 100KB/s of bandwidth and scrolling within directories uses ~250KB/s for Linux machines. A windows machine consumes 80-120KB/s for subdirectory browsing and up to 270KB/s for heavy scrolling in the directory. For web browsing (postimees.ee), both the Linux and windows machine consume ~500KB/s. Default remote desktop bandwidth consumption figures for both operating systems is presented in table 3.

Table 3. Remote desktop connection bandwidth consumption in KB/s.

Bandwidth (KB/s)	Linux Debian	Windows 7
Remote desktop console	60	70
Directory browsing	100	120
Directory browsing (scrolling)	250	270
Web browsing (news site)	500	500

With this kind of connection parameters, it is mathematically possible to simultaneously initiate 20 Linux machine remote connections and 17 windows machine remote desktop connections. For working with the virtual machines, simultaneously two remote desktop connections with web browsing can be handled. For remote consoles through vCenter, the maximum bandwidth consumption for Linux machines is 20KB/s. SSH console for the same machine consumes 5KB/s. Using windows machine's command prompt the maximum bandwidth through vCenter console is 35KB/s and for the SSH connection, the bandwidth consumption between remote sites is 5KB/s.

Next, the usability of remote desktop connections was observed with higher latencies. As for the connection provided between two sites by default, remote desktop was usage was good. Remote desktop usability is good with the latency less than 125ms. Remote desktop is not usable if the latency gets more than 400ms and the remote desktop connection fails at 5000ms. For remote consoles, 400ms produces still a good user experience. Under 2000ms of latency, the usability is poor and after that, the console is not usable anymore. The usability of remote desktops and consoles is summarized in table 4.

Table 4. Remote desktop and console usability with different latency values.

Latency (ms)	Remote desktop	Remote console
Good	<125	<200
Usable	<200	<400
Slow	<400	<2000
Not usable	>400	>2000
Failed	5000	N/a

7. Conclusion

The technologies discussed in the fourth chapter provided a good basis for achieving initial connection with the KYPO Cyber Range. The chosen VPN tunnelling solution can be implemented parsimoniously for future interconnections between new ranges in the federated environment. The fusion with emerging technologies and currently conducted tests, should initiate the future implementation designs for federation. Forthcoming plans can consider the test environment as a template or suggest a better alternative for every specific range connection.

It is needed to understand that networking parameters do not affect the operation of a remote site. The commands and processes can be started in a harsh low bandwidth high latency environment. There are two separate instances that might be affected: the tunnel negotiated between sites and the actual range operation software. In this study, both of the aspects were investigated and with some really positively surprising results. The tunnelling interface could not be broken with values less than stated in the official documentation. The remote data centre operation software is working on a local server resource instance in the partnering geographic location and as far there is a tunnelled Internet connection to it, operations commands can be sent and the initiation will be on the remote site.

Definitely, all the ranges use some kind of open source tools. A tool like such was implemented into the test environment to see how it behaves in a federated environment. This kind of tools are cost effective at the beginning, but when someone starts to implement them and wants to customize features what are not available by default. As wanting to get the maximum out of the implemented tool, lots of time was spent on finding the right information, adding the features and troubleshooting problems. Finding solutions to implementation issues in the federation case might be challenging because it has a specific use case what other governmental and private companies might not have. On the other hand, enterprise solutions worked well out of the box and were not so labour intensive. The only downside of them is the initial investment needed. For this, support for implementations, configurations, hardware and software upgrades and replace parts can be received. Every range needs to adjudicate which heading to take. As for federation, if there were already some common enterprise solutions available, they would greatly benefit and accelerate the implementation process of range interconnection.

Increasing ranges fault tolerance or disaster recovery should not be the goal of federation. Those techniques should be implemented and managed by every party by themselves. Further discussions on that topic can be redefined only after a reasonable use case is presented. Additionally, some prerequisites for achieving high availability like bandwidth or other technological requirements need to be met. Low bandwidth produces slow migrations of resources what in turn means that fault tolerance and disaster recovery would lose their point.

The aim of a federated environment is to create a high availability and shared resources with the positive effect on the interoperability of operational cyber defence systems, organisations and processes, thereby improving the effectiveness and efficiency of cyber defence operations, procedures, testing tools, research and development, simulation, war gaming, competitions, trainings and multinational exercises. For NATO and EU bodies, federation helps to enhance the ability to train with teams that, despite the universal language of information and communications technology, have different cultural approaches to problem solving. It builds new relations, strengthens and deepen trust, cooperation across the borders.

References

1. EDA. (2015). EDA Ad Hoc Working Group (AHWG) on Cyber Ranges.

2. Davis, J., Magrath, S. (2013). A Survey of Cyber Ranges and Testbeds. Australian Government, Department of Defence.

3. Nance, Klara L., Hay B. (2009). Replicating and Sharing Computer Security Laboratory Environments. Proceedings of the 42nd Hawaii International Conference on System Sciences, pp 1 - 10.

4. Valtenberg, U. (2012). Estonian Defence force's Cyber Defence training and exercise environment (KTHK) use cases according to KTHK's capabilities today and in the future. Tallinn University of Technology. Tallinn.

5. Rautio, P. (2007). Planning an Empirical study [WWW] http://www2.uiah.fi/projects/metodi/144.htm (11.11.2016).

6. Estonian Ministry of Defence (2014). Estonian Defence Force’s Cyber Range.

7. Čeleda, P., Čegan, J. (2015) KYPO – A Platform for Cyber Defence Exercises. Masaryk University. Brno.

8. CERIT (2017). CERIT Scientific cloud. [WWW] https://www.cerit-sc.cz/en/ (07.02.2017).

9. Cisco.com (2014). Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide, pp LSC-104 - LSC-116, LSC-351 – LSC-364.

10. Cisco.com (2015). Cisco ASR9000 Enterprise L2VPN for Metro-Ethernet, DC-WAN, WAN-Core, and Government and Public Networks, pp. 1-1 – 1-2.

11. Bipin (2015). Configure Site to Site IPSec VPN Tunnel in Cisco IOS Router [WWW] http://www.mustbegeek.com/configure-site-to-site-ipsec-vpn-tunnel-in-cisco-ios-router/ (02.04.2017).

12. Spiceworks (2017). How IPsec VPN Site-to-Site Tunnels Work? [WWW] https://community.spiceworks.com/topic/341044-how-ipsec-vpn-site-to-site-tunnels-work (02.04.2017).

13. CME Group. Virtual Private Network [WWW] https://www.cmegroup.com/confluence/display/EPICSANDBOX/Virtual+Private+Network (02.04.2017).

14. Richweb (2012). IPSEC Site-to-Site VPN Requirements. [WWW] http://archive.richweb.com/node/100.html (02.04.2017).

15. Mahalingam, M., Dutt, D. (2014). Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks. RFC 7348.

16. Cisco.com (2015). VXLAN Overview: Cisco Nexus 9000 Series Switches [WWW] http://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/white-paper-c11-729383.pdf (04.04.2017).

17. Cisco.com (2014). Cisco ME 2600X Series Ethernet Access Switch Software Con-figuration Guide, pp. 360 – 369.

18. Mott, S. (2012). VPN Bonding. [WWW] https://www.simonmott.co.uk/2012/03/vpn-bonding/ (05.04.2017).

19. VMware (2015). Long Distance vMotion requirements in VMware vSphere 6.0 (2106949). [WWW] https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2106949 (15.03.2017).

20. WonderNetwork (2017). Global Ping Statistics [WWW] https://wondernetwork.com/pings (07.04.2017).

21. Sullivan, R. (2015). VMRC minimum network requirements - latency thresholds. [WWW] https://communities.vmware.com/thread/502371 (29.03.2017).

22. Microsoft, Desktop Virtualization Team (2008). Remote Desktop Protocol Performance. [WWW] http://download.microsoft.com/download/4/d/9/4d9ae285-3431-4335-a86e-969e7a146d1b/rdp_performance_whitepaper.docx (08.04.2017).

23. VMware addicted blog (2016). Difference between vSphere 5.5 vs 6.0 vs 6.5. [WWW] http://vmwareaddicted.blogspot.com.ee/2016/12/below-table-explain-differences-from.html (28.03.2017).

24. Kendrickcoleman.com (2014). Multiple vCenter Servers, SSO, and How to Design for Failure. [WWW] http://www.kendrickcoleman.com/index.php/Tech-Blog/multiple-vcenter-servers-sso-and-how-to-design-for-failure.html (2.12.2016).

25. VMware (2015). Installing vCenter Single Sign-On in a multisite deployment (2034074). [WWW] https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2034074 (08.02.2017).

26. OpenNebula Project (2017). Virtual Machine Definition Template. [WWW] https://docs.opennebula.org/5.2/operation/references/template.html (01.03.2017).

Appendix

I. Configurations

Estonian router configuration

/etc/network/interfaces file

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens160
iface ens160 inet static
        address 10.80.67.80
        netmask 255.255.255.0
        gateway 10.80.67.1

        dns-nameservers 10.80.67.5

auto ens192
iface ens192 inet static
        address 172.30.0.1
        netmask 255.255.255.0
```

Routing table

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         10.80.67.1      0.0.0.0         UG    0      0        0 ens160
10.80.67.0      *               255.255.255.0   U     0      0        0 ens160
147.251.49.10   10.80.67.1      255.255.255.255 UGH   0      0        0 ens160
172.30.0.0      *               255.255.255.0   U     0      0        0 ens192
192.168.2.0     *               255.255.255.0   U     0      0        0 tun0
muni.int        *               255.255.255.255 UH    0      0        0 tun0
192.168.2.192   *               255.255.255.252 U     0      0        0 tun0
```

Czech router configuration

/etc/network file

```
config interface 'loopback'
   option ifname 'lo'
   option proto 'static'
   option ipaddr '127.0.0.1'
   option netmask '255.0.0.0'

config interface 'lan'
   option ifname 'eth0.1'
   option force_link '1'
   option type 'bridge'
   option proto 'static'
   option netmask '255.255.255.0'
   option ipaddr '192.168.2.2'
   option gateway '192.168.2.1'
   option dns '192.168.2.50'

config interface 'wan'
   option ifname 'eth0.2'
   option proto 'dhcp'

config switch
   option name 'switch0'
   option reset '1'
   option enable_vlan '1'

config switch_vlan
   option device 'switch0'
   option vlan '1'
   option ports '0t 2 3 4 5'

config switch_vlan
   option device 'switch0'
   option vlan '2'
   option ports '0t 1'

config interface 'vpn'
   option proto 'none'
   option ifname 'vpns+'
```

Routing table

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         147.251.49.10   0.0.0.0         UG    0      0        0 eth0
10.80.67.0      sw              255.255.255.0   UG    0      0        0 br-lan
147.251.49.10   *               255.255.255.255 UH    0      0        0 eth0
172.30.0.0      sw              255.255.255.0   UG    0      0        0 br-lan
192.168.2.0     *               255.255.255.0   U     0      0        0 br-lan
```

/etc/ocserv/ocserv.conf.template file

```
#The plain option requires specifying a password file which contains
# entries of the following format.
# "username:groupname:encoded-password"
# One entry must be listed per line, and 'ocpasswd' can be used
# to generate password entries.
auth = "|AUTH|"

# A banner to be displayed on clients
banner = "KYPO range"

# When the server has a dynamic DNS address (that may change),
# should set that to true to ask the client to resolve again on
# reconnects.
listen-host-is-dyndns = |DYNDNS|

# Limit the number of clients. Unset or set to zero for unlimited.
max-clients = 128

# Limit the number of client connections to one every X milliseconds
# (X is the provided value). Set to zero for no limit.
rate-limit-ms = 100

# Limit the number of identical clients (i.e., users connecting
# multiple times). Unset or set to zero for unlimited.
max-same-clients = 8

# TCP and UDP port number
tcp-port = |PORT|
|UDP|udp-port = |PORT|

# Stats report time. The number of seconds after which each
# worker process will report its usage statistics (number of
# bytes transferred etc). This is useful when accounting like
# radius is in use.
#stats-report-time = 360

# Keepalive in seconds
keepalive = |KEEPALIVE|

# Dead peer detection in seconds.
dpd = |DPD|

# MTU discovery (DPD must be enabled)
try-mtu-discovery = false

# The key and the certificates of the server
# The key may be a file, or any URL supported by GnuTLS (e.g.,
# tpmkey:uuid=xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxx;storage=user
# or pkcs11:object=my-vpn-key;object-type=private)
#
# There may be multiple certificate and key pairs and each key
# should correspond to the preceding certificate.
server-cert = /etc/ocserv/server-cert.pem
server-key = /etc/ocserv/server-key.pem


# Uncomment this to enable compression negotiation (LZS, LZ4).
|COMPRESSION|compression = true

# GnuTLS priority string
tls-priorities = "NORMAL:%SERVER_PRECEDENCE:%COMPAT:-VERS-SSL3.0"

# The time (in seconds) that a client is allowed to stay connected prior
# to authentication
auth-timeout = 40

# Banning clients in ocserv works with a point system. IP addresses
# that get a score over that configured number are banned for
# min-reauth-time seconds. By default a wrong password attempt is 10 points,
# a KKDCP POST is 1 point, and a connection is 1 point. Note that
# due to difference processes being involved the count of points
# will not be real-time precise.
#
# Score banning cannot be reliably used when receiving proxied connections
# locally from an HTTP server (i.e., when listen-clear-file is used).
#
# Set to zero to disable.
max-ban-score = 50

# The time (in seconds) that all score kept for a client is reset.
ban-reset-time = 300

# Cookie timeout (in seconds)
# which he can reconnect. That cookie will be invalided if not
# used within this timeout value. On a user disconnection, that
# cookie will also be active for this time amount prior to be
# invalid. That should allow a reasonable amount of time for roaming
# between different networks.
cookie-timeout = 300

# Whether roaming is allowed, i.e., if true a cookie is
# restricted to a single IP address and cannot be re-used
# from a different IP.
deny-roaming = false

# ReKey time (in seconds)
# ocserv will ask the client to refresh keys periodically once
# this amount of seconds is elapsed. Set to zero to disable.
rekey-time = 172800
```

```
# UTMP
use-utmp = false

# Whether to enable support for the occtl tool (i.e., either through D-BUS,
# or via a unix socket).
use-occtl = true

# socket file used for IPC with occtl. You only need to set that,
# if you use more than a single servers.
occtl-socket-file = /var/run/occtl.socket

# PID file. It can be overriden in the command line.
pid-file = /var/run/ocserv.pid

# The default server directory. Does not require any devices present.
chroot-dir = /var/lib/ocserv

# socket file used for IPC, will be appended with .PID
# It must be accessible within the chroot environment (if any)
#socket-file = /var/run/ocserv-socket
socket-file = ocserv-socket

# The user the worker processes will be run as. It should be
# unique (no other services run as this user).
run-as-user = ocserv
run-as-group = ocserv

#
# Network settings
#

# The name of the tun device
device = vpns

# Whether the generated IPs will be predictable, i.e., IP stays the
# same for the same user when possible.
predictable-ips = |PREDICTABLE_IPS|

# The default domain to be advertised
|ENABLE_DEFAULT_DOMAIN|default-domain = |DEFAULT_DOMAIN|

# The pool of addresses that leases will be given from.
ipv4-network = |IPV4ADDR|
ipv4-netmask = |NETMASK|

# The IPv6 subnet that leases will be given from.
|ENABLE_IPV6|ipv6-network = |IPV6ADDR|
|ENABLE_IPV6|ipv6-prefix = |IPV6PREFIX|

# Prior to leasing any IP from the pool ping it to verify that
# it is not in use by another (unrelated to this server) host.
ping-leases = false

# Unset to assign the default MTU of the device
# mtu =

# Configuration files that will be applied per user connection or
# per group. Each file name on these directories must match the username
# or the groupname.
# The options allowed in the configuration files are dns, nbns,
#  ipv?-network, ipv4-netmask, ipv6-prefix, rx/tx-per-sec, iroute, route,
#  net-priority and cgroup.
#
# Note that the 'iroute' option allows to add routes on the server
# based on a user or group. The syntax depends on the input accepted
# by the commands route-add-cmd and route-del-cmd (see below).
config-per-user = /etc/ocserv/config-per-user/
```

```
config-per-group = /etc/ocserv/config-per-group/

# Instead of specifying manually all the allowed groups, you may instruct
# ocserv to scan all available groups and include the full list. That
# option is only functional on plain authentication.
#auto-select-group = true

# The system command to use to setup a route. %{R} will be replaced with the
# route/mask and %{D} with the (tun) device.
route-add-cmd = "/sbin/route add -net %{R} dev %{D}"
route-del-cmd = "/sbin/route del -net %{R} dev %{D}"

# Unless set to false it is required for clients to present their
# certificate even if they are authenticating via a previously granted
# cookie and complete their authentication in the same TCP connection.
# Legacy CISCO clients do not do that, and thus this option should be
# set for them.
cisco-client-compat = |CISCO_COMPAT|
```

/etc/ocserv/ocserv.conf file

```
config ocserv 'config'
   option keepalive '360'
   option dpd '180'
   option max_clients '8'
   option zone 'lan'
   option enable '1'
   option auth 'plain'
   option port '443'
   option ipaddr '192.168.2.192'
   option netmask '255.255.255.253'
   option cisco-client-compat 'true'
   option _ca '-----BEGIN CERTIFICATE-----
MIIC6jCCAdKgAwIBAgIBATANBgkqhkiG9w0BAQsFADAVMRMwEQYDVQQDEwpPcGVu
V3J0IENBMCAXDTE2MDIxNzE2MTMxNFoYDzk5OTkxMjMxMjM1OTU5WjAVMRMwEQYD
VQQDEwpPcGVuV3J0IENBMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA
r3aH6IuBChTLEoXUc1pwWEM4xokRU9QIjEEtV9APTWiG5sMb8QgIRknW8Y26hF3X
XMBBCb0bVfuDzo8aajMYbGGg8WUx9QUveN5RyENwrQOAt6GbfqQ7yMuCvZmVs0ai
VH5+DXZyViIqNKk4iyNWzQ6RnpTjb2Clfb70px2FZamKhEp1KohRknal1FPVCUMQ
9az7b9W7t4LJbNjIx1GdKyv91aiMMpqdFWR9/XYOtki3kCqxvWToSYl4+0Do26sj
Jgg54Ea7lJk3GhMVtM7laBYQw2Om1yxQRfM2NUjzvQdo/RtzujcVBg4WrEsTC0FS
hu/693VaiXO+pKotINJ4OQIDAQABo0MwQTAPBgNVHRMBAf8EBTADAQH/MA8GA1Ud
DwEB/wQFAwMHBAAwHQYDVR0OBBYEFEr7wK9fAO9V70wb+9mRAOIe3bYuMA0GCSqG
SIb3DQEBCwUAA4IBAQADW2+TOvWk5qLb0nNeGLbKdGlnCVhoqvzjv5Ix8e3NvT3X
EB45Lns+AI6U6yQKRvJHswoBrGOoH8xU8+6m5KPJEXENrDsXI6RsjUouetG4Sz6G
n8Xf/CGJzld/fior9u8pu7ZOjbeiThFynS2IEfT62d2ZYjKHgQBkq07gqjB7ExJq
T0eT5ZzwHshoqAbOQ8o7mZ10QWmcTsPVz6aZPsriO6mNXuOoOfNAGryM/XfPY/kv
OrOG9Ig3VvlcUDmR5lu46y0hZK6MIxHr7Kd23JWdD48KrDdK5wgVcrFFngWFzg6r
uNtrxEAowNsjxNilTTgDXCbT5Ax3nFSR76OQGBHW
-----END CERTIFICATE-----
'
   option split_dns '1'
   option default_domain 'muni.int'
   option max_same '4'

config dns
   option ip '192.168.2.50'

config routes
   option netmask '255.255.255.0'
   option ip '192.168.2.0'

config ocservusers
   option name 'allar'
   option password '$1$39185748$Pzi.3x1i3rcDhoPGB/ulF1'
```

OpenNebula configuration

/etc/oned.conf file

```
#*******************************************************************************
#                       OpenNebula Configuration file
#*******************************************************************************

#*******************************************************************************
# Daemon configuration attributes
#-------------------------------------------------------------------------------
#  SCRIPTS_REMOTE_DIR: Remote path to store the monitoring and VM management
#  scripts.
#
#  PORT: Port where oned will listen for xmlrpc calls.
#  LISTEN_ADDRESS: Host IP to listen on for xmlrpc calls (default: all IPs).
#
#
#  VNC_PORTS: VNC port pool for automatic VNC port assignment, if possible the
#  port will be set to ``START`` + ``VMID``
#   start   : first port to assign
#   reserved: comma separated list of ports or ranges. Two numbers separated by
#   a colon indicate a range.
#*******************************************************************************

LOG = [
  SYSTEM      = "file",
  DEBUG_LEVEL = 3
]

MONITORING_INTERVAL = 60
MONITORING_THREADS  = 50
SCRIPTS_REMOTE_DIR=/var/tmp/one

PORT = 2633

LISTEN_ADDRESS = "0.0.0.0"

#configuration for MySQL
 DB = [ BACKEND = "mysql",
        SERVER  = "localhost",
        PORT    = 0,
        USER    = "oneadmin",
        PASSWD  = "rootroot",
        DB_NAME = "opennebula" ]

VNC_PORTS = [
    START    = 5900
#    RESERVED = "6800, 6801, 6810:6820, 9869"
]
#*******************************************************************************
# Federation configuration attributes
#-------------------------------------------------------------------------------
# Control the federation capabilities of oned. Operation in a federated setup
# requires a special DB configuration.
#
#  FEDERATION: Federation attributes
#   MODE: Operation mode of this oned.
#       STANDALONE no federated.This is the default operational mode
#       MASTER     this oned is the master zone of the federation
#       SLAVE      this oned is a slave zone
#   ZONE_ID: The zone ID as returned by onezone command
#   MASTER_ONED: The xml-rpc endpoint of the master oned, e.g.
#   http://master.one.org:2633/RPC2
```

```
#*******************************************************************************

FEDERATION = [
    MODE        = "STANDALONE",
    ZONE_ID     = 0,
    MASTER_ONED = ""
]

#*******************************************************************************
# Default showback cost
#-------------------------------------------------------------------------------
# The following attributes define the default cost for Virtual Machines that
# don't have a CPU, MEMORY or DISK cost. This is used by the oneshowback
# calculate method.
#*******************************************************************************

DEFAULT_COST = [
    CPU_COST    = 0,
    MEMORY_COST = 0,
    DISK_COST   = 0
]

#*******************************************************************************
# Physical Networks configuration
#*******************************************************************************
#  NETWORK_SIZE: Here you can define the default size for the virtual networks
#
#  MAC_PREFIX: Default MAC prefix to be used to create the auto-generated MAC
#  addresses is defined here (this can be overwritten by the Virtual Network
#  template)
#
#  VLAN_IDS: VLAN ID pool for the automatic VLAN_ID assignment. This pool
#  is for 802.1Q networks (Open vSwitch and 802.1Q drivers). The driver
#  will try first to allocate VLAN_IDS[START] + VNET_ID
#     start: First VLAN_ID to use
#     reserved: Comma separated list of VLAN_IDs or ranges. Two numbers
#     separated by a colon indicate a range.
#
# VXLAN_IDS: Automatic VXLAN Network ID (VNI) assignment. This is used
# for vxlan networks.
#     start: First VNI to use
#     NOTE: reserved is not supported by this pool
#
#*******************************************************************************

NETWORK_SIZE = 254

MAC_PREFIX   = "02:00"

VLAN_IDS = [
    START    = "2",
    RESERVED = "0, 1, 4095"
]

VXLAN_IDS = [
    START = "2"
]

#*******************************************************************************
# Information Driver Configuration
#*******************************************************************************
# You can add more information managers with different configurations but make
# sure it has different names.
#
#   name      : name for this information manager
#
#   executable: path of the information driver executable, can be an
#               absolute path or relative to $ONE_LOCATION/lib/mads (or
#               /usr/lib/one/mads/ if OpenNebula was installed in /)
#
#   arguments : for the driver executable, usually a probe configuration file,
#               can be an absolute path or relative to $ONE_LOCATION/etc (or
#               /etc/one/ if OpenNebula was installed in /)
#*******************************************************************************

#-------------------------------------------------------------------------------
#  Information Collector for KVM IM's.
#-------------------------------------------------------------------------------
#  This driver CANNOT BE ASSIGNED TO A HOST, and needs to be used with KVM
#    -h  prints this help.
#    -a  Address to bind the collectd socket (default 0.0.0.0)
#    -p  UDP port to listen for monitor information (default 4124)
#    -f  Interval in seconds to flush collected information (default 5)
#    -t  Number of threads for the server (default 50)
#    -i  Time in seconds of the monitorization push cycle. This parameter must
#        be smaller than MONITORING_INTERVAL, otherwise push monitorization will
#        not be effective.
#    -w  Timeout in seconds to execute external commands (default unlimited)
#-------------------------------------------------------------------------------
IM_MAD = [
      NAME       = "collectd",
      EXECUTABLE = "collectd",
      ARGUMENTS  = "-p 4124 -f 5 -t 50 -i 20" ]
#-------------------------------------------------------------------------------

#-------------------------------------------------------------------------------
#  KVM UDP-push Information Driver Manager Configuration
#    -r number of retries when monitoring a host
#    -t number of threads, i.e. number of hosts monitored at the same time
#    -w Timeout in seconds to execute external commands (default unlimited)
#-------------------------------------------------------------------------------
IM_MAD = [
      NAME          = "kvm",
      SUNSTONE_NAME = "KVM",
      EXECUTABLE    = "one_im_ssh",
      ARGUMENTS     = "-r 3 -t 15 kvm" ]
#-------------------------------------------------------------------------------

#-------------------------------------------------------------------------------
#  KVM SSH-pull Information Driver Manager Configuration
#    -r number of retries when monitoring a host
#    -t number of threads, i.e. number of hosts monitored at the same time
#    -w Timeout in seconds to execute external commands (default unlimited)
#-------------------------------------------------------------------------------
# IM_MAD = [
#       NAME          = "kvm",
#       SUNSTONE_NAME = "kvm-ssh",
#       EXECUTABLE    = "one_im_ssh",
#       ARGUMENTS     = "-r 3 -t 15 kvm-probes" ]
#-------------------------------------------------------------------------------

#-------------------------------------------------------------------------------
#  vCenter Information Driver Manager Configuration
#    -r number of retries when monitoring a host
#    -t number of threads, i.e. number of hosts monitored at the same time
#    -w Timeout in seconds to execute external commands (default unlimited)
#-------------------------------------------------------------------------------
IM_MAD = [
      NAME          = "vcenter",
      SUNSTONE_NAME = "VMWare vCenter",
      EXECUTABLE    = "one_im_sh",
      ARGUMENTS     = "-c -t 15 -r 0 vcenter" ]
#-------------------------------------------------------------------------------

# For R 3.4 VMware Information Driver Manager Configuration
IM_MAD = [
 name = "im_vmware",
 executable = "one_im_sh",
 arguments = "-t 15 -r 0 vmware" ]


#-------------------------------------------------------------------------------

#*******************************************************************************
# Virtualization Driver Configuration
#*******************************************************************************
#-------------------------------------------------------------------------------
#  KVM Virtualization Driver Manager Configuration
#    -r number of retries when monitoring a host
#    -t number of threads, i.e. number of hosts monitored at the same time
#    -l <actions[=command_name]> actions executed locally, command can be
#        overridden for each action.
#        Valid actions: deploy, shutdown, cancel, save, restore, migrate, poll
#        An example: "-l migrate=migrate_local,save"
#    -p more than one action per host in parallel, needs support from hypervisor
#    -s <shell> to execute remote commands, bash by default
#    -w Timeout in seconds to execute external commands (default unlimited)
#
#  Note: You can use type = "qemu" to use qemu emulated guests, e.g. if your
#  CPU does not have virtualization extensions or use nested Qemu-KVM hosts
#-------------------------------------------------------------------------------
VM_MAD = [
    NAME           = "kvm",
    SUNSTONE_NAME  = "KVM",
    EXECUTABLE     = "one_vmm_exec",
    ARGUMENTS      = "-t 15 -r 0 kvm",
    DEFAULT        = "vmm_exec/vmm_exec_kvm.conf",
    TYPE           = "kvm",
    KEEP_SNAPSHOTS = "no",
    IMPORTED_VMS_ACTIONS = "terminate, terminate-hard, hold, release, suspend,
        resume, delete, reboot, reboot-hard, resched, unresched, disk-attach,
        disk-detach, nic-attach, nic-detach, snap-create, snap-delete"
]

#-------------------------------------------------------------------------------

#-------------------------------------------------------------------------------
#  vCenter Virtualization Driver Manager Configuration
#    -r number of retries when monitoring a host
#    -t number of threads, i.e. number of hosts monitored at the same time
#    -p more than one action per host in parallel, needs support from hypervisor
#    -s <shell> to execute commands, bash by default
#    -d default snapshot strategy. It can be either 'detach' or 'suspend'. It
#       defaults to 'suspend'.
#    -w Timeout in seconds to execute external commands (default unlimited)
#-------------------------------------------------------------------------------
VM_MAD = [
    NAME           = "vcenter",
    SUNSTONE_NAME  = "VMWare vCenter",
    EXECUTABLE     = "one_vmm_sh",
    ARGUMENTS      = "-p -t 15 -r 0 vcenter -s sh",
    DEFAULT        = "vmm_exec/vmm_exec_vcenter.conf",
    TYPE           = "xml",
    KEEP_SNAPSHOTS = "yes",
    IMPORTED_VMS_ACTIONS = "terminate, terminate-hard, hold, release, suspend,
        resume, delete, reboot, reboot-hard, resched, unresched, poweroff,
        poweroff-hard, disk-attach, disk-detach, nic-attach, nic-detach,
        snap-create, snap-delete"
]
#-------------------------------------------------------------------------------

VM_MAD = [
 name = "vmm_vmware",
 executable = "one_vmm_sh",
 arguments = "-t 15 -r 0 vmware",
 default = "vmm_exec/vmm_exec_vmware.conf",
 type = "vmware" ]



#*******************************************************************************
# Transfer Manager Driver Configuration
#*******************************************************************************
# You can add more transfer managers with different configurations but make
# sure it has different names.
#   name      : name for this transfer driver
#
#   executable: path of the transfer driver executable, can be an
#               absolute path or relative to $ONE_LOCATION/lib/mads (or
#               /usr/lib/one/mads/ if OpenNebula was installed in /)
#   arguments :
#       -t: number of threads, i.e. number of transfers made at the same time
#       -d: list of transfer drivers separated by commas, if not defined all the
#           drivers available will be enabled
#       -w: Timeout in seconds to execute external commands (default unlimited)
#*******************************************************************************

TM_MAD = [
    EXECUTABLE = "one_tm",
    ARGUMENTS = "-t 15 -d
dummy,lvm,shared,fs_lvm,qcow2,ssh,ceph,dev,vcenter,iscsi_libvirt"
]

TM_MAD = [
name = "tm_vmware",
executable = "one_tm",
arguments = "tm_vmware/tm_vmware.conf" ]

TM_MAD = [
 name = "tm_nfs",
 executable = "one_tm",
 arguments = "tm_shared/tm_shared.conf" ]


#*******************************************************************************
# Datastore Driver Configuration
#*******************************************************************************
# Drivers to manage the datastores, specialized for the storage backend
#   executable: path of the transfer driver executable, can be an
#               absolute path or relative to $ONE_LOCATION/lib/mads (or
#               /usr/lib/one/mads/ if OpenNebula was installed in /)
#
#   arguments : for the driver executable
#       -t number of threads, i.e. number of repo operations at the same time
#       -d datastore mads separated by commas
#       -s system datastore tm drivers, used to monitor shared system ds.
#       -w Timeout in seconds to execute external commands (default unlimited)
#*******************************************************************************

DATASTORE_MAD = [
    EXECUTABLE = "one_datastore",
    ARGUMENTS  = "-t 15 -d dummy,fs,lvm,ceph,dev,iscsi_libvirt,vcenter -s
shared,ssh,ceph,fs_lvm,qcow2,vmware"
]




#*******************************************************************************
# Auth Manager Configuration
#*******************************************************************************
# AUTH_MAD: The Driver that will be used to authenticate (authn) and
# authorize (authz) OpenNebula requests. If defined OpenNebula will use the
# built-in auth policies.
#
#   executable: path of the auth driver executable, can be an
#               absolute path or relative to $ONE_LOCATION/lib/mads (or
#               /usr/lib/one/mads/ if OpenNebula was installed in /)
#
#   authn     : list of authentication modules separated by commas, if not
#               defined all the modules available will be enabled
#   authz     : list of authentication modules separated by commas
#
# DEFAULT_AUTH: The default authentication driver to use when OpenNebula does
# not know the user and needs to authenticate it externally.  If you want to
# use "default" (not recommended, but supported for backwards compatibility
# reasons) make sure you create a symlink pointing to the actual authentication
# driver in /var/lib/one/remotes/auth, and add "default" to the 'auth'
# parameter in the 'AUTH_MAD' section.
#
# SESSION_EXPIRATION_TIME: Time in seconds to keep an authenticated token as
# valid. During this time, the driver is not used. Use 0 to disable session
# caching
#
# ENABLE_OTHER_PERMISSIONS: Whether or not users can set the permissions for
# 'other', so publishing or sharing resources with others. Users in the oneadmin
# group will still be able to change these permissions. Values: YES or NO.
#
# DEFAULT_UMASK: Similar to Unix umask, sets the default resources permissions.
# Its format must be 3 octal digits. For example a umask of 137 will set
# the new object's permissions to 640 "um- u-- ---"
#*******************************************************************************

AUTH_MAD = [
    EXECUTABLE = "one_auth_mad",
    AUTHN = "ssh,x509,ldap,server_cipher,server_x509"
]

#DEFAULT_AUTH = "default"

SESSION_EXPIRATION_TIME = 900

#ENABLE_OTHER_PERMISSIONS = "YES"

DEFAULT_UMASK = 177


#*******************************************************************************
# Transfer Manager Driver Behavior Configuration
#*******************************************************************************
# The  configuration for each driver is defined in TM_MAD_CONF. These
# values are used when creating a new datastore and should not be modified
# since they define the datastore behavior.
#   name      : name of the transfer driver, listed in the -d option of the
#               TM_MAD section
#   ln_target : determines how the persistent images will be cloned when
#               a new VM is instantiated.
#       NONE: The image will be linked and no more storage capacity will be used
#       SELF: The image will be cloned in the Images datastore
#       SYSTEM: The image will be cloned in the System datastore
#   clone_target : determines how the non persistent images will be
#                  cloned when a new VM is instantiated.
#       NONE: The image will be linked and no more storage capacity will be used
#       SELF: The image will be cloned in the Images datastore
#       SYSTEM: The image will be cloned in the System datastore
#   shared : determines if the storage holding the system datastore is shared
#            among the different hosts or not. Valid values: "yes" or "no"
#   ds_migrate : The driver allows migrations across datastores. Valid values:
#               "yes" or "no". Note: THIS ONLY APPLIES TO SYSTEM DS.
#*******************************************************************************
```

```
TM_MAD_CONF = [
    NAME = "dummy", LN_TARGET = "NONE", CLONE_TARGET = "SYSTEM", SHARED = "YES",
    DS_MIGRATE = "YES"
]

TM_MAD_CONF = [
    NAME = "lvm", LN_TARGET = "NONE", CLONE_TARGET = "SELF", SHARED = "YES"
]

TM_MAD_CONF = [
    NAME = "shared", LN_TARGET = "NONE", CLONE_TARGET = "SYSTEM", SHARED =
"YES",
    DS_MIGRATE = "YES"
]

TM_MAD_CONF = [
    NAME = "fs_lvm", LN_TARGET = "SYSTEM", CLONE_TARGET = "SYSTEM", SHARED="YES"
]

TM_MAD_CONF = [
    NAME = "qcow2", LN_TARGET = "NONE", CLONE_TARGET = "SYSTEM", SHARED = "YES"
]

TM_MAD_CONF = [
    NAME = "ssh", LN_TARGET = "SYSTEM", CLONE_TARGET = "SYSTEM", SHARED = "NO",
    DS_MIGRATE = "YES"
]

TM_MAD_CONF = [
    NAME = "ceph", LN_TARGET = "NONE", CLONE_TARGET = "SELF", SHARED = "YES",
    DS_MIGRATE = "NO"
]

TM_MAD_CONF = [
    NAME = "iscsi_libvirt", LN_TARGET = "NONE", CLONE_TARGET = "SELF", SHARED =
"YES",
    DS_MIGRATE = "NO"
]

TM_MAD_CONF = [
    NAME = "dev", LN_TARGET = "NONE", CLONE_TARGET = "NONE", SHARED = "YES"
]

TM_MAD_CONF = [
    NAME = "vcenter", LN_TARGET = "NONE", CLONE_TARGET = "NONE", SHARED = "YES"
]

#*******************************************************************************
# Datastore Manager Driver Behavior Configuration
#*******************************************************************************
# The  configuration for each driver is defined in DS_MAD_CONF. These
# values are used when creating a new datastore and should not be modified
# since they define the datastore behavior.
#   name      : name of the transfer driver, listed in the -d option of the
#               DS_MAD section
#   required_attrs : comma separated list of required attributes in the DS
#                    template
#   persistent_only: specifies whether the datastore can only manage persistent
#                    images
#*******************************************************************************

DS_MAD_CONF = [
    NAME = "ceph",
    REQUIRED_ATTRS = "DISK_TYPE,BRIDGE_LIST",
    PERSISTENT_ONLY = "NO",
    MARKETPLACE_ACTIONS = "export"
]
```

```
DS_MAD_CONF = [
    NAME = "dev", REQUIRED_ATTRS = "DISK_TYPE", PERSISTENT_ONLY = "YES"
]

DS_MAD_CONF = [
    NAME = "iscsi_libvirt", REQUIRED_ATTRS = "DISK_TYPE,ISCSI_HOST",
    PERSISTENT_ONLY = "YES"
]

DS_MAD_CONF = [
    NAME = "dummy", REQUIRED_ATTRS = "", PERSISTENT_ONLY = "NO"
]

DS_MAD_CONF = [
    NAME = "fs", REQUIRED_ATTRS = "", PERSISTENT_ONLY = "NO",
    MARKETPLACE_ACTIONS = "export"
]

DS_MAD_CONF = [
    NAME = "lvm", REQUIRED_ATTRS = "DISK_TYPE,BRIDGE_LIST",
    PERSISTENT_ONLY = "NO"
]

DS_MAD_CONF = [
    NAME = "vcenter", REQUIRED_ATTRS = "VCENTER_CLUSTER", PERSISTENT_ONLY =
"YES",
    MARKETPLACE_ACTIONS = "export"
]
#*******************************************************************************
# Authentication Driver Behavior Definition
#*******************************************************************************
# The configuration for each driver is defined in AUTH_MAD_CONF. These
# values must not be modified since they define the driver behavior.
#   name            : name of the auth driver
#   password_change : allow the end users to change their own password. Oneadmin
#                     can still change other user's passwords
#   driver_managed_groups : allow the driver to set the user's group even after
#                     user creation. In this case addgroup, delgroup and chgrp
#                     will be disabled, with the exception of chgrp to one of
#                     the groups in the list of secondary groups
#   max_token_time  : limit the maximum token validity, in seconds. Use -1 for
#                     unlimited maximum, 0 to disable login tokens
#*******************************************************************************

AUTH_MAD_CONF = [
    NAME = "core",
    PASSWORD_CHANGE = "YES",
    DRIVER_MANAGED_GROUPS = "NO",
    MAX_TOKEN_TIME = "-1"
]

AUTH_MAD_CONF = [
    NAME = "public",
    PASSWORD_CHANGE = "NO",
    DRIVER_MANAGED_GROUPS = "NO",
    MAX_TOKEN_TIME = "-1"
]

AUTH_MAD_CONF = [
    NAME = "ssh",
    PASSWORD_CHANGE = "YES",
    DRIVER_MANAGED_GROUPS = "NO",
    MAX_TOKEN_TIME = "-1"
]

AUTH_MAD_CONF = [
    NAME = "x509",
    PASSWORD_CHANGE = "NO",
    DRIVER_MANAGED_GROUPS = "NO",
    MAX_TOKEN_TIME = "-1"
]

AUTH_MAD_CONF = [
    NAME = "ldap",
    PASSWORD_CHANGE = "YES",
    DRIVER_MANAGED_GROUPS = "YES",
    MAX_TOKEN_TIME = "86400"
]

AUTH_MAD_CONF = [
    NAME = "server_cipher",
    PASSWORD_CHANGE = "NO",
    DRIVER_MANAGED_GROUPS = "NO",
    MAX_TOKEN_TIME = "-1"
]

AUTH_MAD_CONF = [
    NAME = "server_x509",
    PASSWORD_CHANGE = "NO",
    DRIVER_MANAGED_GROUPS = "NO",
    MAX_TOKEN_TIME = "-1"
]
```

II. License

Non-exclusive licence to reproduce thesis and make thesis public

I, **Allar Vallaots**,

(*author's name*)

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

   1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

   1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

of my thesis

Federation of Cyber Ranges,

(title of thesis)

supervised by Jaan Priisalu, Uko Valtenberg, Raimundas Matulevičius,

(supervisor's name)

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, **23.05.2017**