Towards Digital Investigation in Virtual Networks: A Study of Challenges and Open Problems

Daniel Spiekermann
FernUniversität Hagen
58084 Hagen, Germany
daniel.spiekermann@fernuni-hagen.de

Tobias Eggendorfer
Hochschule Ravensburg-Weingarten
88250 Weingarten, Germany
tobias.eggendorfer@hs-weingarten.de

***Abstract*—The evolution of virtualization techniques is still changing operating principles in today's datacenters (DC). The virtualization of ordinary servers was just the first step. With virtualization the dynamic and flexibility of the DC is increased. Providers are now able to offer different virtual machines (VM) faster and with less overhead to their customers. But this provision raises new problems for the providers. Aspects like isolation, security or multi-tenancy are increasingly relevant and demand for new setups in the DC. Current network infrastructures are not able to handle these aspects with an acceptable effort. The development of virtual networks offers new possibilities, with benefits for the provider and the user. Based on a physical underlay network different virtual networks can be defined, either by an provider or the customer. Protocols like VXLAN or GENEVE appear to eliminate restrictions of current networks. New paradigms like Software-defined-Networks (SDN) or Network Function Virtualization (NFV) offer new capabilities to redesign the whole network infrastructure in the DC. But the need for digital investigation is still necessary regardless of all new paradigms and evolution. Particularly for network forensic investigation (NFI) modern virtual data centers and networks are adding complexity. NFI is used to examine network traffic by capturing the data of a suspicious target system and analyzing this data. With the rise of virtual networks the capture process needs to be refined. In order to further analyze captured traffic new capabilities are needed. In this paper, we analyze in detail new arising problems of digital investigation in virtual networks and explore the new challenges for NFI. Based on the discussion of network forensics and current utilized methodologies and the new techniques of network virtualization the arising problems are defined and in detail classified. This classification helps to develop new methods and possible solutions, which might simplify further necessary investigations in cloud-computing environments.**

I. Introduction

The evolution of cloud computing environments led to new use of information and communication technology (ICT). Nowadays customers use virtual machines (VM), application or storage pools maintained by a third-party instead of managing the complete ICT infrastructure on their own. Not only are companies and professionals taking advantage of the new possibilities but also the average user.

The cloud service providers (CSP) offers different services, [1] distinguishes between three implementations. An Infrastructure-as-a-Service (IaaS) environment offers the access to different VMs, each of them is configured by the customer. A Platform-as-a-Service (PaaS) provides a runtime environment for different application or programming languages. Users of this PaaS are able to implement their own applications using this virtual environment. The main advantage is the renunciation of administrating the run-time environment, the hardware and all affiliated components. Last the Software-as-a-Service (SaaS) offers only one special application, to be used by the customer.

The customers scope of influence of PaaS and SaaS environments is limited, but the usage of VMs provided by an IaaS environment raise new problems for CSPs. Customers want their VMs interconnected with each other, desire a flexible, but secure environment and the possibilities to change their leased environment on their own. They even want to decide if a VM is connected to the internet or is separated from other VMs in a multi-tier architecture. Providers endeavor an easy to manage environment with few additional manual interventions to reduce costs and error susceptibility.

The implementation of virtual networks involves new options to overcome the problems and to satisfy customer and CSP. Virtual networks abstract from the underlying physical network and provide a large number of so-called overlay networks.

These virtual networks expand the flexibility inside the IT and enable the customer to maintain the assigned network on their own. The CSP does not have to configure the requested network environment, the customer is able to configure the network infrastructure and desired connections without any further support.

However, the need for digital investigation in virtual environments is nevertheless essential. Digital forensics encompass the recovery and analysis of data, stored or processed on digital devices to investigate cyber-crimes. A digital investigation inside a cloud environment is already a complex, error-prone and tedious task, aggravated by the flexible environment and jurisdictional, organizational or technical problems [2].

By up-scaling the virtual environment with the virtual network, the digital investigation gets more serious. Especially NFI is faced with a large number of new problems which may abort the investigation. NFI, as a branch of digital investigations, is mostly a reliable source of information gathering in current networks.

Networks are one of the most important parts within an IT environment. They establish the connection between client and server, provider and customer or storage systems and

application by transferring the requested data from sender to receiver. By intercepting and recording this traffic for a posterior analysis, NFI may extract relevant information to clarify the issue.

The remainder of this article is structured as follows. Section II lists work, relating to network forensics, network virtualization and digital investigation in virtual environments. Section III defines the process of NFI, clarifies differences to other sections of digital investigation and explains in detail the three main phases of a NFI. Section IV provides the background knowledge of network virtualization and its different parts like SDN, NFV and relevant protocols. The arising problems of network forensic investigation are listed in section V. Section VI concludes and gives an outlook of the further intended research.

II. Related Work

A lot of research is done in the wide field of network forensics or cloud computing, but the combination of these two sections is still underexplored.

Cloud forensic and existing problems are discussed in [2] and [3]. The existing coherence of network and cloud forensics are described in [4]. One of the first approaches to implement digital investigation in cloud computing environments is discussed in [5].

The multitude of different implementations of network virtualization is discussed in [6], [7], [8] and [9].

Software Defined Networks and the inherent principles and features of SDN are discussed in [10]. OpenFlow as the first standard communication interface is introduced in [11], a first research of OpenFlow forensics and the opportunity in investigating different attacks in SDN networks can be found in [12].

Network Function Virtualization is discussed in [13] and [14]. An implementation of a virtual switch named Open vSwitch is introduced in [15].

Network forensics is an established part of digital investigations. Like computer or mobile phone forensics it is necessary for correct and valid results. Tools and techniques are discussed in [16] and [17]. [18] describes the integration of digital investigation in network environments.

Different researches were made in the past according to specific network traffic and applications. [19] discusses the possibilities of network forensics in cloud-computing environments and the identification of different services.

This paper discusses critical problems in virtual network, which make a network forensic investigation more complicated or abort them.

III. Network forensics

Digital investigations are used to solve crimes involving computers, networks or other IT components. Depending on the kind of cyber-crime, different highly specialized investigation methods are necessary. If the communication between two systems contains relevant information, the analysis in detail of this data is helpful to identify the further circumstances. The combination of the retrieved results with other branches of digital investigation like computer forensics might improve the overall examination.

[20] defines network forensic as

> *the use of scientifically proven techniques to collect, fuse, identify, examine, correlate, analyze, and document digital evidence from multiple, actively processing and transmitting digital sources for the purpose of uncovering facts related to the planned intent, or measured success of unauthorized activities meant to disrupt, corrupt, and or compromise system components as well as providing information to assist in response to or recovery from these activities.*

Thereby NFI is separated into different phases [21], which can be classified in *Capture*, *Record* and *Analysis*. We define the two aforementioned phases as online, the last part of analyzing as the offline phase.

A. Capture

Contrary to the other branches of digital investigation there is no automatic data storage without interventions. Personal computer (PC) or mobile phones store the user data even when the system is shut down. Network devices transfer the data from sender to receiver over a separate medium without storing any of this data. The sometimes used buffering of the network data is only used to compensate latency or short-ranged interferences and yields no helpful information.

The purpose of the capture process is to get all transferred data of the target system without packet loss or a potentially manipulation of the packets. Because of this an approach of manipulating routing tables or firewall rules is not acceptable in NFI.

The capture process is necessary to gain access to the transferred data and to copy the data. Three main techniques exist to fulfill this requirement:

- Bridge
  A Bridge is a separate device interleaved in the connection between the target system and the next network device. If the bridge device loses the connection, the whole communication to the target system breaks down.
- Tap
  A Test-Access-Port defines a special network device, which is interleaved in the cable connection between target system and network device. The main advantage of a TAP against a bridge is the established connection, which remains online even when the TAP crashes.
- Mirroring
  A port mirror[1] can be created on professional network switches to mirror all traffic from or to a given physical port on the switch. The switch copies all network traffic passing this source port and delivers it to the destination

[1] also called Switch Port ANalyzer (SPAN)

port. If the mirror crash[2], the connection between sender and receiver is still online.

Two other methods to copy the transferred data are a proxy-environment or the use of dedicated VMs, but both implementations are incongruous in virtual networks.

B. Record

The mirrored data needs to be recorded by a separate storage system. The amount of captured data depends of the target system and should be suitable for transmission peaks or a higher transfer rate.

The limiting factor of the storage system is the write rate of the hard disk. Networks with a transfer rate of 10 gigabit per second (GBit/s)[3] send and receive approximately 1.1 GB/s, so the storage system has to write this data with the same speed to the hard disk to avoid data loss. Higher transfer rates like 40GE or 100GE transfer even more data per second, so that the storage system has to be adapted to the network environment.

It is possible to limit the amount of recorded data by using a capture filter. This filter technique is used to investigate each incoming network packet and determine the further processing by exact criteria. By matching a given parameter the packet is stored, otherwise it is discarded. The use of capture filter is contentious, in law enforcement investigation the use of capture filter is denied. The risk to lose relevant packets is too great.

C. Analysis

In this offline phase the captured data is examined and checked in detail. Different software tools help the investigator to reduce the amount of data, filter relevant information and reassemble the transferred data.

IV. Network virtualization

Like VMs are virtual implementation of computer systems running on physical systems, virtual networks are implementations of physical networks running on underlying devices.

[7] defines virtual networks as

> *A networking environment supports network virtualization if it allows coexistence of multiple virtual networks on the same physical substrate. [...] Essentially, a virtual network is a subset of the underlying physical network resources.*

The beginning of network virtualization was made with techniques like virtual private networks (VPN) or virtual local area networks (VLAN). But the need for a higher virtualization rate gets more important with the evolution of cloud computing environments. In this environment, the creation, deletion or motion of VMs is easier than in current ICT-environment. But the limitations of the networks thwart a comprehensive benefit of the virtualization.

Customers desire a secure environment, running their VMs in an isolated network, without foreign VMs inside their setting, however with a connection to external networks. Their environment should be maintained by themselves, without any administration tasks of the provider.

With the virtualization of networks this process is getting more comfortable. VMs are connected to a virtual network, which is provided within the virtual environment. The user does not need to know, how the underlying network is implemented. Administration, Operations and Maintenance of this underlying network is hidden by providing a logical overlay network.

An implementation with VLANs[4] fulfills the requirements of security and isolation, but the limitation of 4096 logical networks impede an implementation in bigger networks. These limitations led to an evolution of new network protocols, which intend to increase the number of possible participants or to hide internal address schemes of the logical networks. Protocols like QinQ [22], Stateless Transport Tunneling (STT) [23] or Virtual eXtended LAN (VXLAN)[24] encapsulate the internal network protocols and transfer the new protocols to a given endpoint.

Other implementations of virtual networks are SDN and NFV, which are described as follows.

A. Software Defined Networks

SDN is a new paradigm of network infrastructure, which allows the network administrator to manage the whole network at an abstract level. The need of configuration single devices is replaced by a central configuration of the whole network as one.

To achieve this, SDN separates the control and the data plane of a switch and shifts the control plane to an external device, namely the controller. The data plane resides on the switch and still forwards the network packets to the correct interfaces. The decision of the forwarding process is made by the decoupled forwarding plane on the controller. A special communication channel is established between controller and switch to transfer the relevant forwarding decision. The most frequently used network protocol to transfer this information is *OpenFlow* [11].

Another fundamental feature of SDN are open interfaces, which enable an interchangeability of network devices of different vendors. In the absence of open interfaces, the advantages would be taken away, which would lead to a proprietary and inflexible environment again.

Based on the open interfaces another key feature of SDN is possible. With the shift to SDN the programmability of the network becomes significant. By decoupling all forwarding planes and shifting them into the controller the network can be administrated by changing the configuration at only one place. This eliminates the necessity of configuring different devices because of a network change. These changes might emerge by the installation of new servers, creation of new network scopes or just by moving services from one to another machine. The network administrator currently has to

[2] e. g. in case of a system overload, port mirrors are mostly established with a lower priority

[3] also called 10GE (10 Gigabit Ethernet)

[4] IEEE 802.1q

adapt logical assignment from ports to VLAN-IDs, rewrite firewall rules or add new routing entries. The abstraction of management enables the configuration at the controller without touching every device. Neither the configuration of single devices nor the vendor-specific configuration remains.

B. Network Function Virtualization

Network Function Virtualization (NFV) describes a new approach to decouple software implementation from the underlying hardware of network devices [13]. A group at the European Telecommunications Standards Institute (ETSI) [25] is working on standardization of these implementations.

By decoupling the software from the hardware it is possible to transfer the software in virtual appliances, which are capable to run on commodity hardware. Thus a network function like a firewall, a router or a switch can be provided on demand at nearly any position in the network infrastructure. Additional benefits are cost reduction by omitting hardware solutions and better service provisioning by scaling up the performance based on the network state.

V. Problems

This section describes the new problems, which arise in NFI of virtual networks. Not all problems are new in other branches of digital investigation, but occur now for the first time in NFI. Aspects like multi-tenancy or the customization of the leased environment complicate computer forensic investigation equally. The evolution of network virtualization led to a change of network forensics. Current workflows as described in section III are now error-prone, with lower success rate of intercepting relevant data and a more difficult analysis.

We divide the arising problems in three sections, the classification guides on the phases of the network investigation. Problems that effect the online-phase of the investigation are summarized in the section *online*, and problems that affect the analyzing phase are summarized in the section *offline*. Problems that do not touch any of these phases are classified as *organizational*.

A. Online

Online problems have a repercussions on the capturing or the recording of the network packets. As discussed later, the problems are mostly hardware-based or depend on the used cloud-environment.

1) virtual NIC: DCs in the past without virtualization techniques provided their customer the requested services on physical servers. In case of digital investigations the suspect system has to be identified to demount the hard disks, perform memory forensics or capture the network traffic. While each server only hosts one or two services stationary without automatically changing this hosting, the whole environment is still rigid. Thereby the identification of a relevant server is easy and thus, the identification of the correct network card was easy to realize. The dogma of ”one server - one port” was valid in this kind of implementation.

The development of high speed connections and the demand for high availability networks lead to a softer use of this dogma. E. g. link-aggregation provides the combination of two or more links acting as one logical interface to accelerate the connection or increase the availability. A NFI has to adapt only the capture process to get all transmitted packets whichever interface is used.

The growing of virtual networks breaks up with this dogma. Current hosting server provide more and more VMs, each of them at least connected to a virtual network. The connection to a virtual network is established by a virtual network interface card (vNIC) which is nearly congruent with a physical network interface card (pNIC) on hardware systems. Each vNIC is equipped with a mac-address and an IP address which are valid inside the virtual network. Depending on the virtualization technique different implementations of vNICs like *Linux bridging* or *MACVTAP* are possible [9]. The vNICs distinguish by name and internal id, which are assigned by the cloud controller (CC) or hypervisor.

All incoming or outgoing packets of all locally hosted VMs to external networks are consolidated with a few pNICs. Thereby the capture of the relevant data on a pNIC is not promising any more. The well-tried hardware solutions like a TAP or a bridge are not applicable anymore with the vNIC of the suspicious VM. The lack of a usable pNIC, which only transmit traffic of the suspected VM impede a hardware solution.

2) Duplicate mac-addresses: Mac-addresses are designed as a unique identifier for the NIC, used for the communication in ethernet based networks. The mac-address is divided in two parts, the first 24 bits represent the vendor, the last 24 bits are a contiguous value of the production process of the vendor. Thus, a mac-address is a common filter criterion in NFI to reduce the amount of data and extract relevant information faster.

The use of hypervisors like *KVM* or *Xen* offers the opportunity to determine the mac-address of a given VM by themselves. By separating the VMs into isolated networks, the mac-address does not have to be unique any more in the virtual environment. The implementation of new network protocols as discussed prior amplify the ambiguous use of mac-addresses. Protocols like VXLAN get rid off this restriction and demand the uniqueness of the mac-address limited inside the VXLAN-segment.

This impede the NFI in virtual networks. Filtering the captured data for mac-addresses is not suitable any more. The captured data might contain, depending of the position of the capture implementation in the network infrastructure, the same mac-address more than one time which renders the identification of the suspected system impossible.

3) Overlapping IP addresses: Mostly a digital investigation in networks starts with a given IP address, which has been found in logfiles or by analyzing the communication of other suspects. This IP address is normally a public IP address, which lead to a provider, who has assigned the address to his customer. In virtual environments the CSP has to record this mapping of public IP addresses to the local ones. The result of the conversion leads to a customer and the assigned

private subnet. A further capture process to this local subnet is not suitable, because of the ambiguous use of the internal IP addresses.

Not only mac-addresses are ambiguous in virtual environments. Even IP addresses and the used network scopes for customers may overlap. Overlapping network addresses are used to improve the flexibility in assigning the IP addresses dynamically [26]. The overlapping IP addresses are limited to internal used private addresses [27], which are not routed in the internet. Public IP addresses are still unique inside a virtual environment to guarantee the communication with the internet.

If the provider implements overlapping IP addresses in his environment, he has to provide additional systems which translate the local IP addresses in unique addresses and back. This can be done e. g. with VXLAN and their virtual tunneling endpoints (VTEP), which encapsulate the internal data and transfer it within a virtual tunnel to the communication partner. *Network Virtualization using Generic Router Encapsulation* (NVGRE) or *Generic Network Virtualization Encapsulation* (GENEVE) offer related techniques to tunnel network data and hiding internal addresses schemes.

But the investigation of cyber-crime is not limited to VMs with connection to public networks. VMs with an internal connection or within a VPN may transfer relevant data which are important to intercept. Thus the analysis of network traffic being transmitted or received by these systems is essential.

The identification of these suspicious systems is difficult without any further information. The knowledge about the used internal private IP addresses is not sufficient any more.

B. Offline

The offline phase defines all necessary steps to analyze the captured data and to extract the relevant information. Problems in this section effect the analysis, either by complicating or by preventing the examination. These problems are software-related, which means, that they are independent from deployed hardware or the organizational structure of the environment. Nevertheless they are not independent from the real implementation in the cloud-environment.

1) Software support: The amount of data captured in a NFI is mostly enormous, either the definition of suitable filter is impossible or the implementation of these filter might result in a possible data loss. So the analysis of this data is time-consuming and impossible without usable software tools and automated processes. But the absence of eligible software, which supports the digital investigation in cloud environments impedes this examination. The market of network forensic software is exhausting, from short scripts to software solutions with thousand of programming lines. Some of these tools are open-source, others are commercial. [16] and [21] present an overview of different tools used in NFI.

We analyzed different network forensic tools and their capability to handle new network protocols or huge input files. We assumed that an import of files with at least 1 GB taking more than 45 seconds is not suitable for NFI. We performed the import on an specialized investigation PC with an Intel Core i7 cpu, 32 GB RAM and a 250 GB SSD installed with Debian 8. Table I lists a short overview of the results.

The analyzed tools were *Wireshark* [28], *Bro* [29] or *Moloch* [30] as open-source or *Network Miner*[5] as commercial tools.

Software	VXLAN	OpenFlow	Import of huge data
Wireshark	x	x	-
Moloch	-	-	x
Network Miner	-	-	-
Bro	-	-	x

TABLE I
CAPABILITIES OF NETWORK FORENSIC SOFTWARE

Only *wireshark* is capable to handle the analyzed protocols. But it fails in analyzing huge amounts of network traffic and big capture files, which arise in virtual networks. Tools like *Bro* or *Moloch* are able to handle big files, but they fail in analyzing the network protocols.

Thus no software tool is capable to handle all requirements of a NFI in a virtual network.

2) Protocols: New protocols like VXLAN, GENEVE or NVGRE offer new possibilities, expand the flexibility and are the basis for a virtual network. The main purpose of these protocols is to get rid of limitations of older protocols like VLAN or Cisco ISL.

The usage of these protocols does not complicate the online phase of a network investigation. The capture and the recording of the network data is independent from the internally used protocols. The analysis depends on the data found.

The new protocols use mostly encapsulation techniques to hide internal information and transfer the data to the communication endpoint. GENEVE encapsulates the payload by prefixing a GENEVE and a UDP-Header. VXLAN implements a new VXLAN-Header with a 24bit tag called *VNI* (VXLAN network identifier) and a UDP-Header, too. NVGRE uses GRE to encapsulate the previous frame.

The encapsulation complicates the further analysis of the captured data. As described prior the lack of suitable software tools to analyze the captured data is still an existing problem.

To extract the transferred information the software has to decapsulate each protocol information by itself, without losing any information given on this layer. Depending on the rate of encapsulation, different network information are extracted. Without deeper knowledge of the used virtual infrastructure, the decision which IP- or mac-addresses are relevant is impossible. Depending on the capture position in the network infrastructure either the addresses of the outer header or the inner header are relevant, the extraction and filtering of internal addresses requires a decapsulation of all outer header.

C. Organizational

We define organizational problems as independent of the used environment and the deployed hard- and software. Prob-

[5]Details can be found at *http://www.tamos.com/products/netresident/*

lems in this section depend on the inherent behavior of cloud-computing and exist in every environment with different customers and a great amount of hardware devices.

1) Multitenancy: The virtual environment enables a highly dynamic and scalable provision of VMs to lots of customers at the same time. Current servers are able to host more than 100 running VMs concurrently, each under administration of an other customer. These different VMs share the same hardware components like hard disks, memory or network card. Sharing those relevant components still complicates the digital investigation, especially the extraction of the relevant user data is hardly feasible [3].

The transferred network data of the hosting server contains communication data of all running VMs, possible internal system data or backup and imaging information, depending on the configuration of the underlay network. A capture process may obtain lots of odd data without further arrangements.

Whereas the recording of the system and backup data delivers unneeded network packets to the capture file, the storage of non-involved user data is mostly incompatible with the local legislation.

In addition to this the storage of unnecessary data leads to an overcrowded capture-file, which complicates the further analysis. As described prior the use of a capture filter to decrease the amount of data by prevention of storing these data might be critical.

2) Multitude of controllers: In SDN-based networks the controller is the central device, which supervises the complete traffic control and all decision according the network. A multitude of controllers exist on the market, differentiated by the platform, the vendor, the API or the language support [31]. Our analysis of vendors, projects and implementations reveals 28 open-source controller and 64 commercial SDN controller and vendors. This high number with different implementations reveals the unworkable multitude of controllers.

The current state of these controllers is extremely diverging, each implementations might be developed with an other model or purpose.

[32] describes seven controllers, table II displays exemplary distinctions.

Controller	**DescrIPtion**
NOX	Developed in C++, multi-threaded
Pox	Python, single-threaded
Floodlight	Java using Netty framework, multi-threaded
Beacon	Java using OSGi, multi-threaded
Ryu	Python-based, using *gevent*-wrapper
MuL	C, multi-threaded
Maestro	Java, multi-threaded

TABLE II
COMPARISON OF SDN CONTROLLER

This diversity of programming languages and runtime environments complicate the implementation of a ubiquitous approach, which is valid for NFI. Either the developed solution has to be adapted to the current environment or a very abstract approach has to be implemented. The commercial controllers complicate an approach furthermore. Without the knowledge of communication protocols or an existing API every digital investigation will fail or will only slowly be ready.

3) Migration: VMs are hosted on physical hosts, which have a high capacity of performance. In DCs lots of physical server host the different VMs, which are under control of a cloud-management platform (CMP) like OpenStack, OpenNebula or CloudStack. All of these implementations use a central instance (cloud controller (CC)) to control, monitor and manage the VMs. Compute nodes (CN) provide the runtime environment of the VMs with locally installed hypervisors.

The load in a virtual environment is highly dynamic, with VMs consuming less cpu or storage capacity and others with a high consumption rate. To guarantee an average system load, the CC monitors in combination with the CN the available resource pool to recognize peaks or increasing system loads. If the load of a CN reaches a critical point, the CC is able to move particular VMs to an other, less busy CN. Not only the CC, even the customer is able to activate such a migration.

There are two main techniques to migrate the running VM, which are classified as postcopy or precopy migration [33]. Independent of the implementation, a migration of the suspected VM results in a failure of the established capture process. A migration affects either a hosting server inside the same server rack, another server rack inside the same DC or another DC. The election of the new hosting server is unpredictable, therefore each hosting system provides VMs of varying customers. The flexible deployment of the VMs do not facilitate the predicting of the affected server.

Each of the new target systems needs a reconfiguration of the capture process, depending on the new parameters which are valid after restarting the suspicious VM. The time to reconfigure the new capture process is critical, the longer the configuration takes, the longer the capture process stays invalid and transmitted data will not be captured. This loss of possible packets make the further analysis complex, error-prone and may result in a wrong conclusion.

The movement of suspicious systems exist in webhosting solutions since the early 2000, when the era of commercial webhosting began. At these days a webserver hosted a lot of different domains under only one IP address. If the customer ordered an other hosting-solution, like more free-space or more emailadresses, the provider moves the relevant data to another host and changes the current network information like IP address or firewall policy. An installed capture process geared towards this customer host could be changed at the same time. The downtime during the manual migration enables the simultaneous reconfiguration of the capture process. The possibility to restart the suspicious server after creating the capture process ensures a complete capture file and a simplified analysis.

Nowadays a migration of a VM is performed without any additional steps inside the virtual environment, no matter if the command to migrate was given by the customer, the administrator or by the cloud environment itself. The result

will be the same, an installed capture process will fail because the given parameter has changed.

4) Customizable environment: One of the main purposes of virtual networks is to decrease the amount of administrative work and increase the dynamic in the network. The virtual networks provided to the customer relieve the provider of maintaining the user network. The CSP has only to guarantee a working underlay network.

The virtual network of a particular user is maintained by himself, which includes all aspects like addressing, internal routing, firewall policies or an implementation of VPN.

A NFI of a suspicious user is getting more and more complex depending on the customization of the network. A running capture process might fail, if the user implements a new routing policy, activate a separate vpn-connection or just migrates a particular VM to another hosting server.

VI. Conclusion

With the increasing demand of flexible ICT-environment, the use of virtual networks gains in importance. Concurrent to this development, there is an increasing emphasis on reliable digital examination methods, in particular the network forensic investigation. In this paper, we have summarized the arising problems for digital investigation in virtual networks. We separated the problems in three parts which are organizational, online and offline. This chapter summarizes the problems and provides possible solutions, after that we present our future research.

A. Summary

The evolution of network virtualization led to a highly dynamic and flexible infrastructure offering two views on the network. The underlying physical network is the basis for lots of possible overlay networks. These overlay networks provide a virtual implementation of independent networks. The administration of these virtual networks is separated from the administration of the underlay network and can be done by the customer of the virtual environment. Each virtual network is isolated from others and might be customized without additional work of the provider.

A higher rate of abstraction is reached with SDN and NFV. SDN implements a central control of the network and provides the ability to program the network infrastructure. This led to a highly flexible environment, which may act mostly autonomous without any further interaction of the network administrator.

NFV implements given network services in virtual appliances to improve the flexibility in the network by providing the needed services on demand.

The digital investigation in this virtual networks is complex and error-prone. Aspects like migration of suspicious VMs or vNICs complicate the capture process, encapsulating protocols like VXLAN impede the further analysis. Different CMPs compound the development of an available approach.

Forensic investigation in a SDN is a complex task, the control of the network by the CC led to an mostly unpredictable behavior of the network and a suspicious VM.

The deployment of NFV enables a customizable environment with high user influence of the installed virtual network infrastructure. The CSP is no longer concerned for any internal changes in the virtual network. An implementation of a new VPN by activating a separate virtual appliance in the local network might change the whole communication mode in the virtual network and abort the capture process.

Some problems referring to local jurisdiction might be evaded by a correct positioning of the capture technique. The closer to the source the capture technique is installed, the less the multi-tenancy gets critical.

B. Future Work

Digital investigation in virtual networks is still a not yet fully researched, rather a largely uncharted topic in digital forensic.

Network virtualization makes the network forensic process inside a cloud environment more complex, because none of the traditional techniques seem to be successful. VNICs impede the data recording, new network protocols like VXLAN complicate the analysis.

A promising approach is to implement a capture process on the target vNIC. But as discussed in this paper each operational solution might fail after a period of time. Shutting down and restarting the VM changes the internal id of the vNIC, which aborts the running capture process. Migrations or customized environments abort a running capture process, too. So an additional monitoring process is needed to recognize each change in the controlled network infrastructure.

Our focus in future work is to create a process of capturing network data in virtual environments, including all network data arising of a suspicious VM, even when this VM is stopped or migrated, or the virtual environment is changed by the customer. Regarding all of these aspects, NFI is again an important source of information to clarify cyber-crime cases.

References

[1] P. Mell and T. Grance, "The nist definition of cloud computing," *National Institute for Standards and Technology*, vol. Special Publication 800-145, 2011.

[2] K. Ruan and J. Carthy, "Cloud forensic maturity model," in *ICDF2C*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 114. Springer, 2012, pp. 22–41.

[3] S. Zawoad and R. Hasan, "Cloud forensics: A meta-study of challenges, approaches, and open problems," *CoRR*, 2013.

[4] K. Ruan, J. Carthy, T. Kechadi, and M. Crosbie, "Cloud forensics: An overview," 2011.

[5] J. Dykstra and A. T. Sherman, "Design and implementation of frost: Digital forensic tools for the openstack cloud computing platform," *Digital Investigation*, vol. 10, pp. S87–S95, 2013.

[6] N. Chowdhury and R. Boutaba, "Network virtualization: state of the art and research challenges," *Communications Magazine, IEEE*, vol. 47, no. 7, pp. 20–26, 2009.

[7] N. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks*, vol. 54, no. 5, pp. 862–876, 2010.

[8] M. F. Bari, R. Boutaba, R. Esteves, L. Z. Granville, M. Podlesny, M. G. Rabbani, Q. Zhang, and M. F. Zhani, "Data center network virtualization: A survey," *Communications Surveys & Tutorials, IEEE*, vol. 15, no. 2, pp. 909–928, 2013.

[9] R. Jain and S. Paul, "Network virtualization and software defined networking for cloud computing: a survey," *Communications Magazine, IEEE*, vol. 51, no. 11, pp. 24–31, 2013.

[10] M. Jarschel, T. Zinner, T. Hoßfeld, P. Tran-Gia, and W. Kellerer, "Interfaces, attributes, and use cases: A compass for sdn," *Communications Magazine, IEEE*, vol. 52, no. 6, pp. 210–217, 2014.

[11] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.

[12] A. Bates, K. Butler, A. Haeberlen, M. Sherr, and W. Zhou, "Let SDN be your eyes: Secure forensics in data center networks," in *Proceedings of the NDSS Workshop on Security of Emerging Network Technologies (SENT'14)*, Feb. 2014.

[13] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *Communications Magazine, IEEE*, vol. 53, no. 2, pp. 90–97, 2015.

[14] R. Vilalta, R. M. noz, A. Mayoral, R. Casellas, R. Martínez, V. López, and D. López, "Transport network function virtualization," *J. Lightwave Technol.*, vol. 33, no. 8, pp. 1557–1564, Apr 2015. [Online]. Available: http://jlt.osa.org/abstract.cfm?URI=jlt-33-8-1557

[15] B. Pfaff, J. Pettit, T. Koponen, E. Jackson, A. Zhou, J. Rajahalme, J. Gross, A. Wang, J. Stringer, P. Shelar *et al.*, "The design and implementation of open vswitch," in *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*, 2015, pp. 117–130.

[16] R. Hunt and S. Zeadally, "Network forensics: An analysis of techniques, tools, and trends." *IEEE Computer*, vol. 45, no. 12, pp. 36–43, 2012.

[17] N. Meghanathan, S. R. Allam, and L. A. Moore, "Tools and techniques for network forensics," 2009. [Online]. Available: http://arxiv.org/pdf/1004.0570

[18] K. Shanmugasundaram, H. Brönnimann, and N. Memon, "Integrating digital forensics in network infrastructures," in *Advances in Digital Forensics*, ser. IFIP — The International Federation for Information Processing, M. Pollitt and S. Shenoi, Eds. Boston: Kluwer Academic Publishers, 2006, vol. 194, pp. 127–140.

[19] D. Spiekermann, T. Eggendorfer, and J. Keller, "Using network data to improve digital investigation in cloud computing environments," in *High Performance Computing & Simulation (HPCS), 2015 International Conference on*. IEEE, 2015, pp. 98–105.

[20] G. Palmer and M. Corporation, "A road map for digital forensic research," Digital Forensic Research Workshop, Utica, New York, Tech. Rep. 1, August 2001. [Online]. Available: http://isis.poly.edu/kulesh/forensics/docs/DFRWS_RM_Final.pdf

[21] E. S. Pilli, R. C. Joshi, and R. Niyogi, "Network forensic frameworks: Survey and research challenges," *Digital Investigation*, vol. 7, no. 1, pp. 14–27, 2010.

[22] IEEE Computer Society, "Virtual bridged local area networks amendment 4: Provider bridges," IEEE Standard for Local and metropolitan area networks, Tech. Rep., 2005.

[23] E. B. Davie and J. Gross, "A stateless transport tunneling protocol for network virtualization," Online, April 2014.

[24] M. Mahalingam, D. Dutt, K. Duda, P. Agarwal, L. Kreeger, T. Sridhar, M. Bursell, and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks," RFC 7348 (Informational), Internet Engineering Task Force, Aug. 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7348.txt

[25] ETSI, "Network functions virtualisation - introductory white paper," *SDN and OpenFlow World Congress*, vol. 1, October 2012. [Online]. Available: https://portal.etsi.org/NFV/NFV_White_Paper.pdf

[26] Cisco Systems Inc., "Ip overlapping address pools," Online, 12 2012. [Online]. Available: http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipaddr_ipv4/configuration/xe-3s/asr1000/IPv4-xe-3s-asr1000-book/IP-overlap-addr-pools.pdf

[27] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address Allocation for Private Internets," RFC 1918 (Best Current Practice), Internet Engineering Task Force, Feb. 1996, updated by RFC 6761. [Online]. Available: http://www.ietf.org/rfc/rfc1918.txt

[28] G. Combs *et al.*, "Wireshark," *Web page: http://www. wireshark. org/last modified*, pp. 12–02, 2007.

[29] V. Paxson, "Bro: a system for detecting network intruders in real-time," *Computer networks*, vol. 31, no. 23, pp. 2435–2463, 1999.

[30] A. Wick and E. Miller, "Moloch," *Web page: http://molo.ch*, pp. 12–02, 2013.

[31] R. Khondoker, A. Zaalouk, R. Marx, and K. Bayarou, "Feature-based comparison and selection of software defined networking (sdn) controllers," in *Computer Applications and Information Systems (WCCAIS), 2014 World Congress on*. IEEE, 2014, pp. 1–7.

[32] A. Shalimov, D. Zuikov, D. Zimarina, V. Pashkov, and R. Smeliansky, "Advanced study of sdn/openflow controllers," in *Proceedings of the 9th Central & Eastern European Software Engineering Conference in Russia*. ACM, 2013, p. 1.

[33] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in *Proceedings of the 2Nd Conference on Symposium on Networked Systems Design & Implementation - Volume 2*, ser. NSDI'05. Berkeley, CA, USA: USENIX Association, 2005, pp. 273–286.