Paul Müller, Bernhard Neumair, Helmut Raiser,
Gabi Dreo Rodosek (Hrsg.)

10. DFN-Forum Kommunikationstechnologien

30. – 31. Mai 2017
Berlin

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume 271

ISBN 978-3-88579-665-7
ISSN 1617-5468

Volume Editors
Titel Vorname Nachname
Universität
Adresse, Land
Email

Series Editorial Board
Heinrich C. Mayr, Alpen-Adria-Universität Klagenfurt, Austria
(Chairman, mayr@ifit.uni-klu.ac.at)
Dieter Fellner, Technische Universität Darmstadt, Germany
Ulrich Flegel, Infineon, Germany
Ulrich Frank, Universität Duisburg-Essen, Germany
Andreas Thor, HFT Leipzig, Germany
Michael Goedicke, Universität Duisburg-Essen, Germany
Ralf Hofestädt, Universität Bielefeld, Germany
Michael Koch, Universität der Bundeswehr München, Germany
Axel Lehmann, Universität der Bundeswehr München, Germany
Thomas Roth-Berghofer, University of West London, Great Britain
Peter Sanders, Karlsruher Institut für Technologie (KIT), Germany
Torsten Brinda, Universität Duisburg-Essen, Germany
Ingo Timm, Universität Trier, Germany
Karin Vosseberg, Hochschule Bremerhaven, Germany
Maria Wimmer, Universität Koblenz-Landau, Germany

Dissertations
Steffen Hölldobler, Technische Universität Dresden, Germany
Thematics
Andreas Oberweis, Karlsruher Institut für Technologie (KIT), Germany

© Gesellschaft für Informatik, Bonn 2017
printed by Köllen Druck+Verlag GmbH, Bonn

This book is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

Vorwort

Der DFN-Verein ist seit seiner Gründung dafür bekannt, neueste Netztechnologien und innovative netznahe Systeme anwendungsorientiert zu erforschen und einzusetzen, um damit die Leistungen für seine Mitglieder laufend zu erneuern und zu optimieren. Beispiele dafür sind die aktuelle Plattform des Wissenschaftsnetzes X-WiN und Dienstleistungen für Forschung und Lehre wie die DFN-PKI, DFN-AAI und föderierte Cloud-Dienste. Um diese Technologien einerseits selbst mit zu gestalten und andererseits frühzeitig die eigenen Forschungsergebnisse mit den Entwicklungen anderer Wissenschaftler abzugleichen, veranstaltet der DFN-Verein seit vielen Jahren wissenschaftliche Tagungen zu Netztechnologien. Mit den Zentren für Kommunikation und Informationsverarbeitung in Forschung und Lehre e.V. (ZKI) und der Gesellschaft für Informatik e.V. (GI) gibt es in diesem Bereich eine langjährige und fruchtbare Zusammenarbeit.

Das DFN-Forum Kommunikationstechnologien „Verteilte Systeme im Wissenschaftsbereich“ im Jahr 2017 ist bereits die 10. Veranstaltung in dieser Reihe. Es setzt die Tradition der neun sehr erfolgreichen Vorgänger in Kaiserslautern, München, Konstanz, Bonn, Regensburg, Erlangen, Fulda, Lübeck und Rostock fort. Als Jubiläumsveranstaltung wird das diesjährige Forum vom DFN-Verein und der Freien Universität Berlin gemeinsam mit dem ZKI e.V. und der GI am 30. und 31. Mai am Sitz des DFN-Vereins in Berlin veranstaltet. Wie seine Vorgänger soll es eine Plattform zur Darstellung und Diskussion neuer Forschungs- und Entwicklungsergebnisse aus dem Bereich TK/IT darstellen. Das Forum dient dem Erfahrungsaustausch zwischen Wissenschaftlern und Praktikern aus Hochschulen, Großforschungseinrichtungen und Industrie.

Aus den eingereichten Beiträgen konnte ein hochwertiges und aktuelles Programm zusammengestellt werden, das sich neben Fragen von Netzsicherheit und IT Service Management auch mit e-Infrastrukturen für Forschung, Lehre und Forschungsdatenmanagement befasst. Eingeladene Beiträge runden das Programm ab. So wird im Jubiläumsjahr der DFN-Verein über Herausforderungen und Perspektiven bei Entwicklung und Betrieb eines großen Forschungsnetzes berichten. Das auch dieses Jahr hochaktuelle Thema Sicherheit im Internet wird im Beitrag „DDoS 3.0: the day the internet stopped working“ beleuchtet. Abgerundet wird das Programm durch einen Beitrag zur Ausfallsicherheit moderner SW-gesteuerter Systeme aus industrieller Sicht.

Wir möchten uns bei den Autoren für alle eingereichten Beiträge, beim Programmkomitee für die Auswahl der Beiträge und die Zusammenstellung des Programms, bei den Mitarbeiterinnen und Mitarbeitern der Geschäftsstelle des DFN-Vereins für die Organisation und beim Gastgeber für die Unterstützung des Forums sowie die Gastfreundschaft bedanken. Allen Teilnehmern wünschen wir für die Veranstaltung interessante Vorträge und fruchtbare Diskussionen.

Berlin, Mai 2017

Gabi Dreo Rodosek
Paul Müller
Bernhard Neumair
Helmut Reiser

Programmkomitee

Rainer Bockholt, Universität Bonn
Alexander Clemm, Huawei USA
Gabriele Dobler, Universität Erlangen-Nürnberg
Gabi Dreo Rodosek (Co-Chair), Universität der Bundeswehr München
Thomas Eickermann, Forschungszentrum Jülich
Alfred Geiger, T-Systems SfR
Wolfgang Gentzsch, The UberCloud
Andreas Hanemann, Fachhochschule Lübeck
Peter Klingebiel, Hochschule Fulda
Ulrich Lang, Universität zu Köln
Paul Müller (Co-Chair), Technische Universität Kaiserslautern
Bernhard Neumair (Co-Chair), KIT
Gerhard Peter, Hochschule Heilbronn
Christa Radloff, Universität Rostock
Helmut Reiser (Co-Chair), LRZ München
Sebastian Rieger, Hochschule Fulda
Harald Roelle, Siemens AG
Peter Schirmbacher, Humboldt-Universität zu Berlin
Uwe Schwiegelshohn, TU Dortmund
Manfred Seedig, Universität Kassel
Marcel Waldvogel, Universität Konstanz
René Wies, BMW Group
Martin Wimmer, Universität Regensburg

Inhaltsverzeichnis

Sicherheit

Prozessorientiertes IT-Service Management

eInfrastrukturen für Forschung, Lehre und Forschungsdatenmanagement

Sicherheit

Müller, Neumair, Reiser, Dreo Rodosek (Hrsg).: 10. DFN-Forum Kommunikationstechnologien,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 11

A new Attack Composition for Network Security

Frank Beer[1], Tim Hofer[1], David Karimi[1], and Ulrich Bühler[1]

Abstract: As the current cyber threat landscape is becoming more depressing, sophisticated intrusion detection systems must evolve to protect network infrastructures efficiently. Building such a detector is highly data-driven and requires quality datasets to evaluate different phases in both the development and deployment process. However, finding publicly available captures with a ground truth is challenging. Most existing datasets focus on very specific subjects such as botnet, flooding, or brute-force traffic rather than providing a broad arsenal of different attack vectors threatening today's networks. This work addresses this gap by introducing a new attack composition comprising a multitude of classic as well as state-of-the-art attacks. The dataset embrace rich and untreated packet traces including payload, collected log events, and a detailed ground truth. Intitial results reveal the proposed captures complement existing traces and provide a sound base for various mining applications in the field of network security research.

Keywords: Attack dataset, data sharing, ground truth, intrusion detection, network security

1 Introduction

The advances of today's cyber attacks against network infrastructures are versatile and alarming. Hence, there is a huge demand for trustworthy remedies. Research in this direction frequently utilize supervised machine learning (see [BG15]) to build sophisticated network intrusion detection systems (IDSs). The downside of these approaches is the inherent necessity of quality datasets for training and validation purposes. Ideally, such a dataset should represent realistic traffic and cover a multitude of classic as well as state-of-the-art attack types providing legitimate traffic from the underlying network of interest. It further should exhibit a ground truth (GT) annotating traffic with class labels to indicate malicious and benign instances. Moreover, the captures should embrace a fine-grained data format, which offers additional opportunities such as mining meaningful features to increase detection capabilities or to benchmark competing solutions that rely on different inputs including packet or flow data.

Finding a publicly available dataset fulfilling these requirments is a challenging task and a true concern in the network security community with respect to reproducibility and comparability [AB14]. However, several attempts have been made over the last decades: The most prominent datasets are the DARPA 98/99 traces [Li00a, Li00b, Ha01] and derived versions such as KDD-Cup 99[2], GureKDDcup [Pe08], and NSL-KDD [Ta09]. Despite their age, these are yet frequently utilized in the community although widely criticized due to design flaws and their inability to meet contemporary requirements with respect

[1] Network and Data Security Group (NDSec), University of Applied Sciences Fulda, Leipziger Strasse 123, D-36037 Fulda, {frank.beer, tim.hofer, david.karimi, u.buehler}@informatik.hs-fulda.de

[2] `http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html`

to traffic and state-of-the-art attack variants (e.g. [Mc00, MC03, Ta09]). Other more recent traces provide quality data, but pursue specific goals. L-Flows [Sp09] and SSH-DS [Ho14] mainly focus on brute-force attacks supplying highly aggregated flow data. Other novel captures such as CTU-13 [Ga14] and Booters-15 [Sa15] either concentrate on malware or denial-of-service (DoS) attacks. Considering multi-purpose intrusion detection, the only suitable dataset comprising a broader attack arsenal is the ISCX-2012 dataset [Sh12]. However, ISCX-2012 merely covers few malicious traffic compared to its benign counterpart, which makes its application for both training and testing a detector unfavorable particularly when working on flow level. Assembled datasets are another source of interest by fusing several independent traces. Two of these collections are ISOT [Sh11] and ISCX-Botnet [Be14] concentrating on botnet detection. ISOT incorporates real traces from LBNL/ICSI[3] and the Traffic Lab at Ericsson Reasearch [Sz08] with malware captures from the Honeynet project[4]. On the other side, ISCX-Botnet merges partial traffic from ISOT, ISCX-2012, and CTU-13 to compile a representative dataset. Despite the huge effort made over the years, this indicates that no general network intrusion dataset exists to date, which is in line with the opinion of other authors (e.g. [Ce11, Ga14, MBM15]).

Motivated by these findings and discussions among the community to share resulting traces, this paper introduces a new dataset, which already substantiated promising results in [BB17]. It is based on the following perceptions: According to the given requirements, both benign and illicit network traffic are essential developing and validating IDSs. However the legitimate part highly depends on the underlying infrastructure including influencing factors like software configurations and human interaction, which should be collected at the target domain particularly when considering anomaly-based intrusion detection. In this respect, we argue that malicious traffic is of high interest, because most existing captures focus on isolated attack types such as botnet, brute-force, or flooding, which is very specific for a general assessment of an IDS. Therefore, we build a dataset primarily concentrating on attack data rather than legitimate traffic using state-of-the-art penetration testing suites, malware instances collected "in the wild", recently reported exploits, and classic tools. As this arsenal is very basic equipment for cyber criminals, we carefully incorporated it into well-defined scenarios reflecting realistic attack situations, which are applicable to most conventional network infrastructures. Furthermore, our dataset embrace untreated packet traces including payload and captured log events documented by a rich GT. Thus, it can be reused to salt legitimate traffic based on common strategies such as the overlay methodology (see [AH11]) supporting both the development and deployment process of IDSs.

The remainder is structured as follows: First, we introduce the proposed dataset in Section 2 by illustrating the underlying network infrastructure (Section 2.1), attack scenarios (Section 2.2) and a summary of evolved attack types (Section 2.3). Section 3 outlines obtained results comparing the dataset against other related captures (Section 3.1). Moreover, a qualitative analysis is provided applying our traces to a well-known IDS (Section 3.2). In Section 4, we conclude and frame future work.

[3] `http://www.icir.org/enterprise-tracing/`
[4] `http://www.honeynet.org/chapters/france/`

2 NDSec-1 Dataset

Based on the discussed absence of appropriate traces providing a broad range of different attacks, this section proposes a new dataset. In contrast to most other solutions, it contains very few background traffic, and thus serves as attack repository. Additionally, we attached importance to other practical aspects. The following principles were key to the design of the dataset, which we refer to as NDSec-1:

- Support of various attack types and variants
- Attacks wrapped around realistic scenarios
- Simple infrastructure to incorporate other traces
- Detailed GT based on bidirectional flow semantics
- Provide raw packet captures including log events

In what follows, we highlight network infrastructure (Section 2.1) and describe involved attack scenarios (Section 2.2). Section 2.3 summarizes the resulting attack distribution.

2.1 Network Infrastructure

To build up an appropriate infrastructure, we followed a conventional topology placed on a testbed[5] located at our campus network. It operated as hypervisor mimicking two subnets, i.e. a private network (company or organization) and the simulated Internet. Both subnets were seperated by an OpenWRT[6] router acting as NAT gateway with firewall capabilities for the private network. Only port 80 was open for ingress traffic to the internal web server. Additionally, we configured the router to forward real Internet requests to the campus network, while the simulated Internet contained prepared virtualized machines to serve as controlled infrastructure to most performed scenarios (e.g. email system, exploit kit, or bot master). The private side relied on a heterogenous set of workstations and machines based on both different Windows and Linux versions. Traffic was captured by a tcpdump[7] sensor inside the private network. Thus, only incoming and outgoing connections of that subnet were observed. The log event information (i.e. syslog and Windows event log) were collected locally at each host and extracted after each scenario completed. An overview of this simplified network infrastructure is depicted in Figure 1.

2.2 Attack Scenarios

Bring Your Own Device: The bring your own device (BYOD) phenomenon is a pragmatic mindset established by enterprises and organizations enabling employees and partners to

[5] Hypervisor: VMWare ESXi 6.0; Memory: 100 GByte; CPU: 2x 2.30 GHz Xeon (E5-2630); HDD: 4x 1 TByte (RAID 5); NIC: 2x Intel I350 (1 GBit)

[6] `https://openwrt.org/`

[7] `http://www.tcpdump.org/`

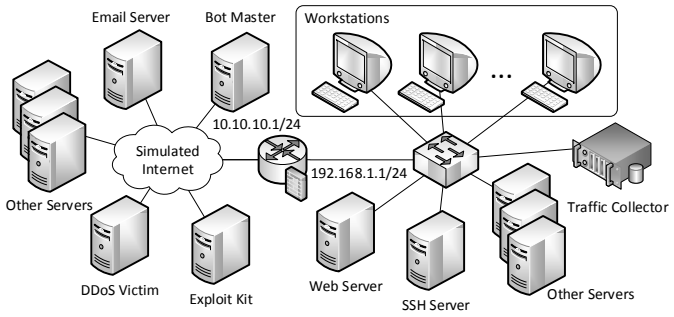


Fig. 1: Simplified topology of our virtualized network landscape: the simulated Internet (left) and the protected private network (right).

utilize personal hardware in-house. Despite all of its advantages, new security risks arises which may permit an attacker to act from the inside. To build such a scenario, a machine was placed to the protected network environment serving as a compromised BYOD. We had full access to the machine via an installed backdoor provided by Metasploit[8] using a binary Linux trojan. In order to study the unknown network, several reconnaissance activities were performed against the infrastructure and potential victims could be identified, i.e. an internal SSH server and a client system connected to both email and web server. We attacked the former by a dictionary brute-force uncovering login credentials. Client and email server were targeted combining ARP and DNS spoofing techniques to get between these machines. Thus, we could pretend to be the legitimate server forcing the client to disclose private information, which was exploited to steal valuable content from the victim's account. Another variant was pursued to get between host and web server utilizing ARP poisoning and an SSL proxy (see SSLsplit[9]). Hence, we could successfully hijack login credentials despite encrypted communication. To exfiltrate all obtained assets, we uploaded related data to an external FTP server using the compromised BYOD.

Watering Hole: It is common for enterprises or organizations to self-host services from within their infrastructure. In this scenario, an external intruder tried to compromise an internally hosted web server with the intention to infect a related group of hosts, i.e. a watering hole attack. First, a brute-force attack was performed against the front end of the web server followed by an SQL injection retrieving several logins and password hashes from the back-end database. Based on that gathered knowledge, we injected cross site scripts (XSS) to private pages of users found in the database. Hence, a small group of users could be targeted. The effect of XSS was a malicious redirect to an external exploit kit (i.e. Crimepack 3.1.3) that seeked for unpatched web browsers and vulnerable plugins on visiting hosts to inject specific malware. In our case, we used an unreported Internet Explorer exploit and ToxiCola as ransomware. Note, each instance of this specific malware contained customized binaries generated by the malware author right before its deployment complicating detection. Using this scenario, two hosts inside the protected network were

[8] `http://www.metasploit.com/`
[9] `https://www.roe.ch/SSLsplit`

successfully infected. ToxiCola encrypted several important local documents and reported back to a known server in the Internet.

Botnet: The rental of botnets operated by cyber crews is a lucrative business in the underground economy. Hence, these illicit infrastructures increasingly gain popularity. This trend is crucial for enterprises and organizations, because essentially any host of a legitimate network may serve as a bot, and thus has potentials to be part of a criminal act once infected. Citadel 1.3.5.1 as revised version of the well-known Zeus botnet was employed in this scenario. Based on a normal operating network, we infected three legitimate hosts with Citadel binaries. This task could be performed through conventional email spam using the recent vulnerabilities CVE-2015-2509 (Windows Media Center), CVE-2015-5122 (Flash Player), and a rogue download caused by XSS placed on a website in the simulated Internet. After the infection, all three bots communicated via HTTP to a prepared bot master. Among several traffic footprints between master and bots, we instructed all bots to download new commands. These contained hostile payload to perform a distributed DoS (DDoS) via SYN flooding to a single destination outside the network. Beside this successful attack, two of the bots stole local configuration files and transfered them to an external FTP server.

Attacks Without Specific Context: In this experiment, all attacks from the previous three scenarios were repeated without specific context. Additionally, we performed a number of other attacks. For instance, we used the tool Yersinia[10] to run DHCP starvation attacks, which exhausted the number of available IP addresses from a known DHCP server utilizing spoofed MAC addresses. HTTP floods were carried out using the Apache HTTP server benchmarking tool[11]. Additionally, we seeked for vulnerabilities with Nikto[12] and attacked an FTP service by the well-known THC Hydra tool[13]. Tsunami[14] was employed to perform DNS amplification attacks resulting in a DNS flooding attempt. Finally, we made use of the classic hping3[15] to send a high amount of UDP packets to specific target hosts in the network.

2.3 Dataset Summary

As a result of the processed scenarios inside the simplified testbed, numerous attack types and variants could be covered within NDSec-1, which can be aggregated to 12 categories. A summary of all instances along with the captured packets and byte distribution is illustrated in Table 1. Note, most of the rogue actions were performed manually. Particularly, the execution of the defined scenarios was carefully crafted to evade detection as much as possible. Thus, we believe the resulting dataset reflects realistic footprints. Each involved activity was labeled according to attack category using network flows. YAF[16] as

[10] `http://www.yersinia.net/`
[11] `http://httpd.apache.org/docs/2.4/programs/ab.html`
[12] `https://cirt.net/Nikto2/`
[13] `http://sectools.org/tool/hydra/`
[14] `https://www.infosec-ninjas.com/tsunami/`
[15] `http://www.hping.org/hping3.html`
[16] `https://tools.netsa.cert.org/yaf/`, version 2.8.4

sophisticated flow exporter was chosen for this reason using default timeout settings and bidirectional flow semantics. In order to share the outcome of this work, all raw packet traces, log files, and GT were published on our website[17]. Again, we would like to emphasize that NDSec-1 was designed to provide pure attack sequences and can be reused to season other legitimate network traces as suggested in [Ce11]. Since the captures are fine-grained, they may support the evaluation of existing or new detection approaches based on packet, flow, or log data.

Attacks	Packets	Bytes per packet
Botnet (Citadel)	5198	707.2697
HTTP brute-force	26093	495.1155
FTP brute-force	1530	63.0137
SSH brute-force	20873	179.0432
HTTP flooding	167238	115.6783
SYN flooding	890895	100.5449
UDP flooding	2275614	137.4647
Malware/exploits	8802	866.7635
Probe	21707	329.1476
Spoofing	1199	60.0083
SSL proxy	11602	776.8269
XSS/SQL injection	334	278.5419

Tab. 1: NDSec-1 attack and packet distribution

3 Results and Discussion

3.1 Comparative Study

This section evaluates six predominant and most related traces found in recent network security literature, i.e. Booters-15, CTU-13, ISCX-2012, ISCX-Botnet, L-Flows, and SSH-DS. We discuss the main characteristics and provide a comparison to NDSec-1.

The available format of a dataset is a key aspect for its applicability. While the majority of captures (i.e. ISCX-2012, ISCX-Botnet, Booters-15, CTU-13, and NDSec-1) provide rich packet traces (PCAP format) allowing detailed analysis on packet header or payload level, L-Flows and SSH-DS supply highly aggregated flow data. Based on this distinction, the latter sources can only be used for the emerging field of flow-based intrusion detection (e.g. [Sp10, Ho14, BB17]). In this context, the applicability is also determined by the involved attack categories, which either permit to assess the performance of IDSs towards a broad arsenal or to very specific attacks. ISCX-2012 and NDSec-1 include the largest attack repertory, while others target on certain instances (e.g. rogue botnet activities, brute-force attempts, or network stresser services). Moreover, the underlying environment is an essential property. Several of the examined datasets were captured “in the

[17] `http://www2.hs-fulda.de/NDSec/NDSec-1/`

wild" or under equivalent conditions providing most realistic traffic (i.e. L-Flows, SSH-DS, Booters-15, and CTU-13). However, traces with this characteristic usually embrace sensitive information raising privacy concerns. Therefore, some of these evaluated traces have been provided on flow level, were anonymized or sanitized such that certain information in these datasets are missed or become ineligible (e.g. anonymized IP addresses for SSH-DS or chopped off payload for CTU-13 on benign data). Note, we refer to raw data if the observed captures are not postprocessed. An alternative to circumvent privacy issues are synthetic datasets. These are recorded in a controlled environments (physical or virtual infrastructure), which does not necessarily mean they are inappropriate or less qualified to train or validate intrusion detection techniques. In fact, this type of datasets is gaining more attention in literature (e.g. [BWM08, Ce11, Be14]) and can produce realistic traffic footprints once the environment is setup properly. Traces comprising this property are ISCX-2012, ISCX-Botnet, and NDSec-1. The last characteristics deal with the underlying GT, which is another critical point for existing datasets [AB14]. Captures including malicious and legitimate traffic generally require a GT to apply supervised learning tasks, that may exist on different levels (e.g. per IP (ISCX-Botnet) or flow (ISCX-2012, L-Flows, CTU-13, and NDSec-1)). Simple annotations (i.e. the distinction between normal and hostile traffic) suffice binary classification problems, but detailed assessments of traces can only be achieved using rich labels. The latter is covered by L-Flows and NDSec-1 only. Note, SSH-DS and Booters-15 do not provide such a GT at all, because all involved data refer to malicious traffic. Table 2 depicts the discussed characteristics of all considered datasets.

Dataset	**Available format**			**Raw**	**Syn-**	**Involved attacks**								**GT**		
	PCAP	**Flow**	**Log**	**data**	**thetic**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**IP**	**Flow**	**Rich**
Booters-15 [Sa15]	✓	–	–	(✓)	–	–	–	–	✓	–	–	–	–	–	–	–
CTU-13 [Ga14]	✓	–	–	(✓)	–	✓	–	–	✓	✓	–	–	✓	–	✓	–
ISCX-2012 [Sh12]	✓	–	–	✓	✓	(✓)	✓	✓	✓	✓	–	✓	–	–	✓	–
ISCX-Botnet [Be14]	✓	–	–	(✓)	✓	✓	–	–	✓	✓	–	–	✓	✓	–	–
L-Flows [Sp09]	–	✓	✓	–	–	–	✓	–	–	✓	–	✓	–	–	✓	✓
NDSec-1	✓	–	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	✓	✓
SSH-DS [Ho14]	–	✓	✓	–	–	–	✓	–	–	–	–	–	–	–	–	–

✓=characteristic included, (✓)=characteristic partially included, −=characteristic not included; 1=botnet (command-and-control, fraud, fast flux, etc.), 2=brute-force, 3=other malware/exploit, 4=flooding, 5=probe, 6=spoofing, 7=web attack (SQL injection, XSS, etc.), 8=others (spam, SSL proxy, etc.)

Tab. 2: Comparison of most related intrusion detection datasets

These observations infer most datasets were captured for specific goals. L-Flows and SSH-DS comprise flow data and log event information, which can be correlated providing further insights to potential attack situations. However, the limiting factor on both captures is their format disallowing detailed analysis below flow level such as payload inspection. On the other hand, Booters-15, ISCX-Botnet, and CTU-13 provide rich traces. They concentrate either on botnet or DDoS-as-a-Service traffic ignoring other sophisticated attack vectors including brute-force or web attacks. These findings are crucial particularly when working towards a general IDS covering various attack types. The only two datasets comprising a wider range are ISCX-2012 and NDSec-1. Yet, ISCX-2012 neither covers log events nor a detailed ground truth, which refuse elaborate examinations per attack. Moreover, it does not cover frequently used spoofing attempts, man-in-the-middle attacks, or a real botnet as opposed to NDSec-1.

3.2 Insights to NDSec-1 Using Snort

As opposed to machine learning techniques which we partly examined in [BB17], this section briefly reports about the qualitative results running NDSec-1 against a signature-based detection engine. Snort[18] as state-of-the-art IDS was chosen for this reason employing default system settings, latest community rules, and emerging threats (ET) extension[19]. Note, all alerts per scenario (see Section 2.2) were mapped to the corresponding flow-based GT utilizing timestamps, IP addresses, and ports to diagnose matches.

Starting with the BYOD scenario, most of the probing activities remained undetected particularly for the interesting port range 0 to 1023. However, Snort discovered some vertical scans for specific ranges, i.e. 5800 to 5820 and 5900 to 5920. Additionally, some signatures caused alarms for database ports providing meaningful messages. The SSH dictionary attack was alerted on a periodical basis including the SSL proxy, while backdoor communication using Metasploit and the ARP spoofing attempts were not exposed by Snort. Considering normal traffic, some minor false alarms took place especially on the alert "PROTOCOL-DNS TMG Firewall Client long host entry exploit attempt", which occured in 7% of all benign cases. Since such a security gateway was not installed in our environment, this message was misleading for all involved scenarios. Within the watering hole scenario, a high amount of traffic was caused by the HTTP brute-force in order to take over the involved web server. No signature triggered on this activity. The same took place for the XSS placement and traffic produced by the injected ransomware. On the other side, the SQL injections as well as traffic induced by Crimepack could be detected. Eight different ET rules applied on the former such that 40% of the injection attempts were uncovered. Traffic caused by the latter was unmasked completely by an ET alarm designed for the Eleonore exploit kit. On the botnet scenario, exploited vulnerabilities (i.e. Windows Media Center and Flash Player) and command-and-control traffic could be identified with explicit ET signatures. However, involved DDoS and data theft remained concealed. Clearly, the exfiltration based on a legitimate FTP upload activity within a rogue context, which is difficult to expose using signature-based engines. Applying the last trace revealed similar results for attacks intersecting with the previous scenarios. Yet, flooding attacks based on HTTP and UDP basically remained undetected (hit rate $< 1\%$), while FTP brute-force and vulnerability scans could be identified in 33% and 63% of the cases.

This confirms several attack instances inside NDSec-1 could be safely uncovered by Snort using a default setup. Yet, some basic attacks were missed or hit only partially. Particularly, the latter attacks comprised a high traffic volume compared to other sure detections (see Table 1). Taking this factor into account, the overall classification on flow level revealed poor results in terms of conventional metrics such as hit rate or accuracy. Being aware that a more sophisticated configuration including enterprise ruleset certainly would boost Snort's accuracy, yet the obtained results using NDSec-1 in practice looked very promising. This indicates that incorporating penetration testing suites, recent malware instances and classic attack tools within realistic scenarios provide a sound base to support the development cycles of a new detector or to reveal weaknesses of deployed IDSs.

[18] `https://snort.org/`, version 2.9.9
[19] `https://rules.emergingthreats.net/`, version 8499

4 Conclusion and Future Work

Labeled data are essential for network security research in order to assess existing or new intrusion detection techniques. Most existing datasets are limited to certain attack types such as botnet, brute-force, or flooding. This fact is crucial particularly when examining intrusion detection systems towards a multitude of different attacks. Therefore quality traces are required comprising both classic as well as recent attack vectors. In order to mitigate these findings, this work proposed a new dataset concentrating on realistic attack scenarios containing a broad arsenal of attacks based on penetration testing tools, recent malware, and exploits. A comparative study revealed, it can compete with related work in terms of rich captures and ground truth, but it is superior considering the quantity of attack variants. Furthermore, the evaluation against the latest Snort version demonstrated several attacks remained undetected or were hit only partially, which manifests the merit of our effort in addition. In this respect, the captures can be deemed complementary to existing traces. Based on these results, future work attends to assess further detectors and to evaluate potential limitations using the dataset. Besides these activities, we plan to expand the traces with more sophisticated attacks focusing on current and emerging threats.

References

[AB14] Abt, S.; Baier, H.: Are We Missing Labels? A Study of the Availability of Ground-Truth in Network Security Research. In: 3rd International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security. pp. 40–55, 2014.

[AH11] Aviv, A.J.; Haeberlen, A.: Challenges in Experimenting with Botnet Detection Systems. In: 4th Workshop on Cyber Security Experimentation and Test. 2011.

[BB17] Beer, F.; Bühler, U.: Feature Selection for Flow-based Intrusion Detection Using Rough Set Theory. In: 14th IEEE International Conference on Networking, Sensing and Control. 2017.

[Be14] Beigi, E.B.; Jazi, H.H.; Stakhanova, N.; Ghorbani, A.A.: Towards Effective Feature Selection in Machine Learning-based Botnet Detection Approaches. In: IEEE Conference on Communications and Network Security. pp. 247–255, 2014.

[BG15] Buczak, A.L.; Guven, E.: A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. IEEE Communications Surveys & Tutorials, 18(2):1153–1176, 2015.

[BWM08] Brauckhoff, D.; Wagner, A.; May, M.: FLAME: A Flow-Level Anomaly Modeling Engine. In: 2nd Workshop on Cyber Security Experimentation and Test. 2008.

[Ce11] Celik, Z.B.; Raghuram, J.; Kesidis, G.; Miller, D.J.: Salting Public Traces with Attack Traffic to Test Flow Classifiers. In: 4th Workshop on Cyber Security Experimentation and Test. 2011.

[Ga14] García, S.; Grill, M.; Stiborek, H.; Zunino, A.: An Empirical Comparison of Botnet Detection Methods. Computers and Security, 45:100–123, 2014.

[Ha01] Haines, J.; Lippmann, R.; Fried, D.; Zissman, M.; Tran, E.; Boswell, S.: 1999 DARPA Intrusion Detection Evaluation: Design and Procedures. Technical report, MIT Lincoln Laboratory, 2001.

[Ho14] Hofstede, R.; Hendriks, L.; Sperotto, A.; Pras, A.: SSH Compromise Detection Using NetFlow/IPFIX. ACM SIGCOMM Computer Communication Review, 44(5):20–26, 2014.

[Li00a] Lippmann, R.; Fried, D. J.; Graf, I.; Haines, J. W.; Kendall, K. R.; McClung, D.; Weber, D.; Webster, S. E.; Wyschogrod, D.; Cunningham, R. K.; Zissman, M. A.: Evaluating Intrusion Detection Systems: The 1998 DARPA Off-line Intrusion Detection Evaluation. In: DARPA Information Survivability Conference and Exposition. volume 2, pp. 12–26, 2000.

[Li00b] Lippmann, R.; Haines, J.; Fried, D.; Korba, J.; Das, K.: The 1999 DARPA Off-line Intrusion Detection Evaluation. Computer Networks, 34(4):579–595, 2000.

[MBM15] Małowidzki, M.; Berezinski, P.; Mazur, M.: Network Intrusion Detection: Half a Kingdom for a Good Dataset. In: NATO STO SAS-139 Workshop. 2015.

[Mc00] McHugh, J.: Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations As Performed by Lincoln Laboratory. ACM Transactions on Information and System Security, 3(4):262–294, 2000.

[MC03] Mahoney, M. V.; Chan, P. K.: An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection. In: 6th Symposium on Recent Advances in Intrusion Detection. pp. 220–237, 2003.

[Pe08] Perona, I.; Gurrutxaga, I.; Arbelaitz, O.; Martin, J. I.; Muguerza, J.; Pérez, J. M.: Service-independent payload analysis to improve intrusion detection in network traffic. In: 7th Australasian Data Mining Conference. pp. 171–178, 2008.

[Sa15] Santanna, J.J.; van Rijswijk-Deij, R.; Sperotto, A.; Hofstede, R.; Wierbosch, M.; Granville, L. Zambenedetti; Pras, A.: Booters - An analysis of DDoS-as-a-Service Attacks. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management. pp. 243–251, 2015.

[Sh11] Sherif, S.; Issa, T.; Ali, G.; Bassam, S.; David, Z.; Wei, L.; John, F.; Payman, H.: Detecting P2P Botnets through Network Behavior Analysis and Machine Learning. In: Ninth Annual International Conference on Privacy, Security and Trust. pp. 174–180, 2011.

[Sh12] Shiravi, A.; Shiravi, H.; Tavallaee, M.; Ghorbani, A. A.: Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Computers and Security, 31(3):357–374, 2012.

[Sp09] Sperotto, A.; Sadre, R.; van Vliet, F.; Pras, A.: A Labeled Data Set For Flow-based Intrusion Detection. In: 9th IEEE International Workshop on IP Operations and Management. pp. 39–50, 2009.

[Sp10] Sperotto, A.; Schaffrath, G.; Sadre, R.; Morariu, C.; Pras, A.; Stiller, B.: An overview of IP flow-based intrusion detection. IEEE Communications Surveys & Tutorials, 12(3):343–356, 2010.

[Sz08] Szabó, G.; Orincsay, D.; Malomsoky, S.; Szabó, I.: On the Validation of Traffic Classification Algorithms. In: International Conference on Passive and Active Network Measurement. pp. 72–81, 2008.

[Ta09] Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A. A.: A Detailed Analysis of the KDD CUP 99 Data Set. In: IEEE Symposium on Computational Intelligence in Security and Defense Applications. pp. 53–58, 2009.

On the Perception of Risk Assessment in Intrusion Detection Systems

Mario Golling[1], Robert Koch[1] and Gabi Dreo Rodosek[1]

Abstract: Especially in the area of Intrusion Detection, the concept as well as the understanding of the term "risk" is of fundamental importance. Generally, risk assessment represents an important means of evaluating certain situations, plans, events or systems in a systematic and comprehensive procedure. As in other areas, within the field of IT security, the systematic assessment process (risk analysis) also aims at recommending how to allocate available resources. Referring to this, both, the categorization of traffic (whether traffic has to be classified as an attack or not - “benign vs. malicious”) as well as a corresponding estimation of the expected damage (severity) are of central importance. Therefore, within this publication, the authors address the following questions in detail: (1) To what extent are the detection results of different IDSs comparable - with regard to the assessment of the risk / extent of damage - or are there strong deviations? (2) How do both vendor-dependent and vendor-independent alerts address the topic of risk assessment and enable the implementation of a comprehensive risk concept? To this end, at the heart of this paper, an overview as well as an evaluation of important representatives of open source IDSs is presented, focusing on methods for risk assessment resp. risk rating including cross-vendor risk rating and the Common Vulnerability Scoring System (CVSS). Furthermore, the paper also contains a brief demise of the most important representatives of commercial IDSs.

Keywords: Network Security, Intrusion Detection, Risk Rating, Risk Assessment, Risk Severity

1 Introduction

Generally, risk assessment represents an important means of evaluating certain situations, plans, events or systems in a systematic and comprehensive procedure. As in other areas, within the field of IT security, the systematic assessment process of risk analysis also aims at recommending how to allocate available resources to perform in-depth analyzes or to develop appropriate counter-measures in order to minimize total exposure. With regard to Intrusion Detection Systems (IDSs)/Intrusion Prevention Systems (IPSs) risk assessment is particularly important as it represents an important means of evaluating the current situation wrt. IT security and to focus on important alerts rather than to treat all alerts in the same way (especially if the resources are insufficient to investigate all alerts in depth). Following these considerations, the various methods of important representatives of IDSs are presented and differentiated from each other within this paper. To this end, this paper is structured as follows: Section 2 contains an overview of important representatives of open source IDSs as well as vendor-independent approaches. Section 3 then briefly deals with commercial IDSs in the same way, before the paper is concluded in Section 4.

[1] Munich Network Management Team (MNM-Team), Universität der Bundeswehr München, Werner-Heisenberg-Weg 39, D-85577 Neubiberg, {mario.golling, robert.koch, gabi.dreo}@unibw.de

2 Risk Rating in Popular Open Source IDSs

Selection Criteria

In the field of open source IDSs, a wide variety of systems and procedures are present. Unlike commercial systems, that are subject to various market analysis, an overview of important representatives of open source IDSs is, for the lack of generally accepted data, far more difficult. Nevertheless, due to the page limitations of this paper, we are obliged to reduce the range of systems, especially in terms of their number. Table 1 briefly illustrates the reasons/motivations for the systems selection.

Tab. 1: Popularity of Open Source IDSs

SOURCE	SNORT	SURICATA	BRO	PRELUDE
Debian Popularity Contest (higher numbers represent greater popularity)	914	83	not included	25
Alienvault Blog: Open Source Intrusion Detection Tools: A Quick Overview *This investigation only covers Snort, Suricata and Bro*	*"The de-facto standard for IDS"*	*"What's the only reason for not running Snort? If you're using Suricata instead"*	*"Starting to gain a larger community following"*	not mentioned
Pathan: The State of the Art in Intrusion Prevention and Detection *Only those 4 open source NIDS are listed*	*"Snort is considered as the de facto standard of the IDS/IPS with millions of downloads and is the most extensively deployed IDS worldwide"*	*"The high-performance Suricata IDS [. . .] has been advanced as an open-source improvement for the popular Snort"*	*"widely used as an intrusion detection system" "distinguishes itself from Snort by offering high-speed network capability"*	*"Prelude is a security information management system [... and] can collect alert data from other security applications or generate its own alert data"*
Intrusion Detection Case Study *Only those 4 open source NIDS are listed*	included	included	included	included
Comparison of Open Source Network Intrusion Detection Systems *Investigation was limited to Snort, Bro, Suricata*	included	included	included	not included
Evaluation studies of three intrusion detection systems under various attacks and rule sets	*"The open source IDS commonly used are Snort, Suricata, and Bro"*			not included

For the sake of clarity, within this Section, we divided the systems into single-IDSs (Snort, Suricata and Bro) and cross-manufacturer approaches (Prelude and Common Vulnerability Scoring System (CVSS). As it will be shown later, CVSS is not a system per se. Instead, it is a cross-systems/cross-vendor approach, which, due to its importance, has to be covered in this paper, too.

Individual Open Source IDSs

Snort

Snort is an open source NIDS originally written by Martin Roesch. Although the first version was focussing on signature-based detection only, the current version of Snort also includes other detection techniques (protocol-based as well as anomaly-based Intrusion Detection).

Snort is generally considered as the de facto standard for Intrusion Detection [Pa14]. Snort is a packet sniffer and logger that features rule-based logging to perform content pattern matching and is capable of detecting a variety of attacks and probes, such as buffer overflows, stealth port scans, CGI attacks, SMB probes, OS fingerprinting attempts etc. Snort has real-time alerting capabilities and a detection engine, which is programmed using a simple language that describes per packet tests and actions. See Figure 1 for an impression of a Snort rule. By default, alert messages of Snort typically include a priority level with a

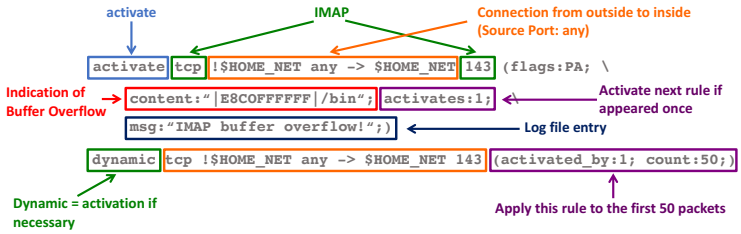


Fig. 1: Example of a Snort Rule (with minimum explanation)

gradation in four categories (high, medium, low, very low). For this, an integer value called *Priority* is used. A value of 1 (high) is the most severe and 4 (very low) is the least severe. Listing 1 shows an example of a Snort alert (with a priority of 2).

List. 1: Example of a Snort alert

```
[**] [1:497:11] ATTACK-RESPONSES file copied ok [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
03/09-20:34:00.379435 205.206.231.13:80 -> xxx:61607
TCP TTL:43 TOS:0x0 ID:54613 IpLen:20 DgmLen:1492 DF
***A**** Seq: 0xD8357B2D Ack: 0x49F73C5E Win: 0x2220 TcpLen: 20
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=2000-0884][Xref => http://www.
    securityfocus.com/bid/1806]
```

In principle, the priority of Snort allows to use numbers greater than 4 as well. Snort itself describes the priority by means of an integer value, hence also allowing values greater than 4. For this, the default priorities assigned by Snort can be overwritten locally by the user. This can, for example, be useful to adapt Snort to the local environment. As Snort, for instance, by default assigns a priority of 3 to Telnet activities, using a higher priority level may be wise, if telnet connections are not a rarity in the investigated network [OBB05]. However, it can be assumed that only a few users are making use of this. With the help of attack taxonomies such as Common Vulnerabilities and Exposures (CVE), an even more detailed prioritization can be made, e.g., by using the CVSS. This is described in more detail at the end of this Section.

Suricata

In contrast to Snort, which is per default limited to use rule sets written in a specific format, Suricata is capable of using additional formats, too. Suricata uses multithreading, which makes it faster than other IDSs. With regard to risk rating, Suricata provides a range of values from 1-255, which is almost exclusively reduced to the values 1-4 in practice. This

is mainly due to the fact, that - while Suricata is able to use rules from different sources (to provide the best rule set possible) - the Suricata developers intended to support the same rule language used in the Snort rules [Al11], which, in practice, is often the case. Listing 2 illustrates some examples of Suricata alerts.

List. 2: Examples of Suricata alerts [Se13]

```
03/10/2011-13:58:00.924783 [**] [1:2009702:4] ET POLICY DNS Update From External net
    [**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {UDP}
    192.168.100.37:53 -> 192.168.100.35:53
03/10/2011-13:58:30.921484 [**] [1:410:5] ICMP Fragment Reassembly Time Exceeded [**]
    [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.100.35:11 ->
    192.168.100.37:1
03/10/2011-13:58:47.715668 [**] [1:2009702:4] ET POLICY DNS Update From External net
    [**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {UDP}
    192.168.100.37:53 -> 192.168.100.35:53
```

Bro

From its origin, Bro is a traffic analyzer, which can also be used as an IDS. Bro has gained its reputation mainly due to its stateful protocol capabilities [Pa14]. Bro detects intrusions by first parsing network traffic to extract its application-level semantics and then executing event-oriented analyzers that compare the activity with patterns deemed troublesome. The detection includes specific attacks: Those defined by signatures (*knowledge-based detection*), but also those defined in terms of events and unusual activities (*behavior-based*); e.g., certain hosts connecting to certain services, or patterns of failed connection attempts. The important feature of Bro that differentiates it from other IDSs, such as Snort (at least from the initial version of Snort), is that Bro scripts can be written to understand application semantics and could be trained to look for anomalies (*behavior*) [Ba06]. As actions to alerts, Bro supports a whole series of options, such as logging, email notification, dropping the traffic or adding geo data to the message. For the actual risk assessment, however, there are no mechanisms (neither a proprietary model nor a manufacturer-independent method such as CVE/CVSS).

Cross-Vendor Approaches

Prelude/Intrusion Detection Message Exchange Format (IDMEF)

The Prelude Intrusion Detection System differs from the approaches described so far as it is a Security Information Event Management (SIEM) system. Prelude collects, normalizes, sorts, aggregates, correlates, and reports all security-related events independently of the product brand or license [CS14]. Prelude can either make use of different types of sensors (other security applications, such as Snort) or use own components for evaluation. Similar to Snort and Bro, an open source version with limited performance (called Prelude OSS) as well as a commercial version (Prelude Pro) is available. Sensors feed their data to the Prelude Manager with the use of the Intrusion Detection Message Exchange Format (IDMEF). As the “Lingua Franca” for Security Incident Management [Co03], IDMEF and its associated protocols enable a common language used to discuss Intrusion Detection events as a basis for cross-product event correlation. For that purpose, the manager collects and normalizes IDMEF data and makes it available to output plugins. Essentially, normalization allows all collected events to be stored in the same database in the same format [Ya09]. With regard

to risk rating, IDMEF in turn uses 4 Severity Levels (info, low, medium, high). However, IDMEF may also include a reference to a vulnerability database (such as the Open Source Vulnerability Database), and thus, ultimately, to the Common Vulnerability Scoring System (CVSS). For a summary and an overview of the mutual interdependencies, see also Figure 2.

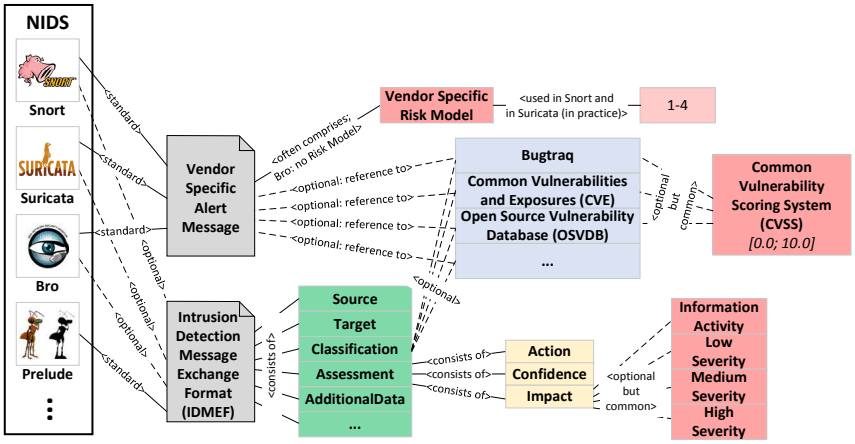


Fig. 2: Relationship between Bro, IDMEF and CVSS, in particular with regard to Risk Assessment

Common Vulnerability Scoring System (CVSS)

CVSS was commissioned by the National Infrastructure Advisory Council (NIAC), a working group of the U.S. Department of Homeland Security, in 2005 and is currently managed by the Forum of Incident Response and Security Teams. In the development of CVSS among others, the following entities have been involved: CERT, Cisco, DHS / MITRE, eBay, IBM, Microsoft, Qualys, or Symantec. As depicted in Figure 3, CVSS consists of different groups: Base, Temporal and Environmental, each comprising a numeric score ranging from 0 to 10, with 10 being the most severe.

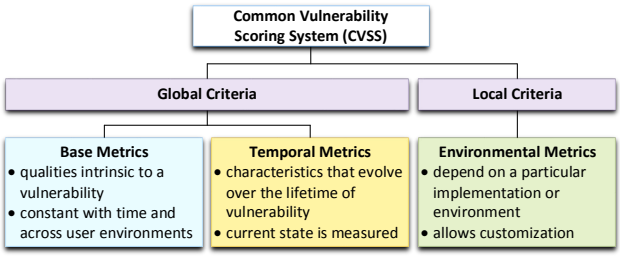


Fig. 3: Main areas of CVSS

The Base group represents the intrinsic qualities of a vulnerability [MSR17]. The Temporal group reflects the characteristics of a vulnerability and may change over time. Generally, base as well as temporal metrics are specified by vulnerability bulletin analysts, security product vendors, or application vendors [MSR17] and therefore can be considered as *global criteria*. In contrast to this, the Environmental group represents the characteristics of a vulnerability with regard to the user's environment (*local criteria*). Each group consists of

multiple separate categories. Base Metrics consists of: Attack Vector, Attack Complexity, Privileges Required, User Interaction, Scope, Confidentiality Impact, Integrity Impact and Availability Impact. Temporal Metrics consists of Exploitability, Remediation Level and Report Confidence. Environmental Metrics consists of Security Requirements (to customize depending on the importance of the affected IT asset, measured in terms of confidentiality, integrity, and availability) and Modified Base Metrics (enable to adjust the Base metrics).

Overview of Risk Rating Models of Open Source IDSs and Cross-Vendor Approaches
For the sake of clarity, the four considered IDSs are listed and distinguished within Table 2.

Tab. 2: Overview of Open Source Risk Rating Models

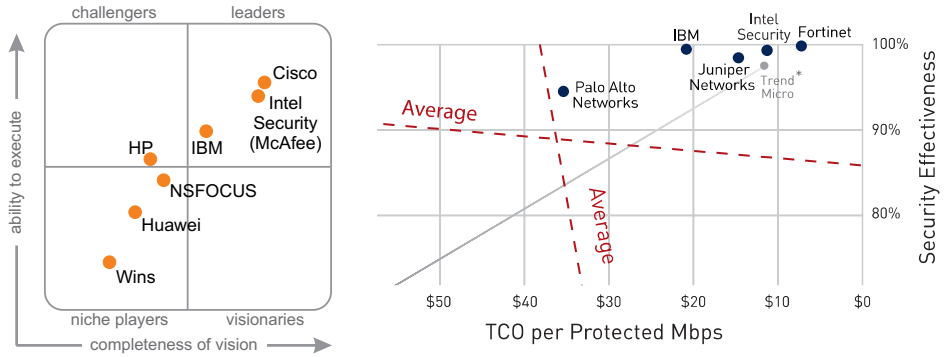
SNORT	SURICATA	BRO	PRELUDE/ IDMEF	CVSS
by default, simple model with 4 classifications: • high (1), • medium (2), • low (3), • very low (4)	simple model: • theoretical classification area: 1-255 (1 = high, 255 = low) • in practice: 1-4 (see Snort)	Bro itself offers no mechanisms for risk rating (neither a proprietary model nor a manufacturer-independent method such as CVSS)	simple model with 4 classifications: • info (0), • low (1), • medium (2), • high (3)	complex model consisting of 3 main classes of severity: • Base Metrics • Temporal Metrics • Environmental Metrics Overall scoring has an interval from 0 to 10.

3 Risk Rating in Commercial IDSs

Similar to Section 2, this Section presents commercial IDSs, whereby often much less information is available on how these systems evaluate alerts (in terms of motivation etc.). There are also no generally accepted cross-manufacturer approaches.

Selection Criteria
With regard to the risk rating of commercial IDSs, a restriction has been made to those systems that have been evaluated in the current study of Gartner called "Magic Quadrant for Intrusion Prevention Systems" as particularly outstanding (leaders and challengers; see [HYD13]). Figure 4a depicts the latest version. Other studies such as the well-known NSS-study (see [NS16] and Figure 4b) have not been taken into account. For the latter, “irregularities”, such as the very high detection rates (> 99%) were the reasons. In general, the fact that (i) the Gartner study has also acquired considerable importance outside the “pure scientific area” and (ii) the Gartner study is also free to obtain (in some parts) - which is often not the case with many other analyses (these costs quickly 5,000 USD or more) - were the driving factor.

Individual Commercial IDSs
Cisco
In contrast to simple models (like the one of Suricata/Snort, in which, by default, only



(a) Magic Quadrant for IDSs

(b) NSS Labs' 2016 Security Value Map for Data Center Intrusion Prevention System

Fig. 4: Overview of important Commercial IDSs

four values are used), Cisco uses a slightly more complex risk rating model, realized as an integer value in the range from 1 to 100 [Ci09]. Here, the greater the security risk, the higher the value [Ci09]. Three subcomponents are used within the overall risk rating formula:
Signature Fidelity Rating is a variable, which in turn can contain values from 1 to 100, measuring the accuracy of the signature. The Signature Fidelity Rating is assigned by Cisco, but can be modified by the user depending on the environment, such as the OS, service, application, or patch level. With regard to applications, it may also occur that a legitimate application produces traffic that mimics the behavior of an exploitation of a network vulnerability.
Alert Severity Rating describes the damage if the attack succeeds. Its value is again predefined with four degrees: *Information* (25, generally poses no immediate threat), *Low (50, somewhat unusual on most networks)*, *Medium (75, should generally not be seen on the network)* and *High (100, indicative of an active attack or an obvious precursor to an attack)*.
Target Value Rating is used to modify the risk rating based on the target of the attack and is therefore user-defined. This allows the user to increase the risk of an event associated with a critical system and to deemphasize the risk of an event on a low-value target [Ci09]. A *Low Asset Value* corresponds to a score of 75, a *Medium Asset Value* to 100, a *High Asset Value* to 150 and a *Mission Critical Asset Value* is assigned a scoring of 200.
The corresponding overall risk rating formula is as follows:

$$\textit{Risk Rating} = \frac{\text{Signature Fidelity Rating} * \text{Alert Severity Rating} * \text{Target Value Rating}}{100 * 100 * 100} \tag{1}$$

with the final results that exceed 100 being rounded down to 100 [Ba11].

Intel Security/McAfee
The Network Security Platform of McAfee assigns a default severity to every attack using a numbering score from 0 to 9, based on the immediate effect, or impact, on the target

system [Mc15]. The guidelines in assigning severity levels are very similar to those used in many open security forums [Mc15]. Table 3a describes the numbering scheme as well as the mapping to indicate Informational, Low, Medium, and High. Table 3b illustrates McAfee's attack categories and corresponding severity ranges.

Tab. 3: McAfee's risk assessment perception

(a) Alert numbering scheme of McAfee's Network Security Platform [Mc15]

INFORMATIONAL	LOW	MEDIUM	HIGH
0	1-3	4-6	7-9

(b) Excerpt from McAfee's attack categories and corresponding severity ranges [Mc15]

CATEGORY	THREAT TYPE	RANGE USED IN NETWORK SECURITY PLATFORM
Reconnaissance	Host sweep	4-4
	Port scan	4-4
	OS Fingerprinting	6-6
Exploits	Buffer Overflow	7-9
	Bot	7-9
	DoS	3-5
	DDoS Agent Activity	7-9
	Worm	6-9
Policy Violation	Unauthorized IP	5-5
	Covert Channel	5-5
	Command Shell	4-4

IBM
At IBM, the individual security events also receive a severity level; here, in the categories of High, Medium and Low [IB17]. Integrated into the IBM Network IPS is also a Snort system. The severity of these rules is also specified in terms of the categories High, Medium and Low. The corresponding Network IPS appliance provides alerts with the use of the categories Low, Medium and High as well.

Trend Micro
Trend Micro's TippingPoint has a classification mechanism, which for instance serves as a basis for the ability to color code the security reports. Table 4 illustrates the different severity levels.

Tab. 4: Severity Levels of TippingPoint

SEVERITY LEVEL	DESCRIPTION	COLOR USED FOR THE REPORTS
Critical	attacks that must be looked at immediately	Red
Major	attacks that must be looked as soon as possible	Yellow
Minor	attacks that should be looked at as time permits	Cyan
Low	traffic that is probably normal, but may have security implications	Gray

Overview of Risk Rating Models of Commercial IDSs
Table 5 summarizes the various commercial risk rating models.

4 Conclusions and Outlook

Today, numerous IDSs and IPSs are available, sometimes tailored to special scopes of application. Although risk rating approaches are integrated (partially) in some of the major

Tab. 5: Overview of Commercial Risk Rating Models

CISCO	JUNIPER	TIPPINGPOINT	MCAFEE	IBM
complex model consisting of: • accuracy of the signature (Signature Fidelity Rating) • damage if the attack succeeds (Alert Severity Rating) • target impact (Target Value Rating) aggregated risk rating model, realized as an integer value in the ranging from 0 to 100 • 1 = very low • 100 = very high	simple model represented by five groups: • Critical (Severity 1) • Major (Severity 2) • Minor (Severity 3) • Warning (Severity 4). • Informational (Severity 5)	simple model consisting of: • Critical (Number 4) • Major (Number 3) • Minor (Number 2) • Low (Number 1)	simple model consisting of a rating from 0-9 and 4 superclasses: • High (Number 7-9) • Medium (Number 4-6) • Low (Number 1-3) • Informational (Number 0)	simple model consisting of 3 severity levels: • High • Medium • Low

IDSs, their current functionality and exploitation is not sufficient. In the paper, we presented and compared different well-known open source as well as commercial IDSs including cross-vendor approaches and their respective risk rating capabilities. In this respect, there is a notable deviation, in particular with respect to the graduation (scale) as well as the underlying motivation. Besides those syntax-related differences, there are also differences in the area of semantics. It thus may happen that manufacturer A classifies an incident as serious and manufacturer B as medium. CVE/CVSS only partly changes this, since an CVE/CVSS-evaluation must first be made by a person and thus is only available with a certain time delay after the first occurrence.

Currently, we are working on the design of a generally applicable risk rating component which is able to process alert information of all major open source as well as proprietary IDSs presented based on IDXP, the Intrusion Detection Exchange Protocol (IDXP), the recommended transport protocol for IDMEF. In a further step, we also aim to assess their criticality wrt. the monitored network and pre-defined risk patterns. By that, intrusion alarms can be prioritized and evaluated based on, e.g., pre-defined SLAs or risk values. This enables multi-layered and more sophisticated Intrusion Detection reactions: e.g., an alarm which may be a false alarm can be dropped, if an incorrect decision based on it would generate a higher financial damage than missing a true alert would do (e.g., because the penalty that has to be paid when a service is deactivated (as part of a counter-measure) - by far - exceeds the expected damage).

References

[Al11] Albin, Eugene: A comparative analysis of the snort and suricata intrusion-detection systems. Master's thesis, Monterey, California. Naval Postgraduate School, 2011.

[Ba06] Babbin, Jacob: Security log management: identifying patterns in the chaos. Syngress, 2006.

[Ba11] Barker, Keith: , Protecting Critical Resources with Target Value Ratings (TVRs), August 2011. http://www.pearsonitcertification.com/articles/article.aspx?p=1739167, last seen on 11.01.2017.

[Ci09] Cisco Systems: , Cisco IPS Risk Rating Explained, 2009. http://www.cisco.com/en/US/prod/collateral/vpndevc/ps5729/ps5713/ps4077/prod_white_paper0900aecd80191021.pdf, last seen on 07.04.2017.

[Co03] Corner, DS: IDMEF-" Lingua Franca" for Security Incident Management Tutorial and Review of Standards Development. SANS Institute, 2003.

[CS14] CS Group: , Prelude: Security Information and Event Management, 2014. http://www.prelude-siem.com/en/, last seen on 11.01.2017.

[HYD13] Hils, Adam; Young, Greg; D'Hoinne, Jeremy: , Magic Quadrant for Intrusion Prevention Systems, December 2013. http://www.gartner.com/technology/reprints.do?id=1-1OAVJS3&ct=131217&st=sb, last seen on 11.01.2017.

[IB17] IBM: , IBM Security Network Intrusion Prevention System (IPS): Configuring general settings for security events, 2017. https://www.ibm.com/support/knowledgecenter/SSB2MG_4.6.1/com.ibm.ips.doc/tasks/configuring_general_settings_for_security_events.htm, last seen on 11.01.2017.

[Mc15] McAfee: , McAfee Network Security Platform 8.2, 2015. https://kc.mcafee.com/resources/sites/MCAFEE/content/live/PRODUCT_DOCUMENTATION/25000/PD25605/en_US/NSP-8275-8275-Virtual-IPS_Release-Notes_revB_en-us.pdf, last seen on 11.01.2017.

[MSR17] Mell, Peter; Scarfone, Karen; Romanosky, Sasha: , A Complete Guide to the Common Vulnerability Scoring System Version 2.0, 2017. http://www.first.org/cvss/cvss-guide.pdf, last seen on 11.01.2017.

[NS16] NSS LAbs: , Data Center Intrusion Prevention System Test, 2016. https://www.nsslabs.com/research-advisory/security-value-maps/2016/data-center-ips-svm-graphic/, last seen on 11.01.2017.

[OBB05] Orebaugh, Angela; Biles, Simon; Babbin, Jacob: Snort cookbook. O'Reilly Media, Inc., 2005.

[Pa14] Pathan, Al-Sakib Khan: The State of the Art in Intrusion Prevention and Detection. CRC Press, 2014. http://books.google.de/books?hl=de&lr=&id=o39cAgAAQBAJ&oi=fnd&pg=PA115, last seen on 11.01.2017.

[Se13] Sebastien Damaye: , Suricata-vs-snort/Test-cases/Fragmented-packets, November 2013. http://www.aldeid.com/wiki/Suricata-vs-snort/Test-cases/Fragmented-packets, last seen 11.01.2017.

[Ya09] Yasm, Curt: , Prelude as a Hybrid IDS Framework, 2009. http://www.sans.org/reading-room/whitepapers/awareness/prelude-hybrid-ids-framework-33048, last seen on 11.01.2017.

CAKE: Hybrides Gruppen-Schlüssel-Management Verfahren

Peter Hillmann, Marcus Knüpfer und Gabi Dreo Rodosek[1]

Abstract: Mit zunehmender Vernetzung von Systemen gibt es immer mehr Teilnehmer, die innerhalb einer Gruppe über ungesicherte Kommunikationskanäle vertrauliche Informationen austauschen. Um den Zugriff auf diese Daten durch Dritte zu verhindern, ist das Verschlüsseln dieser unumgänglich. Dazu nutzen die Gruppenteilnehmer einen gemeinsamen Schlüssel, der gesichert zu verteilen ist. Gerade im Bereich von MANETs bilden die begrenzten Ressourcen hinsichtlich der Rechen- und Übertragungskapazität sowie die dynamisch ändernde Gruppenzusammensetzung besondere Herausforderungen. Zur Koordinierung und Verteilung wird in der vorliegenden Arbeit ein neuartiges, hybrides Gruppen-Schlüssel-Management Verfahren vorgestellt, welches zentral organisiert ist und strikten Sicherheitsanforderungen genügt. Durch einen hybriden Ansatz werden die Vorteile der existierenden Protokolle kombiniert, mit dem Ziel den Berechnungs- und Kommunikationsaufwand zu verringern. Es wird gezeigt, dass sich das Verfahren für ändernde MANET-Gruppen besser eignet als die bestehenden. Darüber hinaus lässt sich das Verfahren auch in weiteren Anwendungsgebieten, wie kabelgebundenen Weitverkehrsnetzen einsetzen. Der Gruppenschlüssel ist dabei für beliebige Dienste anwendbar.

Keywords: Sichere Kommunikation, Ad hoc Netz, MANET, GKMP, Schlüsselmanagement

1 Einleitung

Für den Austausch von Daten innerhalb einer Gruppe existieren zur Verwaltung der Gruppenschlüssel sogenannte Gruppen-Schlüssel-Management (GKM) Verfahren. Diese übernehmen den sicheren und effizienten Austausch der Schlüssel, welche zur Kommunikation innerhalb einer Gruppe genutzt werden. Alle Teilnehmer einer Gruppe besitzen dazu den gleichen symmetrischen Schlüssel, wodurch die Informationen nur ein einziges Mal für die Teilnehmer der Gruppe zu verschlüsseln sind. Für welchen Dienst der gemeinsame Gruppenschlüssel eingesetzt wird, ist schlussendlich jedoch freigestellt.

Verschiedene Anwendungsbereiche haben spezifische Anforderungen an die Eigenschaften eines GKM Verfahrens, wodurch sich kein Verfahren als allgemeingültiger Lösungsansatz anwenden lässt. In der vorliegenden Arbeit wird ein GKM Verfahren vorgestellt, welches an eine Umgebung mit begrenzter Kommunikationsbandbreite und geringer Rechenleistung der Gruppenteilnehmer angepasst ist. Die vorrangige Motivation liegt in der Verbesserung der Effizienz gegenüber derzeitigen Verfahren, sodass im Netz mehr Bandbreite für die Nutzdaten zur Verfügung steht. Auf Basis des klassischen *Group Key Management Protocol* (GKMP), des *Secure Lock* (SL) und der *Local Key Hierarchy* (LKH) wird ein hybrides Verfahren zum GKM entwickelt.

[1] Universität der Bundeswehr München, Lehrstuhl für Kommunikationssysteme und Netzsicherheit, Werner-Heisenberg-Weg 39, D-85577 Neubiberg, {peter.hillmann, marcus.knuepfer. gabi.dreo}@unibw.de

Abschnitt 2 erläutert zunächst ein Szenario, um daraus Anforderungen abzuleiten. Anschließend wird in Abschnitt 3 auf den Stand der Technik eingegangen und in Abschnitt 4 das neue Verfahren vorgestellt. Eine Bewertung der Sicherheitsanforderungen befindet sich in Abschnitt 5. Abschließend werden die Ergebnisse der Arbeit zusammengefasst.

2 Szenario und Anforderungen

Als Anwendungsbeispiel dient ein militärisches Szenario im Bereich der sogenannten mobilen Ad-hoc-Netze (MANET). Hierbei bewegen sich mehrere Teilnehmer in einem Gelände und tauschen währenddessen mittels Funkkommunikation Nachrichten aus. In Abb. 1 sind verschiedene Zustände dargestellt, aus denen sich die typischen Gruppenoperationen ableiten lassen. Die jeweils verschiedenfarbigen Kreise entsprechen dem Kommunikationsradius bzw. dem Schlüsselbereich der jeweiligen Gruppe. Die 8-12 Teilnehmer werden durch die taktischen Zeichen dargestellt. Im Zustand 1 befindet sich eine Gruppe A auf dem Weg eine andere Gruppe B zu unterstützen. Zur gesicherten Kommunikation beider Gruppen müssen diese in einen gemeinsamen Schlüsselbereich eintreten (Gruppenverschmelzung). Im Zustand 2 ist dargestellt, wie ein einzelner Späher die vereinigte Gruppe erreicht und in den gemeinsamen Schlüsselbereich eintritt (Eintritt). Dieser berichtet von nicht identifizierten Personen, welche in der Nähe gesichtet wurden. Infolgedessen trennt sich Gruppe A aus dem Verbund (Gruppenteilung) und bewegt sich in Richtung der aufgeklärten Personen, dargestellt durch Zustand 3. Im letzten Zustand 4 wird aus Gruppe A ein Melder zur Materialübergabe in den rückwärtigen Raum geschickt, welcher somit die Gruppe verlässt (Austritt).
Weitere zivile Anwendungsbeispiele sind Multicast-Datenübertragungen im Bereich vom Video on Demand oder dynamische Projektteams in der Forschung.

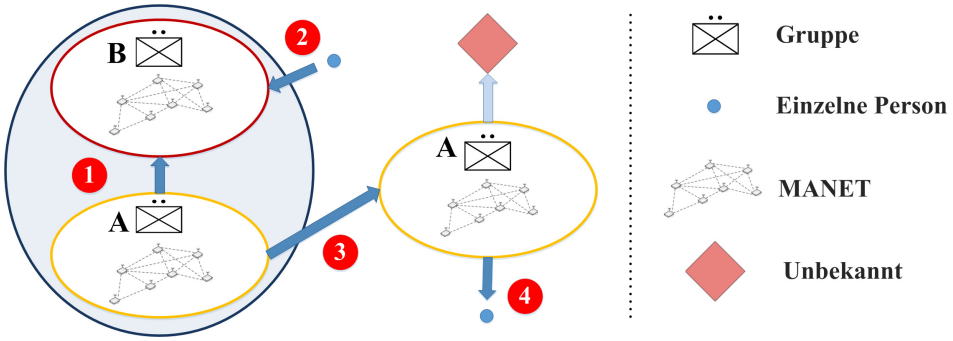


Abb. 1: Verschiedene Operationen bei der verschlüsselten Gruppenkommunikation

Für die sichere Übermittlung von Nachrichten ist ein Kryptosystem notwendig, welches sich in die Rahmen des MANET einfügt. Hierbei werden folgende Anforderungen an die Sicherheit gestellt [Bu13]:

- *Forward Secrecy:* Austretenden Teilnehmern soll es nicht mehr möglich sein, weiterhin empfangene Nachrichten entschlüsseln zu können.
- *Backward Secrecy:* Eintretenden Teilnehmern soll es nicht möglich sein, im Vorfeld empfangene Nachrichten im Nachhinein entschlüsseln zu können.

- *Kollisionsfreiheit:* Eine unerlaubte Vorgehensweise einer Teilmenge von Teilnehmern führt nicht zum Schaden eines einzelnen Gruppenteilnehmers.
- *Schlüsselunabhängigkeit / Folgenlosigkeit:* Die Kenntnis eines Schlüssels ermöglicht keine Schlussfolgerung auf weitere Schlüssel.

Im Zusammenhang mit dem Szenario ergeben sich daraus die folgenden Anforderungen an die Gruppenoperationen, welche durch ein modernes GKM Verfahren abzudecken sind, ohne dabei die Sicherheitsanforderungen zu missachten:

- *Einzel- und Mehrfach-Eintritt:* Ein oder mehrere Teilnehmer treten in eine bestehende Gruppe ein. (Beachtung der Backward Secrecy)
- *Einzel- und Mehrfach-Austritt:* Ein oder mehrere Gruppenmitglieder treten aus der Gruppe aus. (Beachtung der Forward Secrecy)
- *Re-Keying:* Die Aktualisierung des Gruppenschlüssels muss über eine effiziente Vorgehensweise möglich sein.
- *Gruppenverschmelzung:* Mehreren Gruppen ist durch Re-Keying ein gemeinsamer Schlüssel effizient bereitzustellen. (Beachtung der Backward Secrecy)
- *Gruppenteilung:* Eine Gruppe teilt sich in mehrere Teilgruppen auf. (Beachtung der Forward Secrecy)

Zur Entlastung der Teilnehmer werden die Schlüssel zentral von einer Basis-Station mit mehr Leistung generiert und über eine hierarchische Struktur verteilt. Dadurch verlagert sich der Rechenaufwand für die Schlüsselgenerierung zu einer Instanz. Jeder Teilnehmer besitzt über das MANET bzw. über einen Satelliten-Link eine Kommunikationsverbindung zu der zentralen und vertrauenswürdigen Basis-Station. Hierbei werden folgende Anforderungen gestellt:

- Verschlüsselte und gesicherte Nachrichtenübermittlung, da diese insbesondere bei Verwendung von Funkkommunikation abhörbar ist.
- Eine geringe verfügbare Bandbreite bedingt kleine Datenpakete und eine geringe Anzahl an Nachrichten.
- Die geringe Rechenleistung der mobilen Funkgeräte ist zu berücksichtigen.
- Der Overhead durch das Management der Gruppenschlüssel ist gering zu halten.

3 Stand der Technik

Die verschiedenen GKM Verfahren lassen sich in drei Hauptkategorien einteilen, welche wiederum aus verschiedenen Unterkategorien bestehen [CS08, Ra00]. Bei zentralisierten Verfahren erfolgt die Vergabe des Gruppenschlüssels durch eine zentrale Kontrollstelle. Demgegenüber stehen dezentralisierte Verfahren, bei denen die Schlüsselgenerierung und -verteilung durch wechselnde Instanzen möglich ist. Darüber hinaus gibt es noch Verfahren mit verteilten Schlüsselvereinbarungen, wobei eine Aufteilung in Untergruppen stattfindet, welche voneinander unabhängige Gruppenschlüssel nutzen.

Gemäß der Forderung nach einem zentralisierten Verfahren wird nur diese Kategorie vertiefend betrachtet. Hierbei gibt es die folgenden drei Unterkategorien [CS08]:

- *Paarweise Schlüssel:* Übermittlung des Gruppenschlüssels durch die zentrale Instanz mittels individueller Teilnehmerkommunikation
- *Broadcast Geheimnis:* Übertragung des Gruppenschlüssels mittels Broadcast anstelle individueller Verbindungen
- *Hierarchische Schlüssel:* Einordnung der Teilnehmer in eine Baumstruktur mit entsprechenden kryptografischen Schlüsseln zur Verteilung der Gruppenschlüssel

Der bekannteste Vertreter der paarweisen Schlüssel ist das GKMP [HM97]. Bei diesem Protokoll speichert und teilt sich der zentrale Server einen geheimen Schlüssel mit jedem Gruppenmitglied. Dieser wird *Key-Encryption-Key* (KEK) genannt. Zur Verteilung des Gruppenschlüssels sendet der Server jedem Teilnehmer einzeln diesen Schlüssel individuell mit dem jeweiligen KEK überschlüsselt, was zu einem hohen Aufwand führt. Durch die mehrfache Datenübertragung von der zentralen Instanz zu den einzelnen Teilnehmern, sowohl bei der Erstellung einer Gruppe als auch bei jeder Veränderung der Gruppenkonstellation, sind viele Nachrichten notwendig. Dadurch entsteht eine erhebliche Belastung für den Server und der ohnehin geringen Netzkapazität des MANET.

Demgegenüber steht das Broadcast-basierte Verfahren SL [CC89, AM08], welches dem Server ermöglicht komplette *Re-Keying* Prozesse mit jeweils einer einzigen Broadcast-Nachricht durchzuführen. Es basiert auf dem *Chinese Remainder Theorem* (CRT) [XCD12], welches zur Erzeugung einer Verschlüsselung die Eigenschaften der Kongruenz ausnutzt. Jedoch ist die Berechnung des CRT im Vergleich zum GKMP noch aufwendiger, sodass dies für Endgeräte im Bereich von MANETs mit geringer Rechenleistung nur in begrenztem Umfang durchführbar ist.

Das alternative Verfahren LKH [Li12, Sa14] gehört zu den hierarchischen GKM Verfahren. Die Schlüssel und damit die Gruppenteilnehmer werden hierbei in einem für jede Gruppe eigens angelegten Binärbaum gepflegt. Jeder Knoten im Baum repräsentiert einen KEK, welcher den darunter liegenden Teilnehmern bekannt ist. Durch den Aufbau der Baumstruktur entsteht ein erhöhter Aufwand hinsichtlich der Verwaltung der vielen inneren Knoten sowie der Berechnung und Verteilung zugehöriger Schlüssel was nur im Fall eines Austritts einen mäßigen Vorteil bietet. Da es nicht bei jeder Verwendung zu dieser Operation kommt, ist dies unnötiger Aufwand. Zudem sind im Vergleich zum SL mehrere Nachrichten beim Austritt zu versenden, was zu einer erhöhten Netzlast führt.

4 CAKE - Hybrides Gruppen-Schlüssel-Management Verfahren

Für die gestellten Anforderungen im Bereich von MANETs wird folgendes Konzept zum Management der Gruppenschlüssel vorgeschlagen. Das neu entwickelte Verfahren *Central Authorized Key Extension* (CAKE) nutzt dazu einzelne Bestandteile der zuvor genannten Verfahren und kombiniert diese zu einem integrierten hybriden System. Gemäß den Anforderungen des Szenarios ist die Schlüsselverwaltung bei CAKE zentral organisiert.

Hierzu existiert eine autorisierte und vertrauenswürdige Instanz (AI), welche die Generierung, Verwaltung und Verteilung der Schlüssel übernimmt sowie notwendige Berechnungen durchführt. Jeder Teilnehmer N_i meldet sich am System an, indem dieser mit der AI ein privates Schlüsselpaar (KEK_i) aushandelt. Weiterhin generiert jeder Teilnehmer eine Primzahl m_i für ein CRT System. Mit diesen beiden Geheimnissen KEK_i und m_i ist ein Teilnehmer bei der AI im System CAKE angemeldet.

Neben den individuellen und persönlichen Schlüsseln umfasst CAKE noch einen *Group-Transmission-Encryption-Key* (GTEK), welcher für die eigentliche Kommunikation in der Gruppe verwendet wird. Hierzu muss jeder Teilnehmer einer Gruppe in Besitz dieses Schlüssels sein. Zusätzlich existiert ein *Group-Key-Encryption-Key* (GKEK). Dieser wird bei Bedarf zur Überschlüsselung des GTEK verwendet, sodass dieser gesichert an alle Kommunikationsteilnehmer verteilt werden kann. Zur Überschlüsselung werden die beiden Schlüssel GTEK und GKEK bitweise mit XOR verrechnet. Die Verwendung der XOR Operation bietet informationstheoretische Sicherheit gemäß dem One-Time-Pad.

4.1 Initiale Erzeugung einer Gruppe und des ersten GTEK

Bei der Erzeugung einer Gruppe generiert die AI zufällig einen GTEK und einen GKEK. Diese sind an alle Gruppenteilnehmer für die gesicherte Kommunikationsgruppe zu verteilen. Dazu berechnet die AI analog zum SL Verfahren initial ein CRT System und übermittelt die Daten mittels einer Broadcast-Nachricht. Hierbei werden von allen Teilnehmern der spezifizierten Gruppe die Werte m_i mit in die Berechnung des sogenannten Locks *MX* einbezogen. Ein Teilnehmer N_i kann das Lock gemäß dessen Prinzip nur auflösen, wenn der spezifische Wert m_i in der Berechnung enthalten ist. Die Empfänger der Nachricht erhalten als Ergebnis des CRT Systems den Schlüssel GKEK. Der Schlüssel GTEK wird durch bitweises XOR mit dem GKEK überschlüsselt und in einer weiteren Broadcast-Nachricht übermittelt. Folgende Bestandteile sind zur initialen Erzeugung einer Gruppe bei allen Gruppenmitgliedern notwendig: $\{GKEK\}_{Lock\,MX}$, $\{GTEK\}_{GKEK}$

Die Teilnehmer der Gruppe müssen zuerst das Lock *MX* auflösen, um den GKEK zu erhalten. Anschließend lässt sich der Schlüssel GTEK ermitteln. Somit haben alle Teilnehmer die Kenntnis über die einheitlichen Gruppenschlüssel GTEK und GKEK. Bei Bedarf kann die Nachricht entsprechend digital mit dem Zertifikat der AI signiert werden.

4.2 Eintritt von neuen Teilnehmern in eine Gruppe

Wenn nach der Gruppenerzeugung ein neuer Teilnehmer N_{i+1} an der gesicherten Gruppenkommunikation teilnehmen möchte, muss dieser zuerst die initialen Prozesse mit der AI durchlaufen. Dieser meldet sich dazu bei der AI des Systems CAKE an und tauscht die Schlüsselinformationen (KEK_{i+1}, m_{i+1}) aus. Anschließend wird ein Re-Keying für die bestehende Gruppe durchgeführt. Dazu generiert die AI den Schlüssel $GTEK_{neu}$ und $GKEK_{neu}$. Diese werden bitweise mit XOR unter Zuhilfenahme des gehashten $GKEK_{aktuell}$

überschlüsselt. Die entstandene Nachricht wird an alle bisherigen Gruppenteilnehmer gesendet. Für den neuen Teilnehmer N_{i+1} wird der $GTEK_{neu}$ und $GKEK_{neu}$ mit dem privaten Schlüssel KEK_{i+1} verschlüsselt und separat übertragen. Durch entsprechend zeitsynchrones Umschalten vom $GTEK_{aktuell}$ auf $GTEK_{neu}$ können alle Teilnehmer gesichert miteinander kommunizieren.

Bei einem Masseneintritt von mehreren neuen Teilnehmern in eine Gruppe wird äquivalent zum Eintreten eines neuen Teilnehmers verfahren. Alternativ lassen sich die neuen Gruppenteilnehmer über ein CRT zusammenfassen, sodass nur eine Nachricht für alle neuen Teilnehmer notwendig ist. Somit sind beim Eintreten eines neuen Teilnehmers in eine Gruppe zwei einfache Berechnungen (XOR und Verschlüsselung gemäß Verfahren) sowie zwei Broadcast-Nachrichten notwendig. Beim Eintreten mehrerer neuer Teilnehmer sind durch die Verwendung eines CRT Systems ebenfalls nur zwei Nachrichten notwendig. Bei der Verschmelzung bestehender Gruppen wird jeweils ein Re-Keying auf Grundlage der bestehenden GKEKs der Gruppen durchgeführt, wodurch nur entsprechend der Anzahl der zu verschmelzenden Gruppen Nachrichten notwendig sind.

4.3 Austritt von Teilnehmern aus eine Gruppe

Beim Austritt eines Teilnehmers aus einer Gruppe können die bestehenden GTEK und GKEK nicht genutzt werden, da der austretende Teilnehmer diese entschlüsseln kann. Eine erneute Initialisierung der Gruppe mittels CRT System ist aufgrund des Berechnungsaufwandes ebenso nicht praktikabel. Zur Reduktion des Aufwandes wird in CAKE ein verkleinertes CRT System angewendet und eine ternäre Baumstruktur eingesetzt, welche durch die AI verwaltet wird.

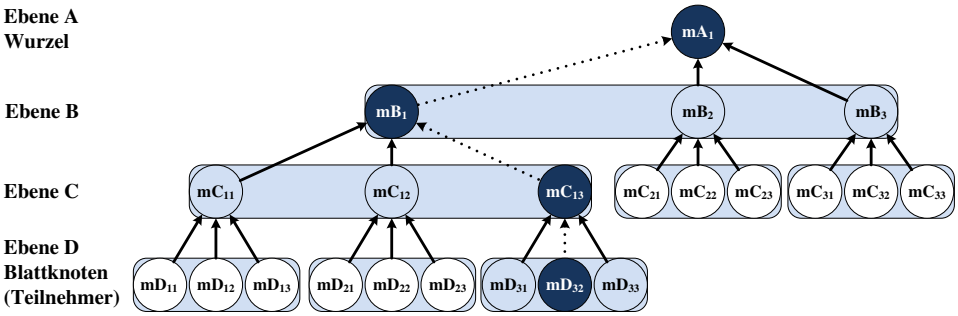


Abb. 2: Ternäre Baumstruktur zur Verwaltung der Schlüssel und zur Reduktion des Berechnungsaufwandes beim Austritt.

Abb. 2 verdeutlicht diese Baumstruktur, welche in Ebenen, beginnend mit A an der Wurzel, eingeteilt ist. Wie bei LKH entsprechen die Teilnehmer einer Gruppe mit deren m_i den Blattknoten des Baumes. Darüber hinaus entsprechen die Knoten weiteren KEKs, welche den darunter liegenden Teilnehmern bekannt sein müssen. Die Bezeichnung *mX* eines Knoten definiert ein spezifisches *m* für das CRT System, wobei das *X* die Ebene darstellt. Alle auf dem Pfad von der Wurzel zum Teilnehmer liegenden Schlüssel *m* müssen dem entsprechenden Teilnehmer bekannt sein, welche nur bei Bedarf initialisiert werden. Die

entsprechenden *mX* der Knoten entlang eines Pfades werden durch mehrere, kleine CRT Systeme von unten nach oben im Baum aufgebaut, sodass die Menge der Nachrichten gering gehalten wird. Baumstrukturen mit mehreren Unterknoten sind für größere Gruppen durch die flachere Struktur besser geeignet als Binärbäume. Mit jeweils maximal drei Unterknoten reduziert sich die Menge der zu berechnenden Locks für Gruppengrößen bis 81 Teilnehmer vorteilhaft. Im Bereich der betrachteten militärischen MANETs (selten mehr als 60 Teilnehmer) bietet sich somit die ternäre Baumstruktur an. Vorteilhaft ist ebenso, dass sich die Informationen zu den *mX* zu beliebigen Zeitpunkten bei geringer Netzbelastung aufbauen und versenden lassen.

Beim Austritt eines Teilnehmers wird der entsprechende Pfad von der Wurzel zum Blatt in der Baumstruktur markiert (in Abb. 2: Knoten mD_{32}, dunkel markiert). Alle auf dem markierten Pfad liegenden Schlüssel *mX* werden bei der folgenden CRT Berechnung nicht mit einbezogen. Für die Berechnung des notwendigen Locks *MX* des verkleinerten CRT Systems werden die jeweils neben einem markierten Knoten auf gleicher Ebene befindlichen *mX* verwendet (in Abb. 2: schraffiert gekennzeichnet). Diese stellen die Menge der in der Gruppe verbleibenden Teilnehmer dar, mittels welchen das neue Lock *MX* gebildet wird. Diese Menge hat wesentlich weniger Elemente als bei der Initialisierung der Gruppe und folglich ist der Rechenaufwand geringer. Die in die Berechnung einbezogenen Werte reduzieren sich von n-1 auf ($ln\ n_{max}$ / ln 3) * 2 (in Abb. 2: von 14 auf 6). Dabei entspricht n_{max} der Anzahl von Blättern bei einem vollbesetzten Baum. Somit ist nur eine Nachricht an die verbleibenden Teilnehmer zu senden. Die unter einem Teilbaum liegenden Teilnehmer können das Lock *MX* auflösen, da diesen der jeweilige *mX* auf dem Pfad bekannt ist. Nach Abschluss des Austrittes sind die *mX* auf dem dunkel markierten Pfad zu erneuern, indem die AI diese den Teilnehmern mitteilt. Bei einem Austritt mehrerer Teilnehmer oder einer Gruppenteilung sind vor der Berechnung in der Baumstruktur entsprechend mehrere Pfade zu markieren.

4.4 Re-Keying für eine Gruppe

Für das Re-Keying des GTEK einer Gruppe existieren zwei Möglichkeiten. In der ersten Variante generiert die AI oder ein beliebiger Gruppenteilnehmer neue $GTEK_{neu}$ und $GKEK_{neu}$. Diese werden jeweils mit dem gehashten $GKEK_{aktuell}$ bitweise XOR verschlüsselt und digital signiert. Das entstandene Datagramm wird anschließend an alle Gruppenteilnehmer per Broadcast verschickt. Durch die Spezifikation eines Zeitpunktes kann die Gruppe synchron auf den $GTEK_{neu}$ umschalten. Die zweite Variante sieht die Nutzung der individuellen Schlüssel der Teilnehmer bzw. des CRT Systems vor. Dabei verfährt die AI entsprechend des Konzeptes der initialen Erzeugung einer Gruppe bzw. des ersten GTEK.

4.5 Vergleichende Bewertung von CAKE

Tabelle 1 stellt das neu entwickelte Protokoll CAKE vergleichend gegenüber den bisherigen Verfahren dar. Es ist ersichtlich, dass CAKE bei der Erzeugung von Gruppen einen

vergleichbar hohen Berechnungsaufwand wie SL hat. Allerdings ergibt sich für die Erweiterung und Verkleinerung der Gruppe ein geringer Berechnungsaufwand im Vergleich zu allen anderen Systemen. In Abb. 3 ist der Vergleich des Berechnungsaufwandes anschaulich dargestellt, wobei für die Aufwandsbetrachtung entsprechend der Berechnungskomplexität die XOR-Operation mit Wertigkeit 1, die symmetrische Verschlüsselungsoperation mit Wertigkeit 2 und die CRT-Berechnung mit Wertigkeit 3 vereinfacht abgebildet sind. Darüber hinaus ist bei CAKE die Anzahl der Nachrichten gering, wodurch die Netzlast niedrig bleibt. Abb. 4 stellt diesen funktionalen Zusammenhang dar. Diese Vorteile des Systems werden durch einen Mehrbedarf an Schlüsseln und Speicherplatz für die Schlüssel erreicht. Da Schlüssel nur eine geringe Größe haben, ist dieser Nachteil vernachlässigbar. Diese Eigenschaften des Systems CAKE sind insbesondere im Bereich von gesicherter Funkkommunikation bei MANETs als vorteilhaft zu bewerten. Zusammenfassend entsteht durch den integrierten hybriden Ansatz ein Verfahren, welches den Berechnungsaufwand und die Anzahl zu übertragender Nachrichten minimiert.

Tab. 1: Gegenüberstellung verschiedener Gruppen-Schlüssel-Management Protokolle.

	Berechnungsaufwand			Nachrichtenanzahl			Bandbreite in L			Speicherbedarf in L	
	Initial	Eintritt	Austritt	Initial	Eintritt	Austritt	Initial	Eintritt	Austritt	Server	Teilnehmer
GKMP	nE	2E	(n-1)E	n	n+1	n-1	n	n+1	n-1	n	1+G
SL	nC	(n+1)C	(n-1)C	1			1			nG	2G
LKH	$2log_2$nE			$2log_2$n			$2log_2$n			(2n-1)*G	(2n-1)*G
CAKE	nC	XOR+E	2E *ln* n/*ln* 3	2			2			n+(1+3^k)*G	(2+k)*G+1

n ... Anzahl der Gruppenteilnehmer
G ... Anzahl der Gruppen eines Teilnehmers
C ... Teilnehmer der CRT Lock Berechnung
L ... Länge des Schlüssels
E ... Symmetrische Verschlüsselungsoperation
k ... Höhe des verwalteten Baumes

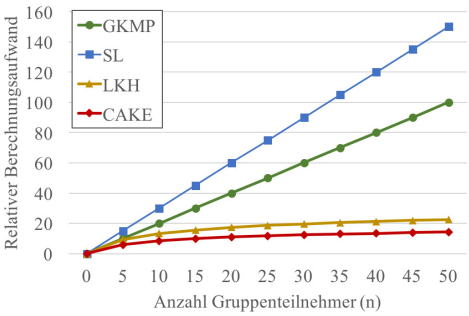


Abb. 3: Vergleich des durchschnittlichen Berechnungsaufwandes der betrachteten Verfahren

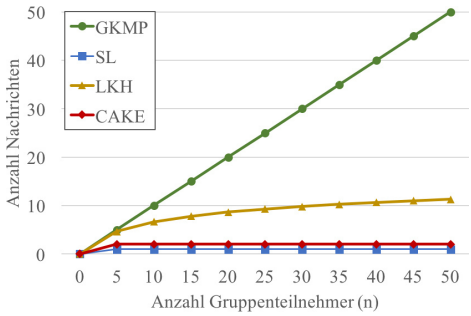


Abb. 4: Vergleich der durchschnittlichen Nachrichtenanzahl der betrachteten Verfahren

5 Bewertung der Sicherheit und Erläuterung des Designs

Im Folgenden wird die Sicherheit von CAKE hinsichtlich der in Abschnitt 2 erwähnten Sicherheitsanforderungen bewertet. Darüber hinaus erfolgen Erläuterungen zur Konstruktion des Sicherheitsdesigns.

Unterteilung in GKEK und GTEK
Ein separater GKEK zur Verbreitung eines neuen GTEK ist notwendig, da die Nutzung des aktuellen $GTEK_{aktuell}$ als $GKEK_{aktuell}$ bei bitweisen XOR dazu führt, dass der neue Teilnehmer die alten Nachrichten entschlüsseln könnte. Durch Anwendung von XOR auf die abgefangene Nachricht mit $GTEK_{neu}$ ergibt sich $GTEK_{aktuell}$. Dies wird durch den Einsatz eines separaten GKEK verhindert.

Forward- und Backward Secrecy
Backward Secrecy wird hierbei durch die Kombination der beiden Schlüssel GKEK und GTEK sowie dem CRT erreicht. Bei Eintritt von Teilnehmern werden die Schlüssel entsprechend neu generiert. Das CRT System beruht auf der Verwendung der persönlichen KEK der berechtigten Teilnehmer, sodass es für andere Teilnehmer nicht zu entschlüsseln ist. Bei Austritt von Teilnehmern oder einer Gruppenteilung findet der beschriebene Prozess zum Schlüsselwechsel statt, welcher die Forward Secrecy sicherstellt. Durch die Verwendung der ternären Baumstruktur in Verbindung mit dem reduzierten CRT System (Zwischenknoten im Baum) werden der GTEK und der GKEK in einem solchen Szenario mit vergleichbar geringem Berechnungsaufwand neu bestimmt und verteilt.

Im Zusammenhang mit dem gestellten Szenario ist die Sicherheit des allgemeinen Re-Keying-Prozesses durch die Nutzung des GKEK gewährleistet und der Aufwand gering gehalten. Im Falle von sogenannten “Man in the Middle“ Angriffen, bei denen einzelne Nachrichten abgefangen und verfälscht bzw. nicht weitergeleitet werden, bleibt der Nachrichteninhalt somit trotzdem geheim.

Schlüsselunabhängigkeit / Folgenlosigkeit
Durch den Einsatz einer XOR-Verknüpfung, welche nachweislich nicht zu dechiffrieren ist (vgl. [MM13], [BL12]), beim Verschlüsseln der GTEK und GKEK ist es auch nach Abfangen eines der benutzen Schlüssel nicht möglich auf den ursprünglichen zu schließen. Die voneinander unabhängige Erzeugung der einzelnen Schlüssel und die Verwendung von Hashfunktionen gewährleistet die Folgenlosigkeit.

Kollisionsfreiheit
Bei der Verschlüsselung von Nachrichten wird, gängigerweise unter Zuhilfenahme von Zufall, bei jeder Durchführung ein einzigartiger Key erstellt. Dies verhindert Rückschlüsse von bereits abgefangenen bzw. dechiffrierten Schlüsseln auf neu erlangte. Ein von nicht vertrauenswürdigen Teilnehmern initiierter Re-Keying Prozess wird durch die Struktur des Verfahrens verhindert, da dies nur mit Insiderwissen von der bestehenden Gruppe und der AI durchgeführt werden kann. Ferner ist eine Autorisierung durch die AI bei Eintritt eines Teilnehmers in das Kommunikationsnetz erforderlich. In Verbindung mit dem Einsatz von digitalen Signaturen ist die benötigte Sicherheit gewährleistet.

6 Zusammenfassung

Im Bereich von MANETs ist zur verschlüsselten Kommunikation die Verwendung von Ressourceneffizienten Verfahren essentiell. Mit CAKE wird in der vorliegenden Arbeit ein entsprechend den Anforderungen genügendes GKM Verfahren vorgestellt und mit dem

Stand der Technik verglichen. CAKE bietet die Möglichkeit bei geringem Berechnungsaufwand und einer niedrigen Belastung des Netzes, Schlüssel innerhalb einer Gruppe auszutauschen und effizient auf dynamische Veränderungen der Gruppe zu reagieren. Dabei ermöglicht CAKE stets eine vertrauliche Verteilung der Schlüssel und die Einhaltung der Anforderungen hinsichtlich Backward und Forward Secrecy. Auf Basis der aufgezeigten Betrachtungen ist es im nächsten Schritt notwendig, das entwickelte Verfahren praktisch im MANET einzusetzen.

Danksagung

Wir danken Sandro Passarelli für die Diskussionen und Anregung zum praktischen Szenario. Die Arbeit wurde durch das siebente Rahmenprogramm des Exzellenznetzwerkes (ICT-318488) über das Projekt FLAMINGO durch die europäische Kommission gefördert.

Literaturverzeichnis

[AM08] Antosh, C. J.; Mullins, B. E.: The Scalability of Secure Lock. In: IEEE International Performance, Computing, and Communications Conference. Jgg. 27, S. 507–512, 2008.

[BL12] Borowski, M.; Lesniewicz, M.: Modern usage of "old" one-time pad. In: Military Communications and Information Systems Conference (MCC). S. 1–5, 2012.

[Bu13] Bundesamt für Sicherheit in der Informationstechnik: IT-Grundschutz Katalog, M 2.46 Geeignetes Schlüsselmanagement. 2013.

[CC89] Chiou, Guang-Huei; Chen, Wen-Tsuen: Secure broadcasting using the secure lock. IEEE Transactions on Software Engineering, 15(8):929–934, 1989.

[CS08] Challal, Yacine; Seba, Hamida: Group Key Management Protocols: A Novel Taxonomy. International Journal of Computer, Electrical, Automation, Control and Information Engineering, 2(10):3620 – 3633, 2008.

[HM97] Harney, H.; Muckenhirn, C.: Group Key Management Protocol (GKMP) Specification. Bericht 2093, Internet Engineering Task Force, 1997.

[Li12] Liu, Zenghui; Lai, Yingxu; Ren, Xubo; Bu, Shupo: An Efficient LKH Tree Balancing Algorithm for Group Key Management. In: Proceedings of the International Conference on Control Engineering and Communication Technology (ICCECT). Jgg. 10. IEEE Computer Society, S. 1003–1005, 2012.

[MM13] Matt, C.; Maurer, U.: The one-time pad revisited. In: IEEE International Symposium on Information Theory Proceedings (ISIT). Jgg. 11, S. 2706–2710, 2013.

[Ra00] Rafaeli, Sandro: A Decentralised Architecture for Group Key Management. Bericht, Lancaster University, 2000.

[Sa14] Sakamoto, N.: An efficient structure for LKH key tree on secure multicast communications. In: IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). Jgg. 15, S. 1–7, 2014.

[XCD12] Xu, Guoyu; Chen, Xingyuan; Du, Xuehui: Chinese Remainder Theorem based DTN group key management. In: IEEE International Communication Technology (ICCT). Jgg. 14, S. 779–783, 2012.

Extending the OAuth2 Workflow to Audit Data Usage for Users and Service Providers in a Cooperative Scenario

Marius Politze[1], Bernd Decker[2]

Abstract: The increasing amount and heterogeneity of devices demands changes in IT infrastructure. Many web service architectures used to meet these demands use the OAuth2 workflow to secure their interfaces. These implementations usually tightly couple web services and an OAuth2 authorization service. The presented extension to the OAuth2 workflow is capable handling authorizations for multiple attached services and therefore combines existing services of a central IT service provider but also allows other services running in a cooperative model with only a single instance of the authorization server. Based on auditing parameters it is possible to present access per resource or per method giving service providers and application developers more insight in how their services are used and show users by whom their personal data is used.

Keywords: authorization, micro services, mobile, OAuth2, privacy, security, SOA, transparency

1 Introduction

Globalization and digitalization pose continuous challenges to established processes in government and administration. The standardization of existing processes to improve cooperation between institutions, reduce overall costs or increase efficiency is a routine at several levels: global, national and local. Even at the local level compatibility and reusability of available components is key to meet these challenges.

Due to Increased mobility and the rising number of students the universities have to standardize existing processes, improve cooperation between institutions, reduce overall costs or increase efficiency. In addition, students' demands on universities and their employees have changed. Previously, central IT service providers introduced various processes and supporting IT infrastructure and applications to meet these demands. Not only IT infrastructure and applications are becoming more important to the universities' processes and employees but to students and their daily lives [Ju09]. This leads to increased competition among universities to present most appealing services to their students. Several research groups and IT service providers have recognized this and IT service providers and lead to improvement of existing services [Mi12] as well as new methods in designing future IT services [KL16].

1 IT Center RWTH Aachen University, Seffenter Weg 23 52074 Aachen, politze@itc.rwth-aachen.de
2 IT Center RWTH Aachen University, Seffenter Weg 23 52074 Aachen, decker@itc.rwth-aachen.de

1.1 Problem statement

Modern process supporting information systems usually do not consist of monolithic structures but of loosely coupled services. Each of these components is responsible only for certain steps within the supported processes. From the software engineering point of view, this kind of micro service architecture has several advantages like easier maintainability, expendability and replaceability of the components used.

The processes base on personal data passed between different services within the infrastructure. As the coupling of services decreases, so does the control of transferred, processed or saved date in the different parts of the process. With the OAuth2 workflow service providers and users can identify a subset of information that they allow be exchanged between the different steps of the process. However, it remains opaque which information the application actually uses.

The OAuth2 workflow forms an integral part of most of these processes. The management of authorizations already collects granted authorizations for users and services as well as metadata, like the application that requested the authorization. This allows using the OAuth2 workflow as a starting point to gain more transparency on how applications use personal data for service providers and users.

1.2 The OAuth2 Workflow

The OAuth2 workflow is described in RFC 6749 [Ha12] allows secured, personalized access to web services or resources and handles the users' authorization without supplying credentials to the application itself. This also paves the way for third party developers accessing central IT services. Generally, it follows the steps 1-4 shown in Figure 1.

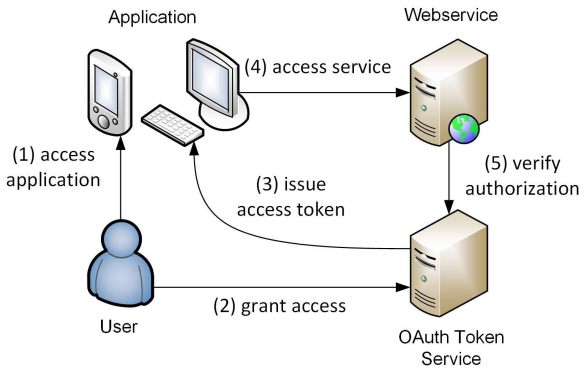


Figure 1: Schematic of the OAuth2 workflow

At first, the user accesses the application (1). To access the web service resources for the user, the application needs an access token. Therefore, the application directs the user to the token service where the user logs in and grant access for the application (2). The token service then issues the access token (3). The application can now use the token to access the web service resources (4). The web service now has to verify the authorization (5). Most implementations tightly couple web services and token services, so this step remains internal. To decrease coupling of the services, explicitly modelling of this step is a key part of the cooperative workflow proposed in the following sections.

2 Current Implementation

In order to access the services currently available, third party developers have to perform a simple registration process. Apart from a contact name and email address, they have to supply the use cases their app covers and what data it needs from the services to perform these tasks. The register of all applications is publicly visible to all users. Third party developers extend the functionality of existing IT systems and applications using the OAuth2 workflow.

Furthermore, the implemented workflow is capable of handling authorizations for multiple attached services. Central IT service providers but also for other services running in the university context use it cooperatively. Allowing not only the reuse of already established infrastructure but also the use of OAuth2 authorization for inter service communication while providing full transparency to the user.

2.1 Existing Service Oriented Architectures

Initially RWTHApp [PD14] demanded an infrastructure to make data provided by legacy systems accessible from smart devices in a secure and consistent fashion. Nevertheless, some rather traditional web applications are already taking advantage of the centralized implementation of the infrastructure. Even though these services centralize access to legacy systems, it builds upon some of the paradigms introduced by micro services: primarily the loose coupling of functional units to access the independent legacy systems. These functional units are centrally published and appear as a single web service instance using consistent access and naming conventions.

Apart from security and consistency, the design of the infrastructure also tries to increase availability and speed of the legacy systems by introducing automated and semi-automated caching layers in order to increase overall performance and improve user experience. The cache uses a probabilistic, proactive model to predict future service calls [PSD16]. While most legacy systems require some kind of integrated authentication infrastructure to handle authorization and data access, most modern systems offer services that are more flexible and allow delegation of authorization. This allows integration into the OAuth2 workflow using the cooperative model. Using this general architecture, several services in

the field of campus management, student lifecycle, e-learning and other university services like canteen menus and university library are already available.

2.2 The Cooperative OAuth2 Model

Since most application developers use OAuth2 to authorize a client application to run in a user context, also implementations of the OAuth2 workflow focus on client side authorization. Many major IT companies like Google, Facebook and GitHub offer to authorize the use of their services using an OAuth2 workflow. However, each of these companies offer their own authorization service that depends on the services offered and vice versa.

To allow authorization within a cooperation of various service providers the OAuth2 workflow needs to offer the means to verify that a certain access token is valid and to identify the application and the user that requested the token. The *4*-tuples of validity, application, identity, and service provider form the context in which the token may be used. The main features of the cooperative model are:

- there is only a single instance of the authorization server
- all service providers are known to the authorization server
- the identity information needed by the service providers is known to the authorization server

Provided a token by the application, the service provider can therefore resolve the context of the token from the authorization server. It is furthermore easy to add new service providers to the cooperation to extend the features of the interfaces offered to the users. It is however important that tokens issued before adding a service remain invalid for the new service until the user authorizes a token for the service.

This model relates to the authentication via authorization workflow used by many current mobile and internet applications. However, there is a major difference: The authentication via authorization workflow usually uses some sort of user information service to identify the user. Consequently, the user authorizes the application only to access this service and not the actual services offered by the application vendor. This leads to major security flaws [YLL16]. The cooperative model resolves these issues by explicitly authorizing specific services within the cooperation.

2.3 Implementation of the Cooperative Model

The initial OAuth2 workflow uses four endpoints to perform the authorization of the tokens:

- *Authorize*
  The endpoint is used to authorize tokens for server side and web applications.

- *Code*
  The endpoint is used to request codes that can be shown to the user in order to authorize an installed application.
- *Token*
  The endpoint is used to manipulate access tokens during and after the authorization process. The manipulations include extending the lifetime of the token or invalidating a token.
- *TokenInfo*
  The endpoint supplies information about a token. Applications can use the endpoint to verify that the token is valid and actually belongs to the application.

With the exception of the Authorize endpoint, all endpoints are REST web service methods that are called using a HTTPS POST request. In the cooperative model, the new context endpoint extends this set:

- *Context*
  The endpoint is used to resolve the context of a token for a certain service provider.

Again the context endpoint is a REST web service that is called using HTTPS POST. If the token is valid for the service provider calling the method, it returns the *4*-tuple of the context.

3 Data Usage Audits in the OAuth2 Workflow

In order to achieve more transparency for the user, the cooperative OAuth2 workflow as shown in Figure 2 needs further extensions. When verifying the context of the token, the service providers may add auditing information to their request. The authorization server then processes and aggregates the information. While most service providers are very well capable of collecting such data, they rarely make it available to users or application developers. This workflow therefore presents a user centric way to provide more transparency on the usage of personal data.

3.1 Auditing in the Cooperative Workflow

Service providers have to call the context endpoint to validate the token essentially every time an application requests a resource. At this point, it is possible to log the number of occurrences of the *4*-tuples that describe the context. Information from this audit log, allows producing an overview for users on which applications and services were active due to their authorization.

Service providers and application developers may also access auditing information. This does not only allow the deduplication of information but also removes some necessity to

save personal data on remote locations. After all enforcement of laws and best practices concerning personal data and privacy on a single central system serving the OAuth2 workflow is easier.

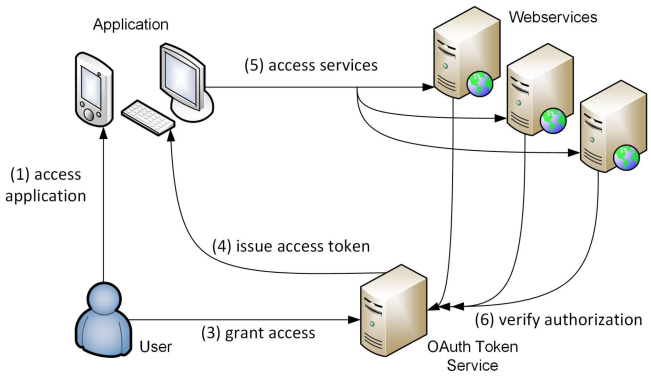


Figure 2: Schematic of the cooperative workflow

3.2 Further Extending the Cooperative Workflow

In the initial model, the context does not include the actual resource accessed by the user. To make auditing more expressive, the context endpoint is extended. Service providers may therefore append additional auditing data to context requests. Basing on the *4*-tuple of the context this information is added. While it is generally possible to extend the *4*-tuple to an *n*-tuple of arbitrary length, the additionally logged fields for auditing are limited to preserve performance of the actual authorization workflow. The auditing system allows appending the following auditing parameters, all of which are optional:

- *Resource*
  The actual resource requested by the user. For most RESTful web services this is equivalent to the URL requested by the application.
- *Operation*
  The kind of operation performed on the resource. For most RESTful web services this is equivalent to the HTTP method used by the application.
- *Cost*
  The cost caused by the call in terms of computing resources. It is best practice that the service provider checks the authorization before taking any actions. Even though real costs are generally unknown a priori, this allows the service provider to supply an estimate based on the resource and the operation.

This extension allows service providers and users to gain insight on how applications use resources and personal data. Again all auditing information collected by the authorization

server are subject to personal data and privacy laws and operators can and should strictly enforce them this point.

4 Auditing Usage of Personal Data

As previously shown both variants of the cooperative workflow generally deliver auditable information. However, it is important to note that insights for the users as well as service providers and application developers are more valuable if additional auditing information is available. It is therefore important that the service providers join in and supply the additional information needed.

4.1 For Service Providers

The auditing information presented to the service provider aim at giving an overview of how their services and resources are used. The basic variant of the cooperative workflow however does not allow resolving information for individual resources but can only distinguish between different applications accessing the resources.

Even though they are not the primary target audience for the auditing, service providers need to be convinced to provide the additional information. Using the extended workflow, service provider gain additional value by partitioning the information further. Based on the auditing parameters it is now possible to present access per resource or per method as shown in Figure 3 giving service providers more insight in how their services are used. Furthermore, the cost parameter allows identifying applications that put more excessive load on the services than others and therefore may provide an indication if applications are abusing the services offered.

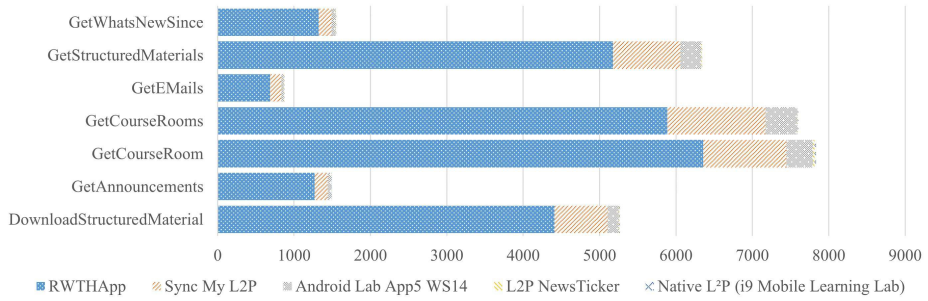


Figure 3: Selection of endpoints offered by the eLearning system used by different applications.

4.2 For Application Developers

In general, it is very common for application developers to retrieve information on how their applications are actually used. Application platforms like Android and iOS already offer detailed reports on the kind and number of devices running the application. In addition, most of these platforms offer usage reports that allow advanced analysis of user behavior.

Even though these tools are sophisticated, their use is debatable due to privacy concerns by the users. To overcome the need to include such tools in their application and protect the users' privacy, application developers can access some insights from the auditing. This especially includes the number of users who used the application as well as the number of requests issued to the different endpoints and service providers as shown in Figure 4.

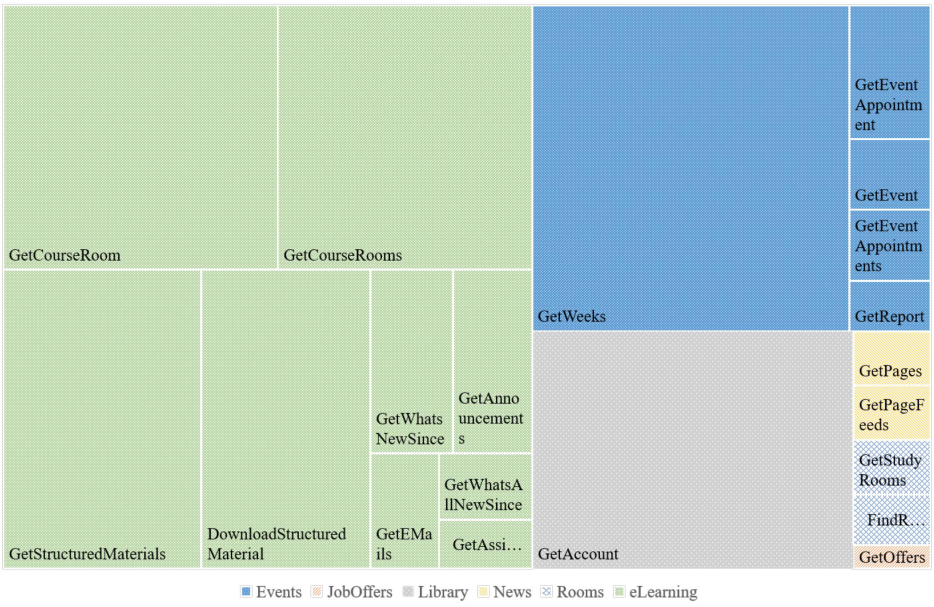


Figure 4: Treemap of a selection of endpoints and service providers called by RWTHApp. The area of the rectangles is relative to the frequency of calling the endpoint.

4.3 For Users

The focus group of the auditing are the users using the services. This is extremely important, as they most certainly have no other means of collecting and auditing how applications access their personal information. Users can therefore gain insight into their

full profile. They see when an application requested a certain resource. An example of such a usage profile is shown in Figure 5.

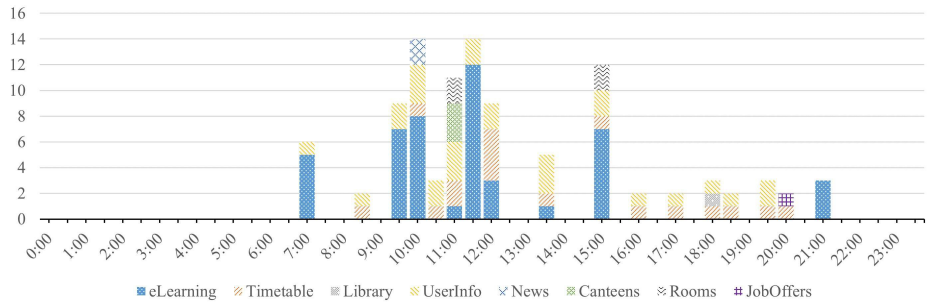


Figure 5: Overview for one user showing the different service providers used by RWTHApp.

To make the information collected for the auditing even more transparent; the users can access to their raw *n*-tuples saved for the auditing process. For concerned users this allows an in depth analysis by whom their data is used. In practice, this information is not saved permanently: To adhere with privacy best practices the data should be anonymized in regular intervals. In the current implementation, this limits users to access their history of the past two weeks.

5 Conclusion

The OAuth2 workflow and the cooperative model allow the introduction of a centralized auditing structure. Apart from providing reports to service providers this approach allows making all information available to the user and therefore leads to a more user centric and transparent auditing. By centrally collecting the auditing information, operators can control the enforcement of data privacy laws and best practices in a better way.

To add more value to reports for users, service providers need to supply additional information. Using the proposed extended cooperative workflow as a standard, lowers hurdles that service providers have to take. It offers additional value by also providing access to reports for service providers.

6 Future Work

The most recent implementation collects auditing data according to the extended cooperative workflow. Monthly reports then condensed the information. However, it is currently not possible for service providers nor users to access their current data. In order to achieve this a more interactive application needs to be built. Dashboards like these are

very common in smartphone app stores and analytics tools, a source of information that is, however, usually not available for uses but only for application developers.

The cooperative model only presents one of the possible extensions in the OAuth2 workflow. In future applications may use OAuth2 outside of these boundaries: While application developers are typically from third parties, users and service providers from outside the cooperation cannot access the current infrastructure, per se. As more and more services and service providers are shared across university boundaries, future extensions are needed to transition from the cooperative to a federative model for OAuth2 workflows.

Literature

[Ha12] D. Hardt, The OAuth 2.0 Authorization Framework: RFC Editor, 2012.

[Ju09] W. Juling, Vom Rechnernetz zu e-Science, PIK - Praxis der Informationsverarbeitung und Kommunikation, vol. 32, no. 1, 2009.

[KL16] P. Kupila and U. Lehtonen, Engaging students in building better digital services, in 22nd EUNIS Congress, Thessaloniki, 2016, pp. 126–129.

[Mi14] J. Mincer-Daszkiewicz, We Publish, You Subscribe — Hubbub as a Natural Habitat for Students and Academic Teachers, in 20th EUNIS Congress, Umea, 2014.

[PD14] M. Politze and B. Decker, RWTHApp: from a requirements analysis to a service oriented architecture for secure mobile access to personalized data, 20th EUNIS Congress, Umea, 2014.

[PS16] M. Politze, S. Schaffert, B. Decker. A secure infrastructure for mobile blended learning applications, European Journal of Higher Education IT 2016-1

[YLL16] R. Yang, W. C. Lau, and T. Liu, Signing into One Billion Mobile App Accounts Effortlessly with OAuth2.0, in Black Hat Europe, 2016.

X.509 User Certificate-based Two-Factor Authentication for Web Applications

Thomas Zink, Marcel Waldvogel[1]

Abstract: An appealing property to researchers, educators, and students is the openness of the physical environment and IT infrastructure of their organizations. However, to the IT administration, this creates challenges way beyond those of a single-purpose business or administration. Especially the personally identifiable information or the power of the critical functions behind these logins, such as financial transactions or manipulating user accounts, require extra protection in the heterogeneous educational environment with single-sign-on. However, most web-based environments still lack a reasonable second-factor protection or at least the enforcement of it for privileged operations without hindering normal usage.

In this paper we introduce a novel and surprisingly simple yet extremely flexible way to implement two-factor authentication based on X.509 user certificates in web applications. Our solution requires only a few lines of code in web server configuration and none in the application source code for basic protection. Furthermore, since it is based on X.509 certificates, it can be easily combined with smartcards or USB cryptotokens to further enhance security.

Keywords: multi-factor-authentication, authentication, crypto token, S/MIME, certificate, X.509

1 Introduction

At the time of this writing, the most commonly used user authentication scheme for login to digital services of all kinds is still a combination of username and password. If an attacker guesses or steals the users password he can effectively steal the users digital identity, access the user's personal information, and perform actions in the user's name. The password is often the weakest link[FH07].

For many applications this level of security might be enough. But sensitive information or the ability to perform critical actions require additional layers of security. Multi-factor authentication (MFA) is a methodology that introduces additional, independent authentication factors that all need to be validated when authenticating a user. In some application domains, like e-banking, multi-factor authentication is already an established mechanism. Many well-known service providers have also adopted 2FA (two-factor authentication, often called '2-step verification'), including Apple, Google, Microsoft, and even Steam.

The factors differ in nature and include:

- Something you know (knowledge of a secret, like a password or PIN)

[1] University of Konstanz, 78457 Konstanz, <first.last>@uni.kn

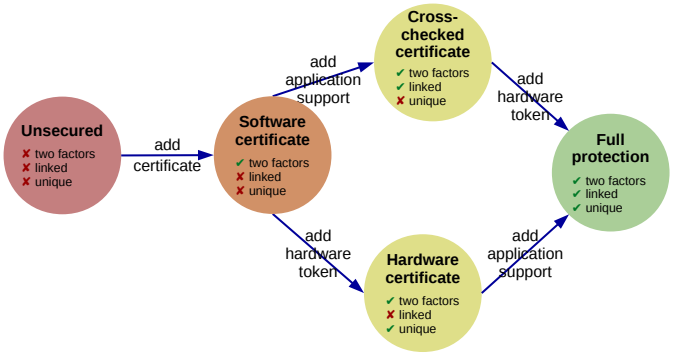


Figure 1: The different stages of securing a web application with certificate-based second factors. Adding a software certificate provides an additional factor without linking it to the users identity. Application support is required to link users with their certificate. Instead of storing the certificate in software, it can be stored on a hardware token, to guarantee it stays private.

- Something you have (possession of an object, like a smart card or cryptographic token, aka cryptoken)
- Something you are (physical features, like fingerprint or retinal patterns)

Independence of the authentication factors is critical to ensure that a compromise of a single factor does not affect the others. Although the authentication factors are independent, their affiliation to the authenticating identity must be apparent and verified by the service. In fact, many authentication schemes, that claim to offer MFA, actually fail to associate all authentication factors with the same single identity, as discussed in Section 3.

MFA introduces additional complexity in applications as well as identity management [Si16]. Many standard services like LDAP or AD do not support MFA out of the box. As a result, additional authentication services and user management is usually required.

In this paper, we suggest another unique form of two-factor authentication for web applications that requires little changes in the application and builds on existing identity management infrastructures. We utilize standard X.509 user certificates (aka S/MIME certificate) as proof for the user's identity and as a first factor for authentication. After the user has provided his certificate and thus proven his identity he is prompted for his password. This scheme can easily be combined with standard hardware solutions like smartcards or USB cryptokens (Figure 1). It is surprisingly easy to implement, yet affective and user friendly, and can be adapted quickly and flexibly for any type of web application without the need for another online service.

2 Related Work

For many decades now, passwords have been considered a necessary evil. Already in 1984, Thompson[Th84] noted that password checking can easily be circumvented. A few years

later, Muffett[Mu91] published *Crack* which allowed brute-force cracking of passwords (and enabled system administrators to identify passwords vulnerable to this attack). The primary target of network sniffing, keyloggers and several trojans[MSK05] also were passwords. Herley and Florêncio[HF06] present a non-technical way to fool keyloggers by interspersing the password with random character sequences which are entered when the password field does not have the input focus.

Given the success of break-ins to steal passwords by the million and even billion in recent years[Th16]. The requirement of some organizations to force their users to regularly change their password without any exposure has led users to increasingly chose weak passwords, write them on Post-Its, or share them between multiple accounts instead of using password managers[FH07, Cr16]. Even passwords stored in encrypted form are worthwhile, as the users often use low-entropy passwords and share them between several accounts, reinforcing the importance of the weakest link.

Countermeasures including one-time passwords in software[Ha95] or hardware tokens [BLP03] have been introduced to overcome the password weakness. However, they have only been deployed in selected environments, as widespread compatibility to existing software has been limited, especially in open and heterogeneous environments such as those seen in research and higher education.

There, various forms of authentication over several protocols (local, LDAP, AD, web-based, ...) are used, often in combination to ensure access even if some systems have failed. For services offered among institutions of higher education, Shibboleth is now becoming the de-facto standard, even beyond web applications[Si12]. Work is under way to include two-factor authentication into Shibboleth[Si16]. This currently requires use of Shibboleth, which is not common for in-house applications yet and requires configuration at the Identity Provider, not (only) at the Service Provider.

3 State of the Art

Support for Multi-factor authentication in daily applications is steadily increasing. Amazon, like many banks, sends SMS text messages to the mobile phone. Apple relies on the ARM processor's TrustZone to keep keys from prying hackers. Github uses TOTP (temporal OTP) using a smartphone authenticator application, it also allows U2F using YubiKey. Similarly, Google also makes use of TOTP and U2F.

TOTP solutions are recommended to be coupled with a central server to avoid reuse of the same token for different services. This creates a dependence on additional servers and the network. Given that this solution is aimed at (distributed) administration in a heterogeneous network, a partial network or service downtime might prevent other systems from running.

LinOTP (and it's fork PrivacyIdea) is a generic OTP solution for Linux and can also integrate into the web server. However, it requires a separate user management and modifications to the application to link authenticating users to tokens. Without these changes, the

assumed two-factor authentication degrades to two single-factor authentications of possibly distinct users. Moreover, LinOTP provides a self provisioning service for users. While this enables users to individually configure a second authentication factor, the service itself is only secured with the users password, undermining the whole 2FA.

X.509 Client certificates for authentication do not see widespread distribution, at least in consumer products. They are mostly found in areas with restricted access, usually in the form of smartcards. Web server support for client certificates has been around for years, however, the common use case is what we call a 'certificate wall'. That is, access to a resource on the server is only granted with a valid certificate signed by a specific authority. Again, this is not a valid form of multi-factor authentication, since the identity of the user is not crosschecked with the identity provided by the client certificate.

4 Threat Model

Our solution was triggered by the requirements of the following two high-risk groups:

Help-desk support personnel. To underline the openness of higher education institution, face-to-face support help desks are increasingly used, often residing in a public space such as the library. The opening times often extend beyond the staffing period of the desk.

Therefore, the machines will be left unattended overnight and could be tampered with, including installing software or hardware keyloggers. Physically locking the machines away daily would require significant effort.

Furthermore, the personnel operating these machines often has significant privileges, such as creating a new user or resetting a user's password.

Roaming system administrators. In a user-friendly setting with diverse client setups throughout an education and research organization, support staff and system administrators will have to visit users at their desk. Often, some configuration steps have to be performed from these computers. However, it remains unclear whether the machine is clean from trojans or keyloggers.

It can generally be assumed by the user in front of the machine that the machine has not been tampered with. However, a small risk remains.

Similar thinking generalizes this to personal machines (laptops or desktops) in office rooms. An intruder might have stealthily broken into the room and modified the computer, i.e., performing an *Evil Maid* attack[Sc09]. Even a machine of a vigilant system administrator can be infected by malware, especially in a targeted attack. The risk there is even smaller, but remains[1].

Our threat model therefore includes

[1] The analysis of one local incident suggests that this has happened at least once.

- potentially untrusted terminals (public, customer, possibly hacked, keylogger),
- operated by a small, controlled user group with elevated privileges,
- logging in to (one or several) web services with elevated privileges.

The goals therefore are to

1. avoid additional effort on this group for unprivileged operations,
2. minimize the impact for privileged operations, and
3. prevent stealing of credentials with long-term validity required to perform these privileged operations, while
4. requiring minimal intervention to those web services.

5 Solution

Our solution is based on the observation, that the vast majority of web applications use the user's email address – or the local part of the email address – as the user's login name. This holds for most public web services, like Amazon or eBay, but also for many public or private organizations and companies that operate their own email infrastructure. Usually, the user's email user name/address is also his unique identifier (UID) used for organization-wide authentication. If the email address is not used for authentication in the organization, another UID scheme usually is in place. X.509 certificates must also be unique for a user and often demand the usage of a 'distinguished name' as subject. This subject can also include the UID, which can then be used for authentication instead of the user's email address.

Many organizations also either have their own certificate authority, or use the services of a public or commercial certificate authority. Especially the public authorities are likely to operate their own public certificate authority for use in government agencies or educational institutions.

While client certificates slowly take off, particularly in areas where trust or confidentiality is crucial, x.509 certificates are still usually found on servers and used for host authentication and SSL/TLS encryption. Because of this, support for x.509 client certificates has long been neglected. However, interest in stronger authentication schemes and x.509 client certificates as well as cybersecurity in general has risen in recent years. This is due to a combination of events, including Snowden's leak of thousands of confidential NSA documents and well-known attacks on cloud providers (e.g. Apple's iCloud "celebrity breaches"). As a result application developers and service providers have started to adopt a variety of multi-factor authentication schemes, from relying on trusted devices (Apple) to hardware tokens (Amazon, Github).

Still, aside from specialized applications that use smartcard authentication, x.509 client certificates are usually only used on web servers to protect locations or directories. For

example, anyone providing a valid certificate signed by a specific authority is allowed to access the login screen of an application (figure 2). While this adds another authentication factor this cannot be considered 2FA, since there is no process that actually validates that the owner of the certificate is the same entity that tries to login to the application. The reason for this is because the certificate request is issued by the web server while the login is requested by the application.

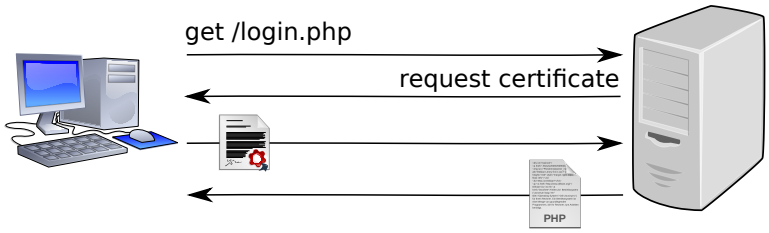


Figure 2: A typical certificate "wall". The server requests a valid certificate from the client to allow access to a location.

5.1 2FA with client certificates

In order to enable true multi-factor authentication the application needs to check the identity of the client certificate and somehow match that to the identity of the user trying to login. This requires additional effort in identity management to provide this missing link.

We observed, that for most web-based applications, users login with their email address or the local-part of their email address. Since x.509 user certificates are issued for a specific email address, it is surprisingly easy to extract the user name directly from the certificate on the server and use this as a validated remote user for the application. If the web server supports client certificate parsing, this can even be done directly on the web server. If not, the web server needs to expose the client certificate to the application which in turn parses the certificate and extracts the user name.

Apache2 supports the necessary options since version 2.4.12. Prior to that, the client certificate must be exposed to the application. Listing 1 shows the configuration for apache2. The important part is the export of the certificate and the environment variables.

List. 1: Apache config prior to version 2.4

```
<VirtualHost *:443>
...
SSLOptions +ExportCertData +StdEnvVars
...
SSLVerifyClient require
SSLVerifyDepth 3
SSLRequire ( %{SSL_CIPHER} !~ m/^(EXP|NULL)-/ )
...
</VirtualHost>
```

On the application side, parsing the client certificate and extracting the email address can be achieved in few lines of code (listing 2).

List. 2: Email extraction with apache < 2.4.12

```
$clientcert = openssl_x.509_parse($_SERVER['SSL_CLIENT_CERT']);
$subjectaltname = $clientcert['extensions']['subjectAltName'];
$email = substr($subjectaltname,strpos($subjectaltname,":")+1);
```

With apache2 version 2.4.12 or later, extraction of the user's email address can be done with a single line of code (listing 3).

List. 3: Email extraction with apache > 2.12.24

```
$email = $_SERVER['SSL_CLIENT_SAN_Email_0'];
```

In any case, listing 2 shows the appropriate login form for the application. The form uses the formerly extracted email address as readonly value for the user name. The user is thus not able to actively change the login name. Since the web server verifies the authenticity of the client certificate, the user's email address is also verified.

List. 4: Login form with readonly user name

```
<form>
<label><b>Username: </b></label>
<input type='text' name='user' value='<?php echo $email; ?>' readonly /> <
    br/>

<label><b>Password: </b></label>
<input type="password" placeholder="Enter␣Password" name="psw" required />

<button type="submit">Login</button>
<button type="button">Cancel</button>
</form>
```

5.2 Securing web applications

The proposed scheme can be used to efficiently secure any kind of web application with multiple authentication factors. A remaining issue is the storage location of the client certificate itself. Saving the certificate in the browsers store exposes the private key to a number of attacks that might lead to it's loss.

Fortunately, our solution can be easily combined with hardware tokens like smartcards or USB cryptokens to ensure that the private key stays private, even if the user's machine is compromised. Not all browsers support hardware-based certificate stores, though. Depending on the required trust level, the application can be secured accordingly.

5.2.1 Example use cases

The ease with which our authentication scheme can be implemented allows many applications.

Web Server or Proxy Using http auth and authentication modules, the whole web server can be secured. To quickly add 2FA support to multiple applications or servers, an authentication proxy could even be used.

Locations/Specific Applications Secure only specific locations or applications served by the web server, using true two-factor authentication.

Specific Actions Using the option `QUERY_STRING` to match post parameters or keywords in the URL, additional authentication factors can be requested for these specific actions.

6 Analysis

Our solution perfectly fulfills the goals stated in Section 4, as follows:

1. Unprivileged users or operations incur no extra burden.
2. Privileged operations are performed with zero effort for the basic (*software certificate*) case of Figure 1 and only a little bit more effort for the advanced protection modes. This applies to both setup and actual use.
3. The credentials that can be stolen do not provide any long-term access to privileged operations. If the computer is infected and under carefully targeted control, the current session may be hijacked; but there is no known general protection against this risk.
4. The web application hidden behind the 2FA does not need to be modified. Minimal changes are required to the web server (or a proxy inserted) for a first strong protection step.

As administrative commands can be identified by URL path or parameters, environment-specific definitions of what is considered a privileged operation can be easily applied. As access to these URLs is only granted to specific certificates, our 2FA solution even protects against several implementation or verification errors inside the application itself. As no online verification is needed with a central service (unlike other services like U2F or linOTP), this system does not increase the damage caused by a partial network or service outage at the university. The user certificates distributed for this purpose can be used to secure other services as well, including email, or vice versa, if policy allows. Then two birds are caught with one stone.

7 Evaluation

We successfully implemented our client certificate-based authentication in our support front-end application used by supporters in our public help desk center. The workstations used to login are publicly exposed and leakage of passwords is a real threat. We already use the DFN PKI for x.509 certificates which allowed us an easy transition towards the certificate authentication.

Each support staff member receives a USB cryptoken to store his personal certificate. The workstations' browsers are configured to access the certificate on the token, which is secured by a person identification number (PIN). To successfully log into the support front-end the supporter has to provide the token, his PIN and then his personal password.

We conducted a user study among the support staff to evaluate the quality of the solution. Unfortunately, only six staff members reacted. However, according to a study by Nielsen [NL93] a small sample is enough to identify a majority of usability problems.

Table 1 shows questions about the awareness of the user regarding the security of his password and the offered solution.

Table 1: Questions regarding user awareness

Question	**Yes**	**No**
Do you have to enter your password on untrusted devices	83%	17%
Do you fear that your password is in danger	83%	17%
Do you think the solution is justified	66.7%	33.3%

We also wanted to know the impact the solution has on the users' daily work and how well it is perceived (table 2).

Table 2: Questions regarding user experience

Question	**not**	**slightly**	**moderate**	**very**
How much does the solution disrupt your work	16.7%	16.7%	16.7%	50%
How satisfied are you with the solution	16.7%	16.7%	16.7%	50%

Finally, we asked if the users know other methods for multi-factor authentication, and if they perceive our solution as better, worse oder equally good.

Table 3: Comparison to other solutions

solution	**don't know**	**better**	**worse**	**equal**
Smartcard	16.7%	16.7%	0%	66.7%
SMS-TAN	0%	16.7%	83.3%	0%
U2F	16.7%	16.7%	0%	66.7%
HOTP/TOTP	16.7%	16.7%	33.3%	33.3%

In summary, most users are aware that their credentials are in danger and that additional security measures are justified. However, users feel that the additional security impedes

their daily work. Compared to other solutions our proposed scheme is always perceived as at least equally well or even better.

7.1 Comparison to other methods

We shortly compare our proposed method with the previously mentioned schemes with respect to the changes required to applications; dependence on additional services, hardware or software; and administrative overhead (e.g. user management).

Apart from smartcards, all other schemes require at least additional user management, in case of U2F or some OTP solutions even additional services that need to be maintained, patched, and require high availability. In addition, applications that want to utilize these additional authentication factors need to be changed significantly.

Our solution can easily be used with smartcards or with USB crypttokens that have x.509 support. In case of smartcards, card readers must be provided. USB tokens work without any additional hardware requirements except for the standard USB port. In practice, however, we realized that driver and application support for both smartcards and cryptokens is still in its infancy. But that also holds for other second factor methods.

8 Conclusion and Future Work

Sections 6 to 7 clearly show how well the system performs and how easy and flexible it is to set up and use. This applies to both service provider and users.

According to our experience, security mechanisms are best deployed, when they can be installed with just a few keystrokes, no questions asked. The instructions in this paper, while easy to follow, are already too complex for wide deployment. We therefore plan to create finished packages including Docker containers for proxies that can be integrated into existing applications or wrapped around these applications as a proxy.

Acknowledgments

This work was supported in part by the Ministry of Science, Research and the Arts (MWK) of the State of Baden-Württemberg through the funding of project bwITsec.

References

[BLP03] Biryukov, Alex; Lano, Joseph; Preneel, Bart: , Cryptanalysis of the Alleged SecurID Hash Function. Cryptology ePrint Archive, Report 2003/162, 2003. http://eprint.iacr.org/2003/162. 2

[Cr16] Cranor, Lorrie: , Time to rethink mandatory password changes. https://www.ftc.gov/news-events/blogs/techftc/2016/03/time-rethink-mandatory-password-changes, Marz 2016. Retrieved 2017-02-01. 2

[FH07] Florêncio, Dinei; Herley, Cormac: A Large-scale Study of Web Password Habits. In: Proceedings of the 16th International Conference on World Wide Web. WWW '07, ACM, New York, NY, USA, S. 657–666, 2007. 1, 2

[Ha95] Haller, Neil: The S/KEY One-Time Password System. RFC 1760, IETF, Februar 1995. https://tools.ietf.org/html/rfc1760. 2

[HF06] Herley, Cormac; Florêncio, Dinei: How to Login from an Internet Café Without Worrying about Keyloggers. In: Proceedings of SOUPS 2006. Juli 2006. Retrieved 2017-02-01. 2

[MSK05] McClure, Stuart; Scambray, Joel; Kurtz, George: Hacking Exposed: Network Security Secrets and Solutions. Osborne/McGraw-Hill, 3. Auflage, 2005. 2

[Mu91] Muffett, Alec David: , CRACK: A Sensible Unix Password Cracker. Message-ID <1991Jul15.183637.6511@aber.ac.uk> in newsgroups alt.sources, alt.security, Juli 1991. Retrieved 2017-02-01 from https://groups.google.com/d/msg/alt.sources/NAlVInmbvpk/GLdI4Dgv95MJ. 2

[NL93] Nielsen, Jakob; Landauer, Thomas K.: A mathematical model of the finding of usability problems. In: Proceedings of ACM INTERCHI'93 Conference. S. 206–213, April 1993. 7

[Sc09] Schneier, Bruce: , "Evil Maid" Attacks on Encrypted Hard Drives. https://www.schneier.com/blog/archives/2009/10/evil_maid_attac.html, Oktober 2009. Retrieved 2017-02-01. 4

[Si12] Simon, Michael; Waldvogel, Marcel; Schober, Sven; Semaan, Saher; Nussbaumer, Martin: bwIDM: Föderieren auch nicht-webbasierter Dienste auf Basis von SAML. In: 5. DFN-Forum Kommunikationstechnologien: Verteilte Systeme im Wissenschaftsbereich. S. 119–128, 2012. 2

[Si16] Simon, Michael: , 2FA mit Shibboleth mit LinOTP. Presentation at 65. DFN-Betriebstagung, Berlin, 2016. Retrieved 2017-02-01 from https://www.dfn.de/fileadmin/3Beratung/Betriebstagungen/bt65/BT65_AAI_Shib-LinOTP_Simon.pdf. 1, 2

[Th84] Thompson, Ken: Reflections on Trusting Trust. Commun. ACM, 27(8):761–763, August 1984. 2

[Th16] Thielman, Sam: Yahoo hack: 1bn accounts compromised by biggest data breach in history. The Guardian, Dezember 2016. Retrieved 2017-02-01 from https://www.theguardian.com/technology/2016/dec/14/yahoo-hack-security-of-one-billion-accounts-breached. 2

Herausforderungen des Identity Management an Hochschulen – Problemfeld Datenintegration

Manuel Haim[1]

Abstract: Die zentrale Verwaltung von Personendaten, Benutzer-Accounts und Zugriffsrechten für verschiedene IT-Systeme – kurz die *Benutzerverwaltung* – erfolgt an Hochschulen meist autonom durch zentrale Einrichtungen wie Bibliotheken und Rechenzentren. Sie wird häufig als Kernaufgabe des *Identity Management (IDM)* verstanden, ist aber solchen Anforderungen wie der regelmäßigen und verbindlichen Datenpflege nicht mehr gewachsen. Vielmehr ist eine zunehmende Integration des IDM in die Geschäftsprozesse und IT-Systeme der Hochschule erforderlich, um auf Änderungen am Personenstamm zeitnah und zuverlässig reagieren zu können.
Dieser Beitrag erläutert am Beispiel der Philipps-Universität Marburg, mit welchen besonderen Herausforderungen Hochschulen bei Einführung, Betrieb und Weiterentwicklung eines IDM-Systems konfrontiert werden. Neben einer *Anforderungsanalyse* wird das Problemfeld der *Datenintegration* herausgearbeitet und ein Weg vorgestellt, wie sich die bestehende Benutzerverwaltung in wenigen Schritten zu einem vollwertig integrierten IDM-System weiterentwickeln lässt.

Keywords: Identity Management, IDM, Datenintegration, Provisioning.

1 Motivation

Stellen wir uns eine beliebige deutsche Hochschule vor: *Welche konkreten Personen* gehören jetzt im Augenblick z.B. zur Philipps-Universität Marburg? Und *in welcher Funktion* sind diese Personen tätig? – Was zunächst nach einer einfachen Zählaufgabe anmutet, erweist sich bei näherer Betrachtung als ein komplexes Problem: Hochschulen sind zumeist dezentral organisiert, Fachbereiche und Einrichtungen arbeiten weitestgehend autonom, ihre Angehörigen sind in den unterschiedlichsten Untereinheiten und Verhältnissen tätig. Das Studierendensekretariat pflegt zwar die Studierendendaten; die Personalabteilung kennt zumindest die Arbeitsverträge; viele weitere Personengruppen und Details müssen aber bei Bedarf mühsam bei zahlreichen Stellen erfragt werden (z.B. Doktoranden, Lehraufträge, konkrete Arbeitsgebiete, Funktionen, Ehrenämter, Gasttätigkeiten usw.).

Bislang wurde dieser Umstand von manchen Hochschulen selten als Problem empfunden, denn es gab wenig Grund und Anlass, ein tagesaktuelles Gesamtverzeichnis aller Personen zu führen. Die zentralen Einrichtungen wie Bibliotheken und Rechenzentren, die Dienste für alle Hochschulangehörigen anbieten, behelfen sich hingegen oftmals mit ihren eigenen Benutzerdatenbanken: Personen werden auf schriftlichen Antrag aufgenommen (z.B. gegen Vorlage von Verträgen oder Bestätigung von Vorgesetzten) und erhalten so einen persönlichen, in der Regel befristeten Zugang.

[1] Philipps-Universität Marburg, Hochschulrechenzentrum, Hans-Meerwein-Straße 6, 35032 Marburg, haim@hrz.uni-marburg.de

Die zunehmende Digitalisierung und der Zuwachs an bereichsübergreifenden Prozessen (Stichwort: Prozessorientierte Hochschule) führen jedoch zu neuen Anforderungen:

- Mittels *Shibboleth* [Sh17] soll die organisationsübergreifende Authentifizierung und Autorisierung gegenüber Online-Literatur und Webanwendungen anderer Hochschulen ermöglicht werden. Hierfür sind innerhalb der *DFN-AAI-Föderation* [DF15] sinnvollerweise nur persönliche Accounts zugelassen, deren Inhaber zweifelsfrei identifiziert wurden, ihre Zugangsdaten auf sicherem Weg erhalten haben und deren Daten bei Änderungen binnen zwei Wochen aktualisiert oder gesperrt werden.
- Im *integrierten Campus-Management (iCM)* sollen sämtliche Prozesse des Studienmanagements vereint werden: „[...] von der Bewerbung über modulbezogene Lehrveranstaltungen und Prüfungen bis hin zur Erstellung von Leistungsübersichten und Zeugnissen“ [Ph14]. Dazu müssen alle an Studium und Lehre beteiligten Personen bekannt sein – das sind neben den Studierenden und dem Personal insbesondere auch die wechselnden Lehrbeauftragten, Doktoranden, Ehrenamtliche und Gäste.
- Ein *Forschungs-Informations-System (FIS)* [Ph16, He17] soll die Forschungsprozesse unterstützen: Von der Planung und Dokumentation von Forschungsprojekten über die Erfassung von Publikationen bis hin zur Berichterstattung nach dem Landeshochschulgesetz. Hierfür müssen alle an der Forschung beteiligten Personen mitsamt ihrer Organisationszugehörigkeit und Funktion bekannt sein und zur Nutzung des FIS-Systems verpflichtet werden.
- Das neue *Hochschulstatistikgesetz (HStatG)* fordert eine regelmäßige Erhebung aller Doktorandinnen und Doktoranden der Hochschule ab dem Berichtsjahr 2017.

Diesen Anforderungen ist eine antragsbasierte Benutzerverwaltung nicht mehr gewachsen, da sie in puncto Qualität und Vollständigkeit der Benutzerdaten entsprechende Mängel aufweist. Vielmehr ist eine Integration der Benutzerverwaltung als *Identity-Management-System (IDM-System)* in die Geschäftsprozesse und IT-Systeme der Hochschule erforderlich, um Änderungen an Personendaten unmittelbar dort abzugreifen, wo sie entstehen.

2 Anforderungen an IDM-Systeme

Der Begriff *Identity Management (IDM)* wird in der Literatur sowie von diversen Software-Anbietern auf vielfältige Weise und in scheinbar unterschiedlichem Umfang interpretiert. In diesem Abschnitt sollen daher zunächst diejenigen Anforderungen genannt und erläutert werden, die bislang im Rahmen der Weiterentwicklung der Benutzerverwaltung an der Philipps-Universität identifiziert wurden und für IDM-Systeme an Hochschulen typisch erscheinen (vgl. [Ha16b, S. 17]). Im weiteren Verlauf des Beitrags können so die Defizite der Benutzerverwaltung konkret benannt werden. Die Anforderungen lassen sich in *funktionale* und *nicht-funktionale Anforderungen* unterteilen.

Die **funktionalen Anforderungen** decken sich überraschend gut mit einer allgemeinen Klassifizierung der Funktionen von IDM-Systemen, die Steffen Hofmann in [Ho07, S. 8–

14] aufgestellt hat. Zum besseren Vergleich werden Hofmanns Begriffe nachfolgend in eckigen Klammern *[]* beigefügt:

2.1 Zentrales Verzeichnis *[Informationsspeicher]*

Für Zwecke wie z.B. die öffentlichen Webseiten, Telefonverzeichnisse oder Webanwendungen wird eine *gemeinsame, verlässliche Datenbasis* benötigt, z.B. eine Datenbank oder ein Verzeichnisdienst. Hier überschneidet sich das vorwiegend auf Personen bezogene Identity Management ggf. mit dem auf beliebige Objekte bezogene Stammdaten-Management (Master Data Management, MDM) und es können ggf. ähnliche Technologien oder dieselbe Datenbank verwendet werden (vgl. auch Unterabschnitt 2.3).

2.2 Datenintegration *[Datenintegration]*

Die Integration des IDM-Systems in bestehende Geschäftsprozesse und IT-Systeme, die Personendaten liefern, ist an Hochschulen mit besonderen Schwierigkeiten verbunden. Es bedarf der nachhaltigen Unterstützung der Hochschulleitung, um die Mitwirkung aller beteiligten Stellen zu erreichen. Konkret gibt es hierbei die folgenden Herausforderungen:

2.2.1 Umgang mit zahlreichen Datenquellen

Unterschiedliche *Personengruppen* bzw. *Funktionen* werden von ganz unterschiedlichen Stellen erfasst bzw. zugeteilt. Tabelle 1 bietet einen groben (unvollständigen) Überblick.

Personengruppe	Verantwortlich	Aktion	Datenbank
Studierende	Studierendensekretariat	Immatrikulation	HISinOne
Professoren/-innen	Personalabteilung	Einstllg. / Verbeamtg.	Hessen-SAP
Mitarbeiter/-innen	Personalabteilung	Einstllg. / Verbeamtg.	Hessen-SAP
Landesbedienstetes Personal des Klinikums	Personalabt. Klinikum	Einstllg. / Verbeamtg.	Klinikums-SAP
apl. Professoren/-innen	Senat	Titelverleihung	unbekannt
Honorarprofessoren/-innen	Senat	Titelverleihung	unbekannt
Privatdozenten/-innen	jew. Dekanat	Titelverleihung	unbekannt
Lehrbeauftragte	jew. Dekanat	Erteilung Lehrauftrag	unbekannt
Doktoranden/-innen	jew. Dekanat	Annahme	unbekannt
Dekane/-innen	jew. Fachbereichsrat	Wahl	unbekannt
Fachbereichsbeauftragte	jew. Dekanat	Benennung	unbekannt

Tab. 1: Übersicht einiger Personengruppen und Datenquellen

Anmerkungen: Das Uni-Klinikum Marburg wurde privatisiert, ein Teil des Klinikumspersonals wird aber weiterhin vom Land Hessen bezahlt und wirkt am Fachbereich Medizin der Philipps-Universität mit. Daneben gibt es zahlreiche weitere Sondergruppen und Kooperationsvereinbarungen, die an dieser Stelle jedoch nicht näher erläutert werden sollen.

2.2.2 Umgang mit unterschiedlicher Datenqualität

Je nach Zielsetzung und Sorgfalt der datenführenden Stellen können die dort vorgehaltenen Informationen für ein zentrales Identity Management mehr oder weniger brauchbar sein.

Generell sollte die *Zweckmäßigkeit* des jeweils datenführenden Systems hinterfragt werden. Liegen die Daten z.B. auf Papier oder elektronisch vor? Werden die Daten regelmäßig aktualisiert? Ist die Bedeutung der Daten eindeutig? Entsprechen z.B. die zugeordneten Organisationseinheiten dem tatsächlichen Einsatzbereich einer Person? Oder deuten sie nur auf die verantwortlichen Kostenstellen hin, aus denen die Planstelle finanziert wird?

Darüber hinaus stellt sich die Frage nach der *Vertrauenswürdigkeit* der Personendaten. Hat eine Person sich z.B. per Webformular selbst registriert? Wurde die Identität der Person überprüft, z.B. anhand von Ausweis oder Zeugnissen? Wurde der Ausweis persönlich vorgelegt? Wurden weitere Identifizierungsmerkmale erfasst wie z.B. das Geschlecht oder das Geburtsdatum? Können reale Personen den Daten zweifelsfrei zugeordnet werden?

Wie steht es konkret um die *Erfassung des Namens*? Ist der Name vollständig, d.h. sind alle Vor- und Nachnamen enthalten? Oder nur der Rufname? Ist der Name evtl. verkürzt, z.B. „Bernd“ statt „Bernhard“? Wie wahrscheinlich sind Tippfehler? Stimmt die Schreibweise mit den amtlichen Ausweisdokumenten überein? Und ist der Name noch aktuell?

2.2.3 Umgang mit Dubletten

Dieselbe Person kann ggf. *mehrfach* erfasst worden sein – z.B. in unterschiedlichen Quellsystemen, in leicht abweichenden Schreibweisen, aufgrund einer zwischenzeitlichen Namensänderung oder ggf. völlig abweichenden Namen bei doppelter Staatsbürgerschaft.

Mögliche Dubletten sollten regelmäßig *aufgespürt* und *zusammengeführt* werden (ggf. in Rücksprache mit den betreffenden Personen, sofern sich die Übereinstimmung nicht ohnehin z.B. aus Name, Geburtsdatum und Fachgebiet ergibt). Auf die Zusammenführung von unterschiedlich vertrauenswürdigen Datensätzen (Stichwort: Selbstregistrierung) sollte ggf. verzichtet werden, nicht zuletzt um Identitätsdiebstahl zu verhindern.

2.3 Datenabgleich *[Provisioning]*

Sowohl im IDM als auch im Stammdaten-Management (Master Data Management, MDM) spielt der *regelmäßige Datenabgleich* (Synchronization bzw. Provisioning / Deprovisioning) zwischen IT-Systemen eine zentrale Rolle. Hierfür gibt es am Markt verschiedenste Softwarelösungen – teils eigenständig, teils in IDM- oder MDM-Software integriert – mit recht unterschiedlichem Funktionsumfang. Aber auch Eigenentwicklungen sind denkbar.

Das Grundproblem ist jedoch von der Software unabhängig: Ein *Regelwerk* muss definiert werden, das festlegt, nach welchen Regeln Daten miteinander abgeglichen werden

sollen. Schließlich müssen *Konnektoren* programmiert werden. Beides ist angesichts der vielfältigen Datenquellen an Hochschulen eine nicht-triviale Entwicklungsaufgabe.

2.4 Authentisierung / Authentifizierung *[Authentifizierung, Passwort Management]*

Um sich gegenüber IT-Systemen *ausweisen (authentisieren)* zu können, benötigt jede Person einen persönlichen Zugang bzw. Account. Die wohl gängigste Authentisierungsmethode ist die Eingabe von Benutzername und Passwort. Diese *Zugangsdaten* müssen der Person auf sicherem Weg übermittelt werden, z.B. persönlich am Helpdesk oder per Hauspost an die Dienstanschrift, damit sie nicht in die Hände Dritter gelangen.

Aus Gründen der Benutzerfreundlichkeit (und zur Vermeidung von Passwort-Spickzetteln) sollten möglichst alle IT-Systeme mit *denselben Zugangsdaten* zugänglich sein, idealerweise bereits nach einmaliger Anmeldung (Single Sign On, z.B. per Kerberos oder Shibboleth).

Eine Ausnahme stellen *kritische Systeme und Aktionen* dar, die man gesondert gegen Passwortdiebstahl absichern sollte, wie z.B. die Prüfungsanmeldung oder Finanzverwaltung. Hierfür ist die Abfrage eines *zweiten Faktors* sinnvoll, z.B. einer Transaktionsnummer (TAN) aus einer TAN-Liste, oder eines per Hardware-Token oder Smartphone-App generierten Einmalpassworts (One-Time Password, OTP).

Für Dienste wie WLAN, die aus praktischen Gründen häufig mit einer Speicherung des Passworts verbunden sind, sind zusätzliche *anwendungs- oder gerätebezogene Passwörter* wünschenswert, die nur zu dem einen Zweck verwendet werden können, z.B. das Smartphone mit dem Hochschul-WLAN zu verbinden.

2.5 Autorisierung und Access Management *[Autorisierung]*

Abhängig von ihrem Verhältnis zur Universität steht einer Person das Nutzungsrecht für eine definierte Teilmenge von IT-Systemen und deren Inhalten zu. *Rollen und Rechte* müssen dazu digital erteilt, gepflegt und entzogen werden – idealerweise automatisiert. Sie sollten dazu an *Bedingungen* geknüpft sein wie z.B. an die Personengruppe, an die Abteilung, an ein Ablaufdatum oder eine regelmäßig notwendige formale Verlängerung.

Ggf. sind *Übergangsfristen* nötig, damit z.B. Lehrbeauftragte noch Seminararbeiten korrigieren oder ehemalige Studierende ihre Prüfungsergebnisse elektronisch abrufen können.

Die *Anhäufung von Rechten* z.B. aufgrund mehrfacher Abteilungswechsel sollte vermieden werden (Stichwort: Praktikanten haben die meisten Rechte).

2.6 Webportal *[User Self-Service, (De-)Zentrale Administration, Workflow Mgmt.]*

Zur Einsicht und Verwaltung der Daten, Passwörter, E-Mail-Adressen usw. sollten entsprechende *Webformulare* oder eine eigene *Webanwendung* bereitgestellt werden. Ein *Self Ser-*

vice verringert nebenbei den Arbeitsaufwand für Helpdesk und Administratoren/-innen, da die Benutzer/innen sich selbst helfen können. Sofern Daten öffentlich sichtbar sind, haben letztere erfahrungsgemäß ein großes Interesse daran, sie regelmäßig zu aktualisieren.

2.7 Audit Logging *[Auditing]*

Änderungen an Account, Rollen und Rechten müssen (ggf. rechtssicher) *protokolliert* werden, um im Zweifelsfall nachvollziehen zu können, wer wann welche Änderungen vorgenommen hat. Sofern sich die Änderungen längerfristig auswirken (z.B. ein längerfristiges Ablaufdatum), sollten die Protokolleinträge entsprechend lang aufgehoben werden.

Nicht-funktionale Anforderungen:

2.8 Datenschutz

Die Verarbeitung personenbezogener Daten erfordert eine Einhaltung der geltenden Datenschutzgesetze. In Deutschland muss die Verarbeitung z.B. gemäß Bundes- (BDSG) und jeweiligem Landesdatenschutzgesetz (LDSG) in einem *Verfahrensverzeichnis* dokumentiert und dieses dem Datenschutzbeauftragten der Universität vorab zur Prüfung vorgelegt werden; die personenbezogenen Daten dürfen anschließend nur zu den festgelegten Zwecken verarbeitet und weiterverwendet werden. Sofern *Personaldaten* verarbeitet werden, ist außerdem der Personalrat einzubeziehen.

Aufbewahrungs- und Löschfristen sind zu definieren und einzuhalten. Die Aufbewahrungsfristen können sich je nach IT-System unterscheiden: Sie können rein technisch bedingt sein (z.B. durch Vorhalten einer allgemeinen Datensicherung/Backup für 3 Monate) oder sich auf Rechtsvorschriften stützen (z.B. gesetzliche Pflicht zur revisionssicheren Archivierung von Daten der Finanzbuchhaltung für 10 Jahre). Die Löschfristen sollten aus Gründen der Datensparsamkeit möglichst kurz gehalten werden.

Leicht übersehen wird der Umgang mit vormals vergebenen *Benutzernamen und E-Mail-Adressen*. Hierbei handelt es sich einerseits um personenbezogene Daten, die schnellstmöglich gelöscht werden sollten. Andererseits kann die Wiedervergabe an Dritte zu unerwünschten Nebeneffekten führen, wenn z.B. in Drittsystemen noch gleichnamige Benutzerkonten existieren oder E-Mail-Adressen schon allgemein bekannt sind. Sofern man solche Bezeichner von der Wiedervergabe an Dritte ausschließen möchte, müssen sie längerfristig gespeichert werden. Ferner bleibt die Frage, ob denn zumindest die Wiedervergabe an die ursprünglichen Nutzer/innen erwünscht ist. Falls ja, ist es nötig, zusätzlich zum Bezeichner auch persönliche Daten vorzuhalten, die eine spätere Identifikation ermöglichen.

Es ist außerdem festzulegen, *wer wann welche Daten bearbeiten und einsehen darf*. Soll z.B. die Personalabteilung Zugriff auf alle Identitäten im IDM-System erhalten, um bestehende Studierende direkt ins Personalsystem übernehmen zu können? Oder lieber auf ihr Personalsystem und das dort verzeichnete Personal beschränkt bleiben?

2.9 Randbedingungen / Kosten

Sofern die Anschaffung einer IDM-Software in Betracht gezogen wird, sind die *laufenden Kosten für Lizenzen, Anpassungen und Wartung* zu berücksichtigen. Die Anzahl der Personen an einer Hochschule ist vergleichsweise groß; sie kann aufgrund steigender Studierendenzahlen, neuer Angebote und Kooperationen jederzeit sprunghaft ansteigen. Lizenzmodelle, die sich linear an der Anzahl der zu verwaltenden Identitäten orientieren, sind für Hochschulen daher tendenziell unattraktiv; Lizenzverlängerungen, die von Jahr zu Jahr unvorhersehbar teurer werden, ebenso.

Darüber hinaus ist zu klären, *in welchem Zeitrahmen* Anpassungen möglich sind (wie z.B. Veränderungen von Workflows oder Entwicklung neuer Konnektoren) bzw. ob diese auch *eigenständig ohne Verletzung der Wartungsverträge* durchgeführt werden dürfen.

3 Bisherige Benutzerverwaltung an der Philipps-Universität

In diesem Abschnitt wird die bestehende Benutzerverwaltung der Philipps-Universität kurz vorgestellt. Hierbei werden insbesondere *Mängel bei der Datenintegration* deutlich.

Bereits seit 1995/1996 stellt das Hochschulrechenzentrum (HRZ) der Philipps-Universität allen interessierten Studierenden sowie Professoren/-innen und Mitarbeitern/-innen auf Antrag einen zentralen E-Mail-Account zur Verfügung. Dieser Account wurde schon bald für die Authentisierung gegenüber weiteren Diensten genutzt. Die Selbstverwaltung erfolgt seither über Webformulare (CGI-Skripte) auf den E-Mail-Servern.

Seit 2001 dient ein *OpenLDAP-System* als zentraler Verzeichnisdienst. Historisch bedingt wird zwischen Students- und Staff-Accounts sowie zwischen Students- und Staff-Personeneinträgen unterschieden, die in getrennten Zweigen gespeichert sind. Weitere Zweige umfassen u.a. Gebäude, Telefone, Netzwerk-Hosts sowie Organisationseinheiten (letztere hierarchisch). Die einzelnen Objekte werden über Links miteinander verknüpft.

Bei den *Studierenden und anderen Veranstaltungsteilnehmern (Students)* gibt es stets eine 1:1-Beziehung zwischen dem Account, dem Personeneintrag und dem Primärschlüssel im jeweiligen Quellsystem (z.B. der Matrikelnummer beim Studierendensekretariat), so dass einer automatischen Datenübernahme und Accountgenerierung nichts im Wege steht. Für die Datenübernahme gibt es entsprechende Perl-Skripte, die nach dem Erhalt eines Datenexports bislang noch manuell angestoßen werden.

Bei den *Mitarbeitern/-innen und anderweitig Tätigen (Staff)* ist es etwas komplizierter. Die Personeneinträge sind ggf. zugleich als *Visitenkarten* im Organigramm auf den Webseiten der Universität sichtbar – d.h. eine natürliche Person, die in mehreren Bereichen tätig ist, kann über mehrere solcher Einträge verfügen. Zusammengehörende Personeneinträge tragen nach Möglichkeit dieselbe Personen-ID, die sich aus Nachnamen, Vornamen und einer laufenden Nummer zusammensetzt. Einen Hauptdatensatz gibt es nicht, d.h. weitere Objekte wie Accounts sind jeweils nur mit einem der vorhandenen Personeneinträge verknüpft. Die Pflege der Personeneinträge und anderer LDAP-Objekte erfolgt manuell über

eine selbst entwickelte Webanwendung (myLDAPadmin), Anpassungen vorhandener Einträge sind so auch dezentral über die Fachbereiche möglich. So wird z.B. die internetweite Sichtbarkeit von Personeneinträgen durch eine schriftliche Einverständniserklärung geregelt, die durch Personendatenbeauftragte am Fachbereich gesammelt und bearbeitet wird.

Im Rahmen der Benutzerverwaltung bietet ein menügeführtes Perl-Skript den HRZ-Administratoren/-innen die Möglichkeit, *Accounts* zu verwalten oder *Mitarbeiterdaten* aus anderen Quellen zu übernehmen. Bei der Bearbeitung von Account-Anträgen, die in Form von Papierformularen eingereicht wurden, wird zunächst der Personeneintrag manuell per myLDAPadmin angelegt oder angepasst, dann der Account per Perl-Skript erzeugt, wobei Teilaufträge für weitere Systeme (in Form von Shell-Kommandos) in einer Queue abgelegt werden. Dahingegen werden bei der automatisierten Übernahme von Mitarbeiterdaten (z.B. bei Personalmeldungen) zunächst für jeden Quell-Personendatensatz die möglicherweise passenden LDAP-Personeneinträge aufgelistet; die tatsächliche Zuordnung oder Neuerstellung von Personeneinträgen erfolgt jedoch erst nach manueller Bestätigung. Veränderte Daten werden ggf. übernommen und ein ggf. vorhandener Account verlängert.

Das Problem hierbei: An den Personeneinträgen und Accounts der Mitarbeiter/innen ist meist nicht mehr ablesbar, welche Daten und Fristen sich aus welcher Quelle ergeben. Bei Unklarheiten müssen die archivierten Account-Anträge und Log-Dateien betrachtet werden. Dies macht eine automatisierte und zuverlässige Pflege der Rollen und Rechte unmöglich. Außerdem häufen Personen ggf. mehrere Personeneinträge an, die von den Fachbereichen oder vom HRZ manuell gepflegt oder bereinigt werden müssen.

4 Aktuelle Weiterentwicklung an der Philipps-Universität

Um die Benutzerverwaltung zu einem universitätsweiten IDM-System weiterzuentwickeln, müssen insbesondere die *Mängel bei der Datenintegration* beseitigt und ein *Regelwerk für den Datenabgleich* konzipiert werden. Die hierfür relevanten Nahziele (und bisherige Fortschritte) werden in den nachfolgenden vier Punkten kurz erörtert.

4.1 Durchgängige Erfassung der Personendaten aller an der Universität tätigen Personengruppen (dezentral durch geeignete Stellen, in geeigneten Systemen)

Anhand der laufend eingereichten Account-Anträge auf Papierformularen und bestehenden Datenaustausche wurde in den letzten Jahren zunächst eine *detaillierte Aufstellung aller zu erfassenden Personengruppen* erstellt. Hierbei war vor allem die Benennung derjenigen Gruppen interessant, die bislang nicht systematisch an das HRZ gemeldet werden, aber trotzdem einen Account oder Personeneintrag erhalten. Insgesamt konnten so knapp 40 Personengruppen ausfindig gemacht werden. Diese Größenordnung ist für eine Hochschule nicht ungewöhnlich, wie die Erfahrungen anderer Hochschulen zeigen (z.B. FAU Erlangen-Nürnberg [Fr16]).

Es folgten (teils motiviert durch weitere Aufgabengebiete wie Shibboleth oder Campus-Management) *Gespräche mit den Betreibern der bestehenden datenführenden Systeme*, um mehr über die Datenqualität, Pflegeprozesse und Bedeutung der Quelldaten zu erfahren. Im Vordergrund standen hierbei die Studierenden und das Personal, da diese den Großteil der Hochschulangehörigen ausmachen und ihnen im Universitätsalltag die meisten Rechte zustehen.

Noch offen ist die *Ausgestaltung der elektronischen Erfassungs- und Meldeverfahren* für viele übrige Gruppen, die bislang nicht oder nur unzureichend elektronisch erfasst werden. Hierfür müssen die verantwortlichen Stellen eingebunden und neue Prozesse geschaffen werden. Es ist denkbar, für die Datenpflege vorhandene Systeme wie HISinOne zu nutzen (mit zusätzlichen Personenklassen) oder eigene Datenbanken bereitzustellen.

4.2 Konsolidierung der Personendaten und zeitlich beschränkten Teil-Identitäten in einer zentralen Datenbank des Hochschulrechenzentrums

Bislang gingen beim Datenimport nach LDAP Informationen verloren, da Daten aus unterschiedlichen Quellen direkt zu einem LDAP-Personeneintrag verschmolzen wurden. Um die Informationen über die *Teil-Identitäten* nicht zu verlieren, wird daher künftig für jede Datenquelle ein separater Verzeichniszweig im LDAP bereitgestellt, in welchem die Quelldatensätze als (Schatten-)Kopie abgelegt werden. In einem weiteren Zweig wird dann separat die *Identität* gepflegt, welche Links zu beliebig vielen Teil-Identitäten erhalten kann. Ziel ist, für jede natürliche Person möglichst nur *eine* Identität vorzuhalten. Die bisherigen Personeneinträge sollen dahingegen wie unterschiedliche Visitenkarten behandelt und ebenfalls mit der Identität verknüpft werden.

Der Datenabgleich erfolgt folglich in zwei Schritten. In einem ersten Schritt werden die Quelldatenbanken 1:1 mit den Schattenkopien abgeglichen – und hierbei die Daten bereits in ein einheitliches Format gebracht. Zur Unterstützung des Datenabgleichs erhalten die Schattenkopien jeweils einen Zeitstempel, der Aufschluss über den Zeitpunkt der Erstellung, letzten Änderung sowie Löschung gibt. Dies ermöglicht später einen zeitnahen Delta-Abgleich mit weiteren LDAP-Zweigen (im Gegensatz zu einem zeitaufwendigen Komplettabgleich).

In einem zweiten Schritt müssen die Schattenkopien mit den Identitäten abgeglichen werden. Die Zuordnung neuer Schattenkopien zu Identitäten soll für die meisten Personengruppen vorerst weiterhin manuell erfolgen, um unnötige Dopplungen oder Falschzuordnungen zu vermeiden. Der automatische Abgleich wird dann nur für die bereits verlinkten Identitäten ausgeführt, wobei die Daten in den Schattenkopien (sofern sie voneinander abweichen) mit unterschiedlicher Priorität bewertet werden können. In Rücksprache mit den betreffenden Personen sind Overwrites möglich, z.B. falls nur der Rufname oder (z.B. bei Ligaturen) eine von den amtlichen Dokumenten abweichende Schreibweise erwünscht ist.

Als *Werkzeug für den Datenabgleich* wurde eine eigene, nur wenige hundert Zeilen Code umfassende Python-Bibliothek namens *CALYPSO* entwickelt [Ha16a]. Diese bietet neben einheitlichen Datenkonnektoren zwei wesentliche, leicht zu konfigurierende Algorithmen:

- Der *Sync-Algorithmus* vergleicht die Datensätze in Quell- und Zieldatenbank 1:1 anhand eines frei wählbaren Attribut-Mappings und reagiert auf vordefinierte Situationen (z.B. absent, found, ambiguous, source_missing) mit wählbaren, vordefinierten Aktionen (z.B. create, update, ignore, delete). Dieses Vorgehen ist an gängige IDM-Software angelehnt (z.B. OpenIDM [Fo16], midPoint [Ev14]).
- Der *Merge-Algorithmus* sucht alle Quell-Datensätze, die zu einem spezifischen Ziel-Datensatz passen, und führt deren Daten zu einem temporären Datensatz zusammen, um diesen anschließend mit dem Ziel-Datensatz abzugleichen. Auf diese Weise lässt sich z.B. die Summe der aktuellen Rollen und Berechtigungen für eine Identität oder einen Account ableiten. Für die Zusammenführung der gefundenen Datensätze kann eine eigene Funktion angegeben werden, z.B. um die Quelldaten mit unterschiedlicher Priorität zu behandeln und so der Namensschreibweise der Personalabteilung Vorrang vor anderen Schreibweisen zu gewähren.

Zu guter Letzt wollen die gesammelten Daten sinnvoll verwaltet werden. Für die weitere Administration und Selbstverwaltung von Personendaten und Accounts ist daher die *Entwicklung einer eigenen Webanwendung* auf Basis von AngularJS und Bootstrap (browserseitig) sowie Flask und uWSGI (serverseitig) geplant; die Kommunikation zwischen Browser und Server wird über REST erfolgen.

4.3 Automatisierte Ableitung von Accountlaufzeit, Rollen und Rechten

Um den Benutzern/-innen die ihnen unmittelbar zustehenden digitalen Berechtigungen zuweisen zu können, waren zunächst die *rechtlichen Zugangsvoraussetzungen für die einzelnen Dienste* zu klären. Hierzu erfolgten Gespräche z.B. mit der HRZ-Netzwerkabteilung bzgl. eduroam-WLAN und DFN-Internet, mit der Universitätsbibliothek bzgl. lizensierter Online-Literatur, oder mit der DFN-AAI-Föderation bzgl. Shibboleth.

Mit dem gesammelten Wissen konnte ein Datenquellen-Rollen-Rechte-Modell entwickelt werden; ggf. sind noch Übergangsfristen beim Entzug von Rechten zu klären. Die automatisierte Pflege der Rollen und Rechte wird durch CALYPSO geschehen, wie im vorigen Abschnitt angedeutet.

4.4 Automatische (De-)Provisionierung aller angeschlossenen Systeme

Schließlich sollen die zentral geführten Informationen *zeitnah* mit allen Anwendungssystemen *abgeglichen* werden. So können Benutzerprofile z.B. schon verfügbar gemacht werden, bevor ein Nutzer eine Anwendung erstmalig nutzt – dies vereinfacht z.B. die Gruppenzuordnung in Drittsystemen wie der Lernplattform ILIAS. Aber auch die Löschung von Daten in Drittsystemen soll automatisiert geschehen. Schattenkopien bieten auch hier eine Möglichkeit, im IDM abzubilden, welche Benutzerprofile in Drittsystemen bestehen, erstellt oder gelöscht werden sollen.

5 Erläuterungen zur Vorgehensweise

Die Einführung eines IDM-Systems an der Philipps-Universität hatte lange Zeit niedrige Priorität. Die Problemanalyse und Entwicklung erfolgte daher vorwiegend agil im Rahmen der Arbeitszeit, die für die Pflege und Weiterentwicklung der Benutzerverwaltung im HRZ dauerhaft vorgesehenen ist (60% einer unbefristeten Vollzeitstelle).

Im Jahr 2012 wurde begonnen, die Anforderungen und Mängel der Benutzerverwaltung neu zu dokumentieren sowie ab dem Jahr 2013 auch Fremdsoftware zu evaluieren, um ein universitätsweites IDM aufzubauen. Insbesondere wurde Open-Source-Software getestet, um eine möglichst kostenneutrale Lösung zu entwickeln.

Mit *OpenIDM* konnten dank mitgelieferter Konnektoren und nachvollziehbarer JSON-Konfigurationsdateien schnell erste Datenabgleiche implementiert werden. IDM-Spezifika wie ein sinnvolles Datenschema, eine Dublettensuche oder gar eine Rechteverwaltung suchte man im Jahr 2014 aber vergebens. Der produktive Einsatz (inkl. Updates und Maintenance-Releases) war außerdem nicht mehr Bestandteil der Open-Source-Lizenz, sondern an einen Supportvertrag gekoppelt: Auf die Philipps-Universität wären hierbei jährliche Ausgaben im Umfang von mehreren zehntausend Euro zugekommen (Stand 06/2014). Angesichts der zweifelhaften Mehrwerte wurde auf die Anschaffung verzichtet.

Gegen den Einsatz der aus OpenIDM hervorgegangenen, frei nutzbaren Software *midPoint* sprach im Jahr 2014 die vergleichsweise höhere Lernkurve sowie die nur mühsam lesbare XML-Konfiguration. Nebenbei war ein Datenabgleich ohne Zwischenspeicherung von Master-Objekten nicht vorgesehen. Die Dokumentation sowie die Roadmap klangen zwar vielversprechend, viele Funktionen ließen aber noch auf sich warten.

Ein gemeinsames Manko bei diesen und auch weiteren betrachteten IDM-Lösungen war die mäßige Unterstützung für Portalanpassungen, eigene Webformulare und Workflows. Die guten Erfahrungen von Kollegen mit AngularJS befürworteten eine Eigenentwicklung.

Da im Jahr 2014 bereits der Entwurf für ein neues Datenmodell bestand, beschränkte sich die weitere Suche vorerst auf eine allgemeine Software zum Datenabgleich, die auch in weiteren Bereichen des HRZ eingesetzt werden könnte. Für komplexe Datentransformationen hat sich hier die kostenlose Community Edition von *Pentaho Data Integration (PDI, ehemals Kettle)* nun schon mehrfach bewährt. Die Datenobjekte im IDM sind hingegen vergleichsweise einheitlich, die Algorithmen zum Datenabgleich ebenso – sie müssen lediglich mit dem z.B. aus OpenIDM bekannten Vokabular parametrisiert werden. Eine schlanke Eigenimplementierung in Form einer *Common Algorithm Library for the Provisioning and Synchronization of Objects (CALYPSO)* [Ha16a] lag somit nahe.

6 Fazit

In diesem Beitrag wurde gezeigt, dass sich die funktionalen Anforderungen der Philipps-Universität an ein IDM-System nicht wesentlich vom marktüblichen Funktionsumfang unterscheiden. Sie werden an der Philipps-Universität zum Großteil von bereits implementierten Lösungen abgedeckt, so dass der Erwerb zusätzlicher Software vorerst entfällt.

Auch der Zeit- und Personalaufwand für die sukzessive Weiterentwicklung ist, verglichen mit der Einführung fertiger Lösungen, überschaubar (vgl. Uni Konstanz: Austausch des *Sun Identity Manager* durch *OpenIDM* von ca. 09/2012 bis 02/2015 [ZK17]). Eine besondere Herausforderung stellen in beiden Fällen die *sehr zahlreichen Datenquellen und Verantwortlichkeiten* dar, die zunächst identifiziert bzw. geklärt werden müssen. Dies ist ein langwieriger, nicht-technischer Prozess mit vielen Beteiligten, der unabhängig von der gewählten Software nötig ist. Zudem muss ein *Webportal* meist selbst entwickelt werden.

Der eigentliche *Datenabgleich* erweist sich hingegen als ein geringeres Problem. Hierfür wurden ein Datenmodell und Algorithmen skizziert, wie sie für namhafte IDM-Software üblich sind. Anhand der Beispiel-Implementierung CALYPSO wird deutlich, dass sich die Abgleichmechanismen bereits mit geringem Aufwand nachbilden lassen.

Literaturverzeichnis

[DF15] DFN-Verein: Klassen der Verlässlichkeit in der DFN-AAI, https://www.aai.dfn.de/der-dienst/verlaesslichkeitsklassen/, Stand: 31.03.2015.

[Ev14] Evolveum: midPoint – Synchronization Situations, https://wiki.evolveum.com/display/midPoint/Synchronization+Situations, Stand: 07.01.2014.

[Fo16] ForgeRock: OpenIDM 4.5 Integrator's Guide – Synchronization Situations and Actions, https://backstage.forgerock.com/docs/openidm/4.5/integrators-guide/chap-synchronization#handling-sync, Stand: 19.12.2016.

[Fr16] Friedrich-Alexander-Universität Erlangen-Nürnberg: IdM-Portal, Kundengruppen/-typen, https://www.idm.uni-erlangen.de/aim/docs/affiliations, Stand: 19.12.2016.

[Ha16a] Haim, Manuel: CALYPSO – ein Python-Skript zum generischen Datenabgleich, https://www.zki.de/fileadmin/zki/Arbeitskreise/VD/Protokolle/2016-09-12/calypso-unimr.pdf, Stand: 12.09.2016.

[Ha16b] Haim, Manuel: Von der HRZ-Benutzerverwaltung zum hochschulweiten Identity Management, https://www.zki.de/fileadmin/zki/Arbeitskreise/VD/Protokolle/2016-03-14/idm-haim.pdf, Stand: 14.03.2016.

[He17] HeFIS-Verbund: Ziele und erwartete Mehrwerte, http://www.hefis-verbund.de/fis/mehrwerte, Stand: 03.01.2017.

[Ho07] Hofmann, Steffen: Architektur eines Identitätsmanagementsystems an einer Hochschule. Diplomarbeit, FernUniversität Hagen, Juni 2007, https://www.zedat.fu-berlin.de/pub/ZEDAT/FUDIS/Home/Architektur_eines_Identitaetsmanagementsystems_an_einer_Hochschule.pdf, Stand: 19.12.2016.

[Ph14] Philipps-Universität Marburg: Integriertes Campus-Management, http://www.uni-marburg.de/integriertes-campus-management/projekt, Stand: 09.06.2014.

[Ph16] Philipps-Universität Marburg: Forschungs-Informations-System, http://www.uni-marburg.de/administration/verwaltung/dez1/fis, Stand: 08.06.2016.

[Sh17] Shibboleth Consortium: What's Shibboleth?, http://shibboleth.net/about/, Stand: 03.01.2017.

[ZK17] ZKI-Arbeitskreis Verzeichnisdienste: Protokolle der halbjährlichen Arbeitskreistreffen, https://www.zki.de/arbeitskreise/verzeichnisdienste/protokolle/, Stand: 12.03.2017.

Prozessorientiertes IT-Service Management

Einführung eines zertifizierten Qualitätsmanagementsystems im IT-ServiceDesk des IT Centers der RWTH Aachen University

Martin Pieters[1], Ingo Hengstebeck[2], Sarah Grzemski[3]

Abstract: Mit der wachsenden Zahl an Studierenden und Mitarbeitenden an der RWTH Aachen University, sowie der angebotenen Dienste des IT Centers, steigt die Herausforderung an den 1st-Level-Support, Anfragen effizient und zufriedenstellend zu bearbeiten. Es benötigt Breitenwissen und Struktur. Um sich den Herausforderungen zukünftig stellen zu können, hat das IT-ServiceDesk ein Qualitätsmanagement nach DIN EN ISO 9001:2015 eingeführt. Dokumentation, Wissensmanagement und Prozess-Auditierung unterstützen die Qualität der Serviceleistungen. Die Einführung ist ein nicht zu unterschätzender Aufwand, das Ergebnis aber eine Erleichterung und Unterstützung des Arbeitsalltags.

Keywords: Audit, Wissensmanagement, 1st-Level-Support, Dokumentation, QMS, DIN EN ISO 9001:2015, ServiceDesk, IT Service Management, RWTH Aachen University

1 Einleitung

Neue Aufgaben, ein stetig wachsendes Service-Portfolio, steigende Studierendenzahlen, komplexere Anfragen und fluktuierende Mitarbeitende sind die Herausforderungen, denen sich das IT-ServiceDesk stellen muss.

Als Single Point of Contact (SPoC) ist das IT-ServiceDesk für die Kommunikation zwischen dem IT Center und den Nutzenden der angebotenen IT-Dienste der RWTH Aachen verantwortlich.

Das Service-Portfolio des IT Centers erstreckt sich von Basisdiensten (Bereitstellung der IT-Infrastruktur, wie z.B. Netzanbindung, Telefon, E-Mail und Campus-Management) über das Angebot des High-Performance-Computing (HPC) bis zu Webhosting und anderweitigen serverbasierten Services. Insgesamt werden rund 50 verschiedene Services bzw. Teilservices angeboten [vgl. IT17].

Der Kundenstamm[4] des IT Centers umfasst neben über 43.721 Studierenden und 7.952

[1] IT Center der RWTH Aachen University, IT-ServiceDesk, Seffenter Weg 23, 52074 Aachen, pieters@itc.rwth-aachen.de
[2] IT Center der RWTH Aachen University, IT-ServiceDesk, Seffenter Weg 23, 52074 Aachen, hengstebeck@itc.rwth-aachen.de
[3] IT Center der RWTH Aachen University, IT-ServiceDesk, Seffenter Weg 23, 52074 Aachen, grzemski@itc.rwth-aachen.de

Mitarbeitenden der RWTH Aachen (Stand: 2015 vgl. [RW17]) - Tendenz steigend, auch diverse Kooperationspartner, wie die TU9 und das Deutsche Historische Institut (DHI) in Paris.

Die Bearbeitung von Anfragen aus diesem Kundenstamm und die gleichzeitige Bereitstellung von dazu benötigtem Wissen ist zum einen die Aufgabe des IT-ServiceDesks, jedoch gleichzeitig auch seine Herausforderung. Das IT-ServiceDesk besteht aus 33 Mitarbeitenden, die sich aus 11 festangestellten Mitarbeitenden, 7 Auszubildenden (Kauffrau/-mann für Dialogmarketing) und 15 studentischen Hilfskräften zusammensetzt. Neben dem 1st-Level-Support für die oben genannten Dienste des IT Centers bietet das IT-ServiceDesk auch eigene Dienste an, wie z.B. die Gerätesprechstunde für die Unterstützung bei Softwareinstallationen und Konfiguration des Eduroam-Zugangs auf mobilen Endgeräten und den hochschulweiten Druckservice. Innerhalb des IT Centers übernimmt das IT-ServiceDesk die IT-Administration, also beispielsweise die Beschaffung, Installation und Instandsetzung für die Arbeitsplatzausstattung der Mitarbeitenden (rund 200 Arbeitsplätze) und der Auszubildenden zum Mathematisch-technischen-Software-Entwickler (Matse, rund 120 Geräte). Der Support wird über fünf Kanäle angeboten: E-Mail, Telefon, Chat Support, Ticket-Portal und persönlicher Kontakt vor Ort. Im Jahr 2015 wurden insgesamt 55.911 Kundenanfragen an das IT-ServiceDesk gestellt und bearbeitet.

In diesem Zusammenhang ergeben sich verschiedene Anforderungen an die Abteilung. So muss zum einen sichergestellt sein, dass ein übergreifendes Breitenwissen über die verschiedenen Dienste und den damit einhergehenden Support-Prozessen bei allen Mitarbeitenden vorhanden ist. Zum anderen müssen notwendige und weiterführende support-relevante Informationen aktuell und für jeden Mitarbeitenden schnell verfügbar sein. Es muss sichergestellt sein, dass definierte Support-Prozesse auch entsprechend funktionieren und umgesetzt werden – die Dokumentation muss der realen Arbeitsweise entsprechen.

Eine weitere Herausforderung stellt die Kundenkommunikation dar. Dabei ist es essentiell, schnell erfassen zu können, welche Informationen für die jeweilige Problemstellung relevant und wichtig sind. Es ist wichtig, die richtigen Fragen zu stellen und angemessene, adressatengerechte Erklärungen zu geben, um den Kunden effiziente und zufriedenstellende Problemlösungen zu bieten. Neben der Kommunikation mit den Kunden ist es auch wichtig, dass alle Anfragen im verwendeten ITSM-Tool korrekt erfasst werden.

Die Aufgabe des IT-ServiceDesk ist repräsentativ für das IT Center. Dies bedeutet, dass die geleistete Qualität im IT-ServiceDesk maßgeblich die Kundenwahrnehmung beeinflusst. Die Kunden nehmen die Leistung des IT-ServiceDesk als erstes wahr und bilden auf dieser Grundlage ihre Meinung über das gesamte IT Center.

[4] Die Bezeichnung Kunde ist in diesem Kontext auch gleichzusetzen mit den Begriffen „Anwender“ und „Nutzende“

Um stets eine hohe Servicequalität gewährleisten zu können, hat sich die Abteilung IT-ServiceDesk - als erstes seiner Art an einer deutschen technischen Universität - entschlossen, sich einem Zertifizierungs- und damit jährlichen Auditierungsprozess zu stellen. Die im ITSM-Bereich oft verwendeten Normen DIN EN ISO 20000 und 20000-1 sind stark verwandt mit der DIN EN ISO 9001, betrachten aber neben dem Qualitätsmanagement auch ITSM-Prozesse. Das IT-ServiceDesk ist im IT Center Teil dieser Prozesse, diese werden Abteilungs-extern gesteuert und verantwortet. Vor diesem Hintergrund und dem Blick auf die Qualitätsverbesserung des Supports zertifizierte sich das IT-ServiceDesk im Jahr 2016 nach den DIN EN ISO 9001:2015.

2 Die DIN EN ISO 9001

Das Qualitätsmanagement nach DIN EN ISO 9001 befasst sich grundlegend mit der Steuerung und Optimierung von Arbeitsabläufen, Prozessen sowie den vorhandenen Strukturen und Ressourcen in einer Organisation. Die Norm dient der Selbstkontrolle und letztlich auch der Steigerung der Kundenzufriedenheit. Um sich nach der DIN EN ISO 9001 zertifizieren zu lassen, muss sich die Organisation mit der internen Prozesslandschaft, den Wechselbeziehungen - auch zu organisationsexternen Prozessen - und den definierten Rollen von Personen intensiv befassen. Der Nutzen von Qualitätsmanagement für die Organisation ergibt sich aus der Selbstbetrachtung [vgl. QU15]. Dies bedeutet konkret neben der Feststellung von Verbesserungspotenzialen in Prozessen und Ressourcen (Weiterbildung) auch die Validierung erreichter Qualitätsziele. Aus Potenzialen lassen sich Maßnahmen festlegen. Durch immer wiederkehrende Audits der einzelnen Teilbereiche des Qualitätsmanagementsystems (QMS) kann sicher- bzw. festgestellt werden, dass/ob gesetzte Maßnahmen umgesetzt worden sind und tatsächlich eine Verbesserung herbeigeführt wurde (vgl. [Wg14]).

Die aktuell gültige Fassung DIN EN ISO 9001:2015 wurde im November 2015 veröffentlicht und löste damit die Fassung DIN EN ISO 9001:2008 ab. In der neuen Fassung wird der Fokus noch intensiver auf den Kontext der Organisation gelegt, d.h. welche äußeren Faktoren (interessierte Parteien), sowie Risiken, in der Organisation Einfluss auf die Produktivität und Qualität der Prozesse haben. Im Allgemeinen bietet die neue Fassung mehr Freiraum für Interpretation und im Speziellen mehr Flexibilität in der Art und Weise der Normumsetzung. Dies erfordert für einen (externen) Auditor einen Mehraufwand, da jede Organisation einen sehr unterschiedlichen Kontext abbilden kann.

Eine weitere große Änderung ist die Übernahme der Verantwortung für ein QMS. Anders als in der Fassung 9001:2008 liegt die Verantwortung in der aktuellen Fassung deutlich höher, nämlich bei der Leitung der Organisation. So ist es z.B. nicht mehr notwendig, aber weiterhin möglich, einen eigens für das QMS abgestellten Qualitätsbeauftragten (QB) zu bestimmen. Die Leitung übernimmt die Verantwortung, dass Prozesse die geforderten Ergebnisse liefern, Ziele erreicht werden und das

Qualitätsmanagement von den Mitarbeitenden gelebt wird.

3 Einführung im IT-ServiceDesk

3.1 Struktur des IT-ServiceDesk

Das IT-ServiceDesk gliedert sich in die Gruppen „IT-Support“ und „Qualitätsmanagement.“

Die Gruppe „IT-Support“ ist verantwortlich für die Kundenkommunikation und die strukturierte Bearbeitung der Kundenanfragen. Die Ticketbearbeitung findet allerdings in beiden Gruppen statt, d.h. alle Mitarbeitenden leisten neben den gruppenbezogenen Tätigkeiten Support für die Endnutzer und arbeiten sehr eng zusammen.

In der Gruppe „Qualitätsmanagement“ liegt der Schwerpunkt auf der Vermittlung und Bereitstellung von Wissen, Informationen und Prozessen, sowie der Auswertung von Reporting-Ergebnissen (zur Überprüfung der gesetzten Qualitätsziele). Es werden IT-ServiceDesk spezifische Audits geplant und durchgeführt, sowie die Dokumentation des QMS gepflegt.

Im Rahmen der Aufwandsabschätzung für die Einführung eines QMS notwendigen Ressourcen wurde durch die Initiierung ein zusätzlicher Mitarbeitender eingestellt.

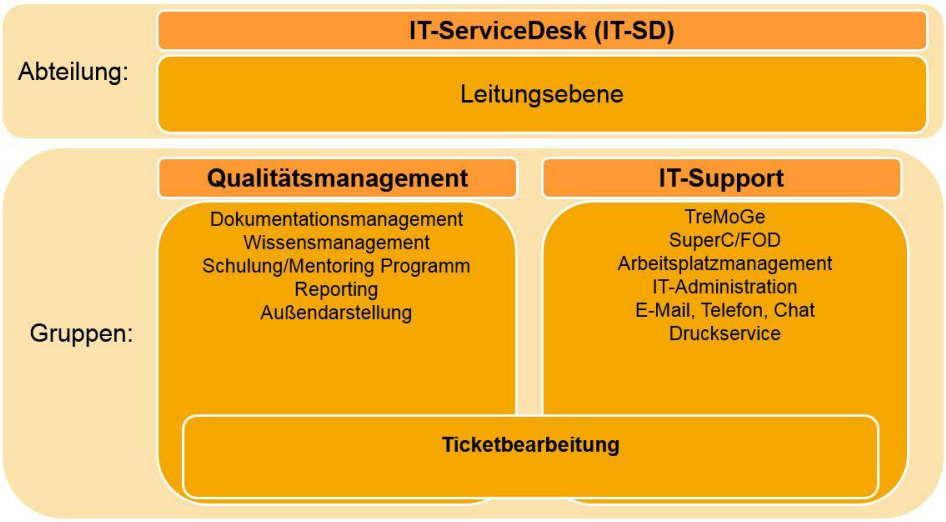


Abbildung 1: Organisation im IT-ServiceDesk; TreMoGe= Treffpunkt Mobile Geräte; SuperC = Spezieller IT-ServiceDesk Standort

Dieser nahm, die in der Fassung DIN EN ISO 9001:2008 noch geforderte Rolle des Qualitätsbeauftragten ein. Die Gruppe bestand somit zu Beginn des Projektes nur aus zwei Personen. Im fortlaufenden Projekt ist sie aktuell auf sieben Mitarbeitende angewachsen.

3.2 Erfolgsfaktor: Akzeptanz in der Praxis

Um Qualitätsmanagement zielgerichtet umzusetzen, ist der entscheidende Erfolgsfaktor eine hohe Bereitschaft für das Thema innerhalb der Organisation (hier IT-ServiceDesk) zu schaffen. Die Einführung bringt Veränderungen mit sich, die durch Mitarbeitende oftmals zunächst skeptisch betrachtet werden (vgl. [Wg14]). Um von Beginn an Akzeptanz zu schaffen, ist es daher notwendig, alle Mitarbeitende an der Gestaltung des QMS zu beteiligen. Zunächst ist es essentiell, entsprechenden Rahmenbedingungen zu schaffen, d.h. Verantwortlichkeiten zu klären, verfügbare Ressourcen zu definieren und zeitliche Vorgaben zu fixieren, um auf Basis dessen, das Thema vorzubereiten. Diese Vorbereitung kann beispielsweise in Abteilungsmeetings stattfinden.

Im IT-ServiceDesk wurde sich darauf konzentriert, zunächst bestehende, bekannte interne Prozesse zu betrachten. Darauf aufbauend wurden verbesserte Prozessbeschreibungen erstellt. In wöchentlichen Abteilungsmeeting wurden diese besprochen und durch das gesamte Team ergänzt und korrigiert.

Um Akzeptanz für das Qualitätsmanagement zu schaffen, wurden die neu gestalteten Dokumente vor der Implementierung in die bereits bestehende Dokumentation im Rahmen der Einarbeitung neuer Mitarbeitender erprobt: Für die Erklärung von Zusammenhängen und Arbeitsabläufen wurden beispielsweise die neuen Prozessbeschreibungen genutzt, sodass neben der Betreuung durch einen Mentor auch das selbstständige Erarbeiten von Themen möglich war.

Erst nach dieser Erprobungsphase und der positiven Bewertung der neuen Dokumentation wurden die festen Mitarbeitenden intensiver in die Prozessbeschreibungen einbezogen.

Angestrebt wurde ein Wandel von einem konsumierenden zu einem aktiven teilnehmenden Mitarbeitenden, um das Gefühl von Beteiligung und Verantwortung zu stärken. Die Herangehensweise zur finalen Abnahme der Dokumentation über den QB und die Gruppenleitung bis hin ins Abteilungsmeeting hat sich bewährt. Im Alltag konnte durch diese Herangehensweise das Grundverständnis von qualitätsgesicherter Dokumentation (z.B.: Genauigkeit in der Formulierung, exakte Informationsübergabe, Validierung durch mehrere Parteien) verbessert werden. Neu geschaffene Bereiche wie die Organisation von Schulungen und das Wissensmanagement konnten bereits von diesem Prozess profitieren. Das Verständnis wurde vor allem dadurch gewonnen, dass sich der Mehraufwand, der durch die Einführung entstanden war, schnell als lohnend herausgestellt hat. Dies bedeutet konkret eine Entlastung der Mitarbeitenden unter anderem, z.B. auf Grund der schnelleren und zuverlässigeren Auffindbarkeit von

Informationen in stressigen und arbeitsintensiven Supportsituationen. Des Weiteren verkürzte sich die Einarbeitungszeit von neuen Mitarbeitenden. Zusätzlich ist es förderlich, dass die Abteilungsleitung (als hauptverantwortliche Instanz) mit gutem Beispiel voranging und vorangeht und Qualitätsmanagement vorlebt. Im Rahmen einer Zwischenbewertung der Umstrukturierung durch das Projekt, waren die Mitarbeitenden einvernehmlich positiv zum Qualitätsmanagement gestimmt.

Im Zuge der Einführung sind auch durch die Mitarbeitenden selbst neue Maßnahmen entstanden, welche das QMS weiter ausgebaut haben. Dazu zählen unter anderem die Ticketbesprechung, d.h. eine anonymisierte Aufbereitung von Supportfällen, in denen inhaltliche Mängel, aber auch positive Beispiele durch die Mitarbeitenden selbst ausgewählt und vorgestellt werden. Grundsätzlich wird die Gruppe Qualitätsmanagement bei Veränderungen miteinbezogen und agiert als Qualitätskontrolle, beispielsweise bei der Dokumentation[5] und Prozessgestaltung.

3.3 Qualitätspolitik und Ziele

In der DIN EN ISO 9001 wird eine Qualitätspolitik und deren Bekanntmachung gefordert. Im Falle des IT-ServiceDesks legt diese gemeinsame Leitlinie ein besonderes Augenmerk auf die Förderung der Kundenorientierung und den kollegialen Umgang miteinander. Kunde bedeutet in diesem Zusammenhang nicht nur der Endkunde, sondern z.B. auch interne Beziehungen zu Fachabteilungen und Stabsstellen im IT Center.

Die Qualitätspolitik und die gesetzten Qualitätsziele müssen miteinander korrelieren (vgl. [QU2015]). Unter dem Gesichtspunkt gemeinsam an einem Ziel zu arbeiten, werden nicht nur arbeitsbezogene Qualitätsziele verfolgt, sondern auch soziale Ziele. Wird in der Qualitätspolitik das Wohlbefinden der Mitarbeitenden explizit als wichtiges Ziel festgelegt, muss sich dieses auch in den messbaren Qualitätszielen widerspiegeln. Hierzu wird jährlich eine Mitarbeiterzufriedenheitsbefragung durchgeführt.

Die klassischen auswertbaren Qualitätsziele im Service-Bereich sind die Auswertungen von Reaktionszeiten (wann erhält der Kunde eine erste Rückmeldung) und Erstlösungsraten (Bearbeitung ohne 2nd-Level-Einheiten) sowie die Kennzahlen zur Erreichbarkeit über den Telefonkanal. Das IT Center bietet einen breiten Katalog von Diensten an, dadurch ist eine Auswertung auf Dienste-Basis möglich. Diese Unterscheidung ist zwingend zu berücksichtigen, denn aus technischen und organisatorischen Gründen ist es nicht möglich, in allen Diensten Erstlösungen im 1st-Level-Support zu erzielen. Bestimmte Bearbeitungen können nur im 2nd-Level-Support erfolgen (z.B.: die initiale Einrichtung von Diensten).

Die definierten Qualitätsziele sind realistisch und im ersten auch kleinschrittig betrachtet worden. Beispielsweise ist es im Kontext des QMS des IT-ServiceDesk unrealistisch eine Verringerung der Ticket-Zahlen anzustreben, da diese (zusätzlich äußerer Faktoren

[5] Dazu wurde ein Dokumentationsteam u.a. auch aus studentischen Mitarbeitenden etabliert

wie z.B.: wachsende Studierendenzahlen) unter anderem von der Gesamt-Service-Leistung des IT Centers abhängen. Im IT-ServiceDesk wurden anhand einer vorangegangenen Auswertung die folgenden Qualitätsziele definiert:

1. **Telefonische Erreichbarkeit:** Es wird angestrebt, dass 80 % aller eingegangenen Anrufe im Jahr 2015 durch die Mitarbeitenden des IT-ServiceDesk angenommen wurden.

2. **Einhaltung der Analysezeit:** Es wird angestrebt, dass 85 % aller im Jahr 2015 eingegangenen E-Mails innerhalb von einer Stunde klassifiziert und kategorisiert werden.

3. **Einhalten der Reaktionszeit:** Es wird angestrebt, dass auf 75 % aller im IT-ServiceDesk eingegangenen Anfragen innerhalb von 24 Stunden eine erste menschliche Antwort erfolgt.

4. **Erstlösungsrate:** Es wird angestrebt, dass 65 % der eingegangenen Anfragen im Jahr 2015 nur durch das IT-ServiceDesk bearbeitet wurden.

Die oben aufgeführten Ziele wurden im Jahr 2015 alle erreicht. Im Rahmen des kontinuierlichen Verbesserungsprozesses (KVP) werden weitere Kenngrößen zur Festlegung von Qualitätszielen erdacht, wie z.B. Bewertung des Einarbeitungsprozesses, der Einarbeitungsmappe und von Meetings.

Die Kundenzufriedenheit ist oberste Ziel und wird im IT Center einmal jährlich durch eine hochschulweite Kundenzufriedenheitsumfrage gemessen. Ein Themenschwerpunkt ist dabei der 1st-Level-Support des IT Centers. Hier ist es wichtig, eine Relation zwischen der Zufriedenheit und den kennzahlbezogenen Zielen herzustellen.

Qualitätsziele müssen schriftlich fixiert und bekannt gemacht werden. Auf diese Weise wird eine Transparenz für alle Mitarbeitenden gewährleistet, was als Ergebnis angestrebt wird. Sicherlich gibt es ein weites Spektrum an Ideen, wie qualitätsbezogen Informationen zusätzlich ausgewertet werden könnten. Als Beispiel dient hierzu die Frequenz der Zufriedenheitsauswertungen, die unterstützend getätigt werden können, um ein noch besseres Bild zu erhalten. Zu beachten ist, dass nicht alle Ideen innerhalb eines Bewertungszeitraums (von Audit zu Audit) umsetzbar sind und Ad-hoc-Einführungen vermieden werden müssen, da diese schnell zu einer Überlastung des QMS beitragen.

Ziele werden stets für einen bestimmten Zeitraum gesetzt. Es bietet sich daher an, die nächste Bewertung des QMS als Datum zu wählen. Die Datierung von Audits und Managementbewertung liegt in der eigenen Verantwortung. Die Durchführung muss zum Zweck einer Zertifizierung durch einen externen Auditor nachgewiesen werden. Aus diesem Grund bietet sich ein regelmäßiger, z.B. jährlicher Rhythmus an. Ziele werden mit der Zeit immer feiner definiert. Im Besonderen dann, wenn diese erfolgreich erreicht wurden.

3.4 Wissens- und Informationsmanagement

Das QMS wurde im Zuge der Einführung durch den Bereich Wissensmanagement erweitert. Dieser beschäftigt sich mit der Wissens-, Informations- und Kompetenzvermittlung. Im Support ist die Aufbereitung und Vermittlung von Wissen und Informationen ausschlaggebend für den Erfolg. Allerdings ist es stets eine Herausforderung im Arbeitsalltag, neues Wissen/Informationen zu vermitteln und bereits bekanntes zu festigen. Als wirksame Methode hat sich die Mikro-Schulung erwiesen, in der Themen innerhalb der Abteilungsmeetings in bis zu zwanzigminütigen Schulungen vermittelt werden. Diese Art der Wissens- und Informationsvermittlung eignet sich besonders gut für den Support-Bereich, da alle Mitarbeitenden kontinuierlich an denselben Support-Themen arbeiten. Wissensmanagement wurde ein wichtiger Teil des QMS und alle Wissens- und Informationsprozesse, wie z.B. die Einarbeitung von neuen Mitarbeitenden, sind in den Bereich übergegangen.

Im Zusammenhang mit Wissens- und Informationsmanagement ist Dokumentation ein wesentlicher Aspekt für die Bearbeitung von Anfragen im IT-ServiceDesk. Alle Mitarbeitenden pflegen den Dokumentationsprozess mit, indem wichtige Informationen und Erkenntnisse an das Dokumentationsteam übermittelt werden, welches die Informationen für die Wissensdatenbank aufbereitet. Seit der Etablierung des Dokumentationsteams ist ein stetiger Anstieg der Dokumentationsanfragen zu verzeichnen. Daraus lässt sich schließen, dass die Aktualität und Qualität der Dokumentation stetig steigt.

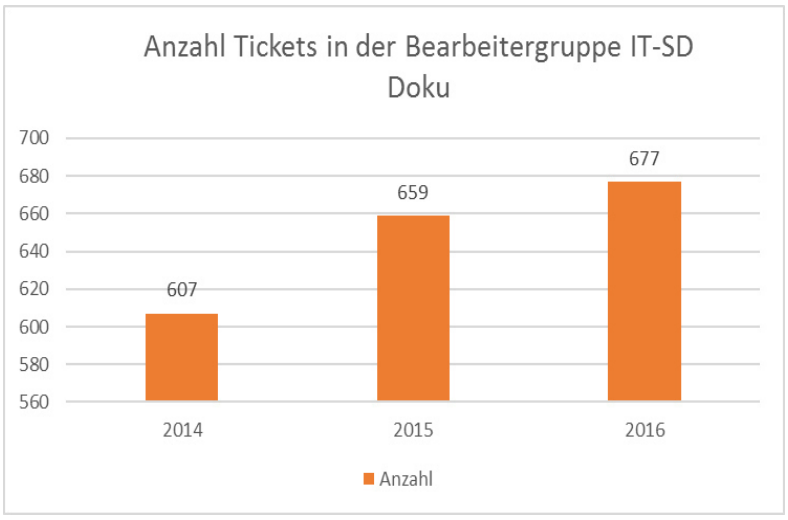


Abbildung 2: Anzahl der bearbeiteten Tickets in der Bearbeitergruppe IT-SD Doku

Die Aufbereitung von Wissen/Informationen hat sich als so essentiell erwiesen, dass eine Erweiterung des bestehenden Ticketsystems angestrebt wird. Mit dieser wird durch eine

kompatible Wissensmanagement-Software die Suche und Pflege von Informationen effizienter gestaltet. Wissensmanagement ist in die Fassung DIN EN ISO 9001:2015 auch als Themenbereich aufgenommen worden. Wissensmanagement bietet zum einen die Möglichkeit, das vorhandene Wissen in einer Organisation zu steuern und bietet eine Planungsgrundlage für Ressourcen, da erfasst wird, welche Kompetenz z.B. beim Weggang eines Mitarbeitenden ersetzt oder neu vermittelt werden muss.

Das für den Support benötigte Breitenwissen ist die Grundvoraussetzung, um effizient Supportleistungen im IT-ServiceDesk zu erbringen. Daher werden einmal jährlich bei der Einstellung neuere Auszubildender und studentischen Hilfskräften Schulungen durchgeführt, um den neuen Mitarbeitenden Themen wie Basisdienste, Kommunikation und Fachdienste näher zu bringen. Jeder neue Mitarbeitende erhält zusätzlich einen Mentor, welcher anhand eines Einarbeitungsprozesses die Arbeit der Abteilung erläutert. Die Auditierung von Kernprozessen befasst sich, aufgrund der Wichtigkeit mit dem Umgang von Wissen und Informationen, intensiv mit der Nutzung der Dokumentation in der Ticketbearbeitung. Diese Verfahrensweisen werden jährlich einer Qualitätskontrolle unterzogen. Dies geschieht auf Basis von Feedback, welches z.B. während der Schulungsphasen oder nach dem Mentoring gesammelt und anschließend evaluiert wird. Die Ergebnisse fließen in den KVP ein.

3.5 Auditierung und Managementbewertung

Ein Audit ist ein Verfahren wonach ein Prozess bzw. Bereich auf festgelegte Kriterien hin geprüft wird, um Verbesserungspotentiale und Fehler zu ermitteln (vgl. [Wg14]). Erste Audits wurden im IT-ServiceDesk durchgeführt, als das QMS bereits eine Zeit lang in der Abteilung integriert war. Dies wurde bewusst so gewählt, da viele Prozesse erst im Zug der Einarbeitung erfasst wurden und demnach eine Eingewöhnungsphase nötig war. Ein Audit wird mit Mitarbeitenden durchgeführt, indem der Auditor sich Prozesse vorführen lässt und weiterführende Verständnisfragen stellt. Der Umgang mit Wissen/Informationen ist ausschlaggebend für die Supportleistung. Im Zuge der Bearbeitung von Anfragen wird die dafür notwendige Dokumentation somit gleichzeitig geprüft. Abweichungen bei der Bearbeitung von Supportfällen zu der in der Prozessbeschreibung dokumentierten Vorgehensweisen zeigen entweder eine veraltete Dokumentation oder eine zu komplexe Prozessbeschreibung auf.

Ebenso kann die Dokumentation nicht verständlich sein und damit eine fehlerhafte Bearbeitung des Supportfalls zur Folge haben. Aus diesen Betrachtungen lassen sich Schlüsse ziehen und korrigierende Maßnahmen ergreifen.

In einem Zertifizierungsaudit wird das QMS des Auftraggebers (in diesem Fall das IT-ServiceDesk) auf die Übereinstimmung mit den in der DIN EN ISO 9001 geforderten Inhalten geprüft. Solch ein Audit muss auch intern als Vorbereitung auf die Zertifizierung durchgeführt werden. Zusätzlich ist es verpflichtend, eine Bewertung des QMS und dessen Wirkung vorzunehmen.

Diese bezeichnet man als Managementbewertung. Sie betrachtet das gesamte QMS, d.h. es werden Ergebnisse aus Audits zusammengetragen, das Erreichen der Ziele bemessen und auch Maßnahmen validiert, die aus vorangegangenen Auditergebnissen festgehalten worden sind. Die Managementbewertung setzt die Einzelergebnisse in den Gesamtkontext und dient der Reflektion über die Qualität des Erfolges, den möglichen Chancen für die Zukunft, aber auch den neu erkannten Risiken. Es werden globale Maßnahmen ergriffen um das QMS zu verbessern.

Eine solche Bewertung zu verfassen, verbessert den Gesamtüberblick über die erbrachte Leistung und erleichtert den Blick auf das QMS von außen. Da in der Regel inmitten von Prozessen agiert wird, dient die Managementbewertung der Betrachtung von Erfolgen. Daraus ergibt sich ein Bild über die langfristige Entwicklung der Organisation (hier IT-ServiceDesk). Die Managementbewertung ist ein Zwischenbericht und wird bei weiteren Bewertungen als Meilenstein oder Grundlage betrachtet.

4 Fazit

Seit Mai 2016 ist das IT-ServiceDesk nach DIN EN ISO 9001:2015 zertifiziert.

Die Einführung eines Qualitätsmanagementsystems ist ein großer Veränderungsprozess in einer Einrichtung. Veränderungsprozesse haben die Eigenschaft, dass sie nur mit erheblichem Aufwand umzusetzen sind und eine gewisse Verpflichtung und Leidensfähigkeit voraussetzen. Erfolg hat man dann, wenn man die Mitarbeitenden, die sich in diesem Prozess bewegen, aktiv beteiligt. Nachhaltigkeit bedeutet im Support, dass jeder Mitarbeitende einen Anspruch an die geleistete Qualität hat, diese mitwirkend verbessern will und seinen Blick auf den Kunden richtet.

Das IT-ServiceDesk hat ein System eingeführt, das zum Ziel hat, diese Eigenschaften zu unterstützen und zu fördern. Es stellt sich die Frage, wieso das IT-ServiceDesk eine Zertifizierung umgesetzt hat und sich nicht mit der Einführung eines QMS begnügte, da schließlich keine Anforderung von Seiten des Kunden (oder des IT Centers) bestand. Das Zertifikat wird für drei Jahre ausgestellt und jährlich einer Überprüfung unterzogen. Ein Grund für die Zertifizierung war den „Zwang“ aufrecht zu erhalten, Ergebnisse zu erzielen und das QMS auch weiter zu entwickeln. Ohne die jährliche Überprüfung besteht die Gefahr, dass sich das QMS nicht weiterentwickelt. Das Erteilen des Zertifikats sieht das IT-ServiceDesk aber auch als Belohnung für zwei Jahre harte Arbeit an, und das zertifizierte IT-Servicedesk trägt zum positiven Außenbild des IT Centers bei. Des Weiteren unterstützt eine ISO-zertifizierte Supporteinrichtung die Einwerbung von Fördermitteln und wird von Fördermittelgebern positiv bewertet.

Qualitätsmanagement ist nun Regelbetrieb im IT-ServiceDesk und stellt sicher, dass der Support kundenorientiert stattfindet. Es unterstützt die Bestrebung die Supportleistung zu verbessern und Mitarbeitende zu entlasten. Ein Vorhaben, das von innen nach außen trägt. Durch das QMS wurde eine Basis geschaffen, anhand derer eine stetige

Verbesserung der Supportleistung möglich ist. Das System ist daher nie als abgeschlossen zu betrachten, sondern entwickelt sich iterativ mit fortlaufender Anwendung weiter, um möglichst effizient auf die wachsenden Anforderungen an das IT-ServiceDesk reagieren zu können.

Qualitätsgesicherte Prozesse im IT-ServiceDesk unterstützen angegliederte Prozesse im IT Center. Die Einführung des QMS trägt einrichtungsweit dazu bei, dass Qualitätsmanagement innerhalb des IT Centers ein wachsendes Thema ist und beispielsweise die Möglichkeiten qualitätsgesicherter, organisationsübergreifender Prozesse evaluiert werden.[6]

Literaturverzeichnis

[IT17] IT Center – kompletter Servicekatalog, www.itc.rwth-aachen.de/cms/IT-Center/Dienste/~evvb/kompletter-Servicekatalog/, Stand 29.03.2017

[QU15] Qualitätsmanagementsysteme – Anforderungen; Deutsche und Englische Fassung EN ISO 9001:2015, Beuth Verlag, Berlin, 2015

[RW17] RWTH Aachen University - Daten Fakten, www.rwth-aachen.de/go/id/enw/Daten-Fakten/, Stand: 03.04.2017

[Wg14] Weidner, G.: Qualitätsmanagement – Kompaktes Wissen – Konkrete Umsetzung – Praktische Arbeitshilfen, Hanser, München, 2014

[6] Eine Gesamt-Zertifizierung des IT Centers wird derzeit nicht angestrebt

Müller, Neumair, Reiser, Dreo Rodosek (Hrsg.): 10. DFN-Forum Kommunikationstechnologien,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 11

Leichtgewichtiges Dokumentenmanagement zur Unterstützung eines Service Management Systems am Beispiel des LRZ

Bastian Kemmler[1], Jule Anna Ziegler[1] und Andreas Lohrer[2]

Abstract:

Mit der Einführung eines Service Management Systems (SMS) gemäß anerkannter Frameworks wie ISO/IEC 20000, ITIL oder FitSM stehen betroffene Organisationen gleichzeitig vor der Herausforderung, Mechanismen zur angemessenen Dokumentensteuerung zu etablieren. Dadurch werden die oft ohnehin schon knappen Ressourcen zusätzlich durch die parallele Einführung eines Dokumentenmanagementsystems (DMS) zur Dokumentensteuerung belastet. Als Lösungsvorschlag wird daher ein auf der Norm ISO/IEC 20000-1 basierendes leichtgewichtiges Dokumentensteuerungsverfahren vorgestellt,welches den gesamten Dokumentlebenszyklus anhand von Abläufen und Status sowie erforderlichen Rollen betrachtet und sich im Rahmen eines Wiki-Systems realisieren lässt. Die prototypische und tatsächliche Umsetzung am LRZ erfolgt anhand des Wiki-Systems Atlassian Confluence und belegt die Wirksamkeit des Konzeptes. Eine Analyse erweiternder Plugins vertieft den Blick auf benötigte System-Komponenten.

Keywords: Service Management, Service Management System, Dokumentensteuerung, Dokumentenmanagementsystem, Atlassian Confluence

1 Einleitung

Die Einführung eines Service Management Systems (SMS) nach anerkannten Frameworks wie ISO/IEC 20000 [IS11], ITIL [11d, 11b, 11e, 11c, 11a], oder FitSM [IT16] stellt die betroffene Organisation neben vielen anderen Aspekten unweigerlich vor die Herausforderung Dokumente, wie Pläne, Richtlinien, Prozesse aber auch Aufzeichnungen zur Prozesskonformität geeignet zu dokumentieren und zu archivieren. So fordert beispielsweise die Norm ISO/IEC 20000-1:2011 die Entwicklung und Pflege von Service Managements relevanten Dokumenten [IS11, 4.3.1] sowie die Steuerung derartiger Dokumente [IS11, 4.3.2] und Aufzeichnungen [IS11, 4.3.1]. Auch im Anforderungskatalog von FitSM finden sich entsprechende Hinweise[IT16, GR2 Documentation].

Gleichzeitig mit dem nicht unerheblichen Aufwand zur Einführung eines SMS sind die entsprechenden Organisationen somit angehalten, Mechanismen zur angemessenen Dokumentation des Managementsystems zu etablieren. Die oft ohnehin schon knappen Ressourcen werden damit zusätzlich durch die parallele Einführung eines Dokumentenmanagementsystems (DMS) zur Dokumentensteuerung belastet. Erschwerend kommt hinzu,

[1] Leibniz-Rechenzentrum, Boltzmannstr. 1, 85748 Garching b. München, {vorname}.{nachname}@lrz.de
[2] Ludwig-Maximilians-Universität München, Oettingenstr. 67, 80538 München, andreas.lohrer@campus.lmu.de

dass relevante Literatur zur Dokumentation in SMS kaum vorhanden ist. Hilfreich sind neben den Anforderungen der Frameworks lediglich Werke die sich auf allgemeiner Ebene mit den Herausforderungen des Dokumentenmanagements beschäftigen [LK12, Gö14, SMS10], jedoch weit über die Konzeption eines leichtgewichtigen Dokumentenmanagements hinausgehen.

Bei genauerer Betrachtung der von den Managementsystemen geforderten Dokumentation zeigt sich, dass sich diese Dokumente in Aufzeichnungen und beschreibende Dokumentation unterscheiden lassen. Oft werden viele der Aufzeichnungen dabei von den ohnehin meist schon vorhanden unterstützenden Service Management Tools erfasst und archiviert. Zentrales Problem hinsichtlich der Dokumentensteuerung im Rahmen der Einführung eines SMS ist also letztendlich der Aufwand zur Einführung und zum Betrieb adäquater Mechanismen zur Steuerung der entsprechenden beschreibenden Dokumentation.

Eine ressourcenschonende, technische Unterstützung geeigneter Verfahren zur Dokumentensteuerung stellt dabei die Verwendung in der Organisation bereits vorhandener Tools, wie Wikisysteme, dar. Offen ist jedoch, ob ein evtl. vorhandenes Wikisystem die Anforderungen an eine Dokumentensteuerung nach den vorgestellten Frameworks erfüllen kann.

Es ergeben sich daher die folgenden Fragestellungen:

- Wie sieht ein schlankes Steuerungsverfahren für beschreibende Dokumente aus?
- Wie kann die Einführung des Verfahrens ressourcenschonend gestaltet werden?
- Welche Anforderungen ergeben sich somit an ein entsprechendes Tool?
- Kann ein Wiki als Basis für die Steuerung von SMS-Dokumenten dienen?

Basis der im folgenden beschriebenen Untersuchung sind die in der Norm ISO/IEC 20000 geforderten Aspekte der Dokumentensteuerung. Eine Übertragbarkeit auf das sehr verwandte SMS FitSM ist aufgrund der bei vielen Prozessen vorhandenen 1-zu-1-Beziehung beider Frameworks leicht gegeben. Auch eine Übertragbarkeit auf ein SMS nach ITIL scheint mit überschaubarem Aufwand realisierbar zu sein.

Abschnitt 2 erläutert Anforderungen aus der ISO/IEC 20000, Rollen, sowie Abläufe und Status zur Implementierung der Dokumentensteuerung. Anschließend werden in Abschnitt 3 die resultierenden technische Anforderungen beschrieben. In der prototypischen Umsetzung in Abschnitt 4 wird auf die Umsetzung des Dokumentenmanagementsystems anhand der Software Atlassian Confluence näher eingegangen. Dazu werden ausgewählte Plugin-Varianten evaluiert. Die Praxistauglichkeit wird in Abschnitt 5 mit Erfahrungen aus dem Live-Betrieb untermauert. Das Paper schließt mit der Zusammenfassung.

2 Verfahren zur Dokumentensteuerung

2.1 Anforderungen

Aus der Norm ISO/IEC 20000-1 lassen sich die folgenden verpflichtenden Anforderungen bzgl. eines Dokumentensteuerungsverfahrens hinsichtlich der beschreibenden Dokumentation des SMS ableiten:

A1 Erstellung und Pflege von beschreibenden Dokumenten, die eine effektive Planung, den Betrieb und die Steuerung des SMS sicherstellen. Insbesondere sind dies (1) Pläne und Ziele des Service Managements [IS11, 4.3.1a], (2) der SMS Plan [IS11, 4.3.1b], (3) Richtlinien und Pläne zu allen ISO/IEC-20000-Prozessen [IS11, 4.3.1c] sowie (4) die zugehörigen Prozessbeschreibungen [IS11, 4.3.1f] und (5) benötigten Verfahren [IS11, 4.3.1g], (6) der Servicekatalog [IS11, 4.3.1g], (7) Service Level Agreements (SLAs) [IS11, 4.3.1e] sowie (8) alle weiteren Dokumente auch externen Ursprungs, die vom Service Provider für einen effektiven Betrieb des SMS als notwendig eingestuft werden [IS11, 4.3.1h]

A2 Verantwortlichkeiten und Befugnisse zu gesteuerten Dokumenten müssen festgelegt werden [IS11, 4.3.2]

Das Dokumentensteuerungsverfahren sollte wenigstens ...

A3 die Freigabe von Dokumenten vor der Veröffentlichung enthalten [IS11, 4.3.2a]

A4 Festlegungen zur Kommunikation der ersten Freigabe oder einer freigegebenen Veränderung an die jeweilige Zielgruppe enthalten [IS11, 4.3.2b]

A5 Festlegungen zur Überprüfung und Pflege der Dokumente enthalten [IS11, 4.3.2c]

A6 sicherstellen, dass Veränderungen und der aktuelle Versionsstand eines Dokumentes identifiziert werden können [IS11, 4.3.2d]

A7 sicherstellen, dass relevante Versionen eines Dokuments stets zur Verfügung stehen [IS11, 4.3.2e]

A8 sicherstellen, dass Dokumente leicht identifizierbar und lesbar sind [IS11, 4.3.2f]

A9 sicherstellen, dass externe benötigte Dokumente identifiziert werden und ihre entsprechende Verbreitung gesteuert wird [IS11, 4.3.2g]

A10 verhindern, dass ungültig Dokumente unabsichtlich verwendet werden [IS11, 4.3.2h]

Weitere optionale Anforderungen bzgl. eines Dokumentensteuerungsverfahrens hinsichtlich der beschreibenden Dokumentation des SMS ergeben sich entsprechend aus Teil 4.3.2 der ISO/IEC 20000-2 [IS12, 4.3.2]. Ggf. sei ergänzend auf dieses Dokument verwiesen. Als zentrale optionale Anforderungen sollten jedoch die folgenden erwähnt werden:

Das Dokumentensteuerungsverfahren sollte ...

A21 das regelmäßige, wenigstens jährliche Review der Dokumente enthalten [IS12, 4.3.2]

A22 den Schutz der Dokumente vor Schäden, z.B. verursacht durch Umwelteinflüsse und Hardwarefehler, gewährleisten [IS12, 4.3.2]

A23 die Sichtbarkeit von Veränderungen an gesteuerten Dokumenten verbessern [IS12, 4.3.2]

A24 eine Versionierung für gesteuerte Dokumente vorsehen [IS12, 4.3.2a]

A25 Verantwortlichkeiten für Schreiben, Verändern, Review, Genehmigung, Entfernen und Archivierung von gesteuerten Dokumenten regeln [IS12, 4.3.2b]

A26 Aufzeichnungen zu Datum, Autor, Genehmigung und Zweck von Revisionen festhalten [IS12, 4.3.2c]

A27 Zugriff und Verteilung der Dokumente steuern [IS12, 4.3.2f]
A28 ein Genehmigungsverfahren enthalten [IS12, 4.3.2g]
A29 ein Verfahren zu Review, Update und erneuter Genehmigung enthalten [IS12, 4.3.2h]
A30 ein Verfahren zur Archivierung veralteter Dokumente enthalten [IS12, 4.3.2k]

2.2 Dokumentensteuerungsverfahren

2.2.1 Ablauf und Status

Damit das resultierende Dokumentensteuerungssystem den Anforderungen der ISO/IEC 20000 zur Dokumentensteuerung entspricht, definieren wir das folgende Verfahren. Die nachfolgenden Schritte stellen die Bearbeitungs- und Anwendungsphasen, die ein Dokument während seines Lebenszyklus durchläuft, dar. Entsprechende Phasenübergänge und zugehörige Status lassen sich der Abbildung 1 entnehmen.

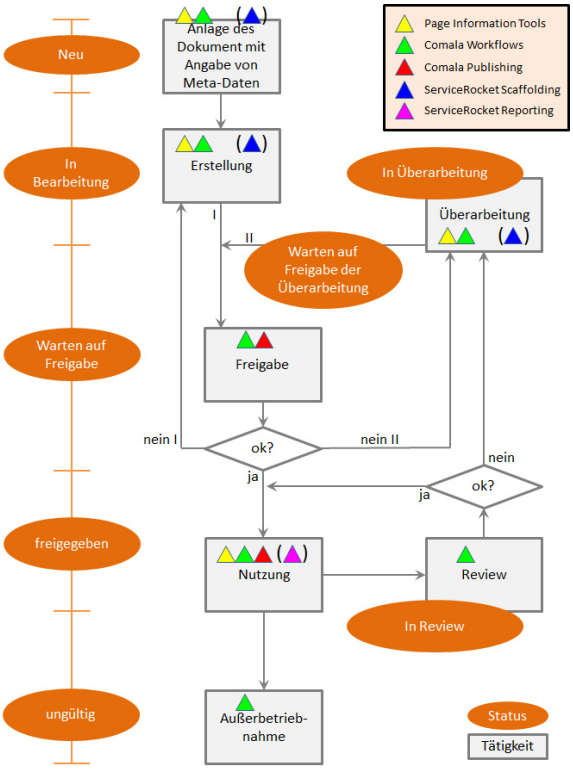


Abb. 1: Dokumentensteuerungsverfahren

Anlage: Der erste Schritt im Dokumentensteuerungsverfahren ist die Erstellung eines neuen Dokuments. Zusätzlich hinterlegt der Benutzer Metadaten, die zur Verwaltung des Dokuments erforderlich sind. Diese sind wenigstens der Dokument-Eigentümer, der Ansprechpartner (regelmäßiger Bearbeiter), das Wiedervorlagedatum sowie der Dokument-Typ. (vgl. A1, A2, A8, A9, A25)

Erstellung: Nach Anlage des Dokuments beginnt der Benutzer mit der ersten Version des Dokuments. (vgl. A1)

Freigabe: Ist die Bearbeitung des Dokuments abgeschlossen, wird es vom Autor zur Kontrolle an den zuständigen Eigentümer weitergeleitet. Er entscheidet über die Freigabe der Erstversion. (vgl. A3, A28)

Nutzung: Nach der Dokumentenfreigabe ist das Dokument von berechtigten Mitarbeitern in Verwendung. (vgl. A1, A23, A27)

Review: Bei Ablauf der Gültigkeitsdauer ist das Dokument vom Eigentümer auf dessen Aktualität zu überprüfen. Sofern keine Änderungen notwendig sind, wird es den Benutzern wieder zur Verfügung gestellt, andernfalls geht es zur Überarbeitung zurück an den Autor. (vgl. A5, A21, A26, A29)

Überarbeitung: Der Autor stellt die Aktualität des Dokuments sicher. Im Anschluss geht das aktualisierte Dokument zur Freigabe zurück an den Eigentümer. (vgl. A1, A5, A29)

Außerbetriebnahme: Veraltete Dokumente werden vom Eigentümer für ungültig erklärt. (vgl. A10, A30)

2.2.2 Rollen

Verantwortlich für die Gestaltung und Umsetzung des Verfahrens ist der **Owner Documentation Management**. Er überprüft die Konformität, Wirksamkeit und Einhaltung des Verfahrens und ist für die kontinuierliche Verbesserung verantwortlich.

Der **Documentation Manager** unterstützt den Owner Documentation Management bei der Gestaltung, Umsetzung und Überprüfung des Verfahrens. Weiterhin errichtet er geeignete weitere Verfahren zur Umsetzung des Dokumentationsmanagements und überwacht diese regelmäßig.

	Dokument-Eigentümer	**Ansprech-partner**	**Dokument-Empfänger**
Anlage	A	R	
Erstellung	A	R	C
Freigabe	A, R	C	I
Nutzung	A		R
Review	A, R	C	
Überarbeitung	A	R	C
Außerbetriebn.	A, R	I	I

R = Responsible[3] C = Consulted[3]
A = Accountable[3] I = Informed[3]

Tab. 1: RACI-Matrix

Der **Dokument-Eigentümer** entscheidet über die Errichtung, Veränderung und Löschung des jeweiligen Dokumentes. Er wird je nach Dokument-Typ entsprechend festgelegt.

Der **Ansprechpartner** ist aktiv an der Gestaltung des Inhalts eines Dokuments beteiligt und involviert.

Die **Dokument-Empfänger** sind die Interessenten für eine Dokumentation. In der Regel sind dies wenigstens alle Mitarbeiter der Organisation und/oder Anwender, Kunden.

Die Zuordnung der jeweiligen Verantwortlichkeiten findet sich in der RACI-Matrix (Tabelle 1) im Überblick.

3 Anforderungen an die technische Umsetzung

Neben den Anforderungen die die ISO/IEC 20000 an ein Dokumentenlenkungsverfahren stellt, berücksichtigen wir zusätzlich die Anforderungen zu einer bereits durchgeführten Expertenbefragung zur Schaffung einer nachhaltigen Dokumentenaktualität.

[3] Responsible: Durchführungsverantwortlich, Accountable: Verantwortlich (Kostenverantwortung und Rechenschaftspflicht), Consulted: Wird um Rat gefragt, Informed: Wird informiert.

Dokumentenverwaltung:
TA1 Import/Erfassung verschiedener Dokument-Dateitypen (vgl. A1, A9)
TA2 Der Inhalt der Dokumente soll such- und bearbeitbar sein (vgl. A1, A8)

Dokumentlebenszyklus:
TA3 Verfahrenskonforme Steuerung/Verwaltung des Dokumentlebenszyklus (siehe 2.2)
TA4 Erstellung von Tasks für Benutzer während aller Status (Usability)
TA5 Freie Statusänderung soll zur Eskalation jederzeit möglich sein (Eskalation)

Dokument-Review-Verfahren:
TA6 Signalisierung eines ausstehenden Reviews nach Ablauf des Wiedervorlagedatums mittels Mail (vgl. A21, A29)
TA7 Erinnerung an abgelaufene Review-Frist per Mail (vgl. A21, A29)
TA8 Automatisches Erzeugen eines Tasks nach Ablauf des Wiedervorlagedatums zum Review (vgl. A21, A29)
TA9 Das Wiedervorlagedatum muss über Metadaten definierbar sein (vgl. A21, A29)
TA10 Berichtsmöglichkeit (Liste) zu anstehenden/überfälligen Reviews (vgl. A21, A29)

Dokumentenversionsverwaltung:
TA11 Getrennte Verwaltung freigegebener und nicht-freigegebener Dokumente (vgl. A10)
TA12 Gegenseitige Aufrufbarkeit verschiedener Versionen (Entwurf vs. gültige Version) eines Dokuments über einen Link (vgl. A7)
TA13 Unterschiede zwischen Versionen müssen abfragbar sein (vgl. A6, A23, A24)
TA14 Beim Übergang in Status "freigegeben" werden Dokumente automatisch veröffentlicht bzw. an Adressaten verteilt (vgl. A3, A7, A8, A28)

Rollen- und Zuständigkeitsverwaltung (vgl. A2, A25, A28, A29):
TA15 Bearbeitung/Freigabe von Dokumenten ist beschränkt, Lesen/Kommentar-Funktion ist für alle Benutzer möglich
TA16 Verwendung bereits vorhandener Personen-Gruppen
TA17 Personen-Gruppen sollen z.B. bei der Angabe der Reviewer genutzt werden können

Änderungshistorie:
TA18 Die Versionshistorie der Dokumente soll angezeigt werden können (vgl. A6, A23)
TA19 Statusänderungen der Dokumente sind Nachvollziehbar (vgl. A23, A26)
TA20 Anzeige einer Versionshistorie (vgl. A6, A23)

Metadatenverwaltung:
TA21 Metadaten sollen durchsuchbar sein (vgl. A8, A27)
TA22 Autom. Erzeugung/Anzeige von Metadaten für neue Dokumente (vgl. A6, A26)
TA23 Freigabe/Anzeige der Metadaten wie Dokument-ID, Versionsnr. (published), Dokument-Owner, Freigabedatum, Dokumenttyp und Wiedervorlagedatum (vgl. A6, A24)
TA24 Auswahl/Anzeige vordefinierter/standardisiert formatierter Metadaten (vgl. A26)

Sonstiges:
TA25 Vertretbare Kosten für Plugin-Lizenzen für bis zu 250 Benutzer
TA26 Hersteller der Lösung muss das Potential für langfristigen Support haben (vgl. A22)

Aufgrund der für Dokumente unauflösbaren Bindung zwischen Information und Datenträger sind im Verfahren ebenfalls genauere Angaben zur verwendeten technischen Unterstützung der Dokumentation, z.B. den Editor oder Ablageformen, erforderlich.

4 Prototypische Umsetzung mit Atlassian Confluence und Plugins

Zur Unterstützung des Wissensmanagements betreibt das Leibniz-Rechenzentrum (LRZ) als beispielhafte Organisation seit einiger Zeit das Enterprise-Wiki *Atlassian Confluence*. Obwohl in einer Vorstudie zunächst auch weitere Tools zur Umsetzung der Dokumentensteuerung in Betracht kamen und auch beispielhaft erprobt wurden, fiel die Entscheidung für die prototypische Umsetzung letztendlich zugunsten von Confluence aus. Insbesondere wurde die einfache Bedienung und niedrigen Einstiegshürden sowie die Erweiterbarkeit von Confluence mit evtl. notwendigen Plugins positiv bewertet. Weiterhin kam die Aussicht hinzu, dass Aufgrund der bereits erfolgten Einführung von Confluence als Enterprise-Wiki am LRZ auf den Betrieb einer zusätzlichen Technologieplatform verzichtet werden kann.

4.1 Plugins

Anhand von Plugins, die Confluence über Makros im Standardfunktionsumfang erweitern, können Dokumentmetadaten auf einem Seiten-Template automatisiert verwaltet werden. Als mögliche Kandidaten zur Unterstützung der Dokumentensteuerung wurden die folgenden Plugins untersucht:

Das Plugin **Adaptavist Page Information Tools** ermöglicht es, versteckte Seiteninformationen wie die für die Dokumentensteuerung notwendigen Metadaten zur Verwaltung der Versions- und Bearbeitungshistorie anzuzeigen.

Die Alternative zum Adaptavist Page Information Tools Plugin ist das **ServiceRocket Reporting Plugin** . Es bietet darüber hinaus die Möglichkeit zur Erzeugung von Übersichtstabellen und zur Definition von Filter- und Sortierkriterien.

Das **Comala Workflow Plugin** realisiert Workflow- und Genehmigungsprozesse als auch die Möglichkeit zur Erstellung von Tasks und Notifications für Benutzer. Zur Umsetzung eines Dokumentensteuerungsverfahrens nach Abbildung 1 ist dazu jeder Status mit der Plugin-eigenen Definitionssprache zu modellieren. Mit Hilfe von Events und Triggern wird das Konzept der Tasks / Notifications umgesetzt.

Um die getrennte Darstellung von Entwurfsdokumenten (Drafts) und publizierten Seitenversionen (Published) zu ermöglichen, können unterschiedliche Confluence-Bereiche definiert werden. Mit Hilfe des **Comala Publishing Plugins** kann der Publikationsprozess zwischen beiden Bereichen automatisiert werden, indem in den Dokumentenworkflow die Seitenpublikation integriert wird. Eine Verlinkung beider Bereiche ist ebenfalls möglich.

[4] inklusive vieler abhängiger Plugins

Das **ServiceRocket Scaffolding Plugin** bietet standardisierte Eingabeformulare und -vorlagen mit konfigurierbaren Pflichtfeldern, die für spezifizierte Benutzer (-gruppen) eingrenzbar sind.

4.2 Evaluation

Für den Prototyp und die notwendigen Tests der für ein Dokumentensteuerungsverfahren notwendigen Plugin-Erweiterungen wurde eine vom Produktivsystem unabhängige Testinstanz von Confluence betrieben. Dazu wurde bei der prototypischen Umsetzung zunächst auf Testlizenzen der Plugins zurück gegriffen. Als Testdokumente dienten LRZ-interne SMS-Prozessdokumente und -Richtlinienbeschreibungen, deren Lebenszyklus zukünftig überwacht und gesteuert werden soll. Diese ermöglichten ebenfalls die Ableitung benötigter Dokumentmetadaten für ein Atlassian Confluence Seiten-Template.

Aus Abbildung 1 ist ersichtlich an welcher Stelle des Verfahrens das jeweilige Plugin in der Testumgebung zum Einsatz kam.

Tabelle 2 verdichtet hingegen für welche Tool-Anforderung das jeweilige Tool ermöglichende Eigenschaften besitzt. Deutlich erkennbar ist dabei, dass sich die Tool-Anforderungsgruppen Dokumentlebenszyklus (TA3, TA4, TA5) und Dokument-Review-Verfahren (TA6, TA7, TA8, TA9, TA10) nicht ohne ein Workflowsystem wie das Plugin Comala Workflow realisieren lassen. Auch die Anforderungsgruppe Rollen- und Zuständigkeitsverwaltung profitiert maßgeblich von einem derartigen Workflowsystem. Für die Dokumentenversionsverwaltung (TA11, TA12, TA13, TA14) hingegen wird das Plugin Comala Publishing benötigt. Die Service Rocket Plugins unterstützen im wesentlichen die Metadatenverwaltung (TA21, TA22, TA23, TA24), sowie diesbezügliche Suchen (TA21) und Berichte (TA10). Leider war in der Testumgebung die Synchronisation der Metadaten zwischen den Plugins im Rahmen der Anforderung TA23 fehlerhaft, so dass die Anforderung beim ergänzenden Service Rocket Scaffolding Plugin als nicht erfüllt gewertet werden musste.

Anforderungen	**Confluence 5.7.3 Build 5781**	**Adaptavist Page Information Tools**	**Comala Workflow 4.6.2**	**Comala Publishing**	**Service Rocket Scaffolding** [4]	**Service Rocket Reporting**[4]
TA1	(✓)					
TA2	✓					
TA3			✓			
TA4			✓			
TA5			(✓)			
TA6			✓			
TA7			✓			
TA8			✓			
TA9			✓		✓	
TA10			(✓)			(✓)
TA11	✓			✓		
TA12				✓		
TA13	(✓)			(✓)		
TA14			✓	✓		
TA15			(✓)			
TA16	✓					
TA17			(✓)		✓	
TA18	✓					
TA19	✓					
TA20	(✓)					
TA21	(✓)				(✓)	(✓)
TA22		✓	✓			
TA23		✓		(✓)	✗	
TA24			(✓)		✓	
TA25	*	*	*	*	*	*
TA26	✓	✓	✓	✓	✓	✓

✓: ermöglicht Anforderung
(✓): ermöglicht Anforderung eingeschränkt
✗: Anforderung nicht erfüllt
*: siehe Abschnitt 4.2

Tab. 2: Evaluation

Um das nach den Anforderungen ebenfalls erforderliche Benachrichtiugngssystem zu realisieren wurden Warteschleifen und Pseudo-Status implementiert, die wiederholt an not-

wendige Bearbeitungsschritte erinnern (vgl. Abbildung 2). Grüne Kreise repräsentieren hierbei die automatischen Erinnerungen; blaue Kreise übernehmen das Task-Management. Weitere Details der Implementierung sind dem Comala-Workflow-Code im Anhang zu entnehmen.

Aufgrund der Evaluation (Tabelle 2) im Rahmen der prototypischen Umsetzung konnten die folgenden 3 alternativen Plugin-Kombinationen (Varianten) erarbeitet werden. Ebenfalls berücksichtigt wurde dabei Tool-Anforderung TA26, da Wartungsaufwand und Anschaffungskosten in etwa proportional zu der Anzahl der verwendeten Plugins steigen. Die ermittelten Vor- und Nachteile der Varianten finden sich in Tabelle 3.

Variante 1 **Nur Comala-Plugins** mit Page Information Tools

Variante 2 **Comala-Plugins & ServiceRocket Scaffolding-Plugin** mit Page Information Tools

Variante 3 **Comala-Plugins & ServiceRocket-Plugins** mit Page Information Tools

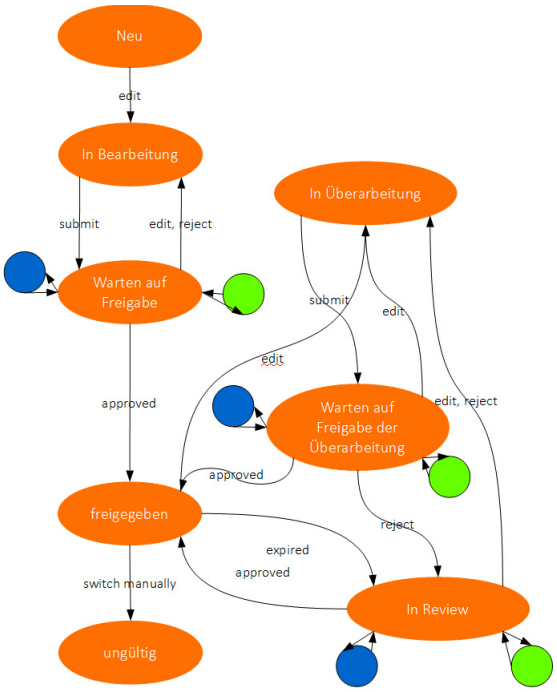


Abb. 2: Comala Workflow

5 Erfahrungen zur Einführung am LRZ

Mit dem Ziel eines leichgewichtigen Managements wurde für die Umsetzung am LRZ insbesondere aufgrund der geringen Anzahl der Plugins Variante 1 als Minimallösung gewählt und für die Dokumentensteuerung der beschreibenden Dokumentation des Service Management Systems verbindlich umgesetzt. Grundsätzlich entspricht dabei die Umsetzung den Erwartungen. Das entwickelte Verfahren zur Dokumentensteuerung ist mit der Tool-Auswahl nach Variante 1 und den Ergebnissen aus der Evaluationsphase mit dem Aufwand von ca. 1-2 Personenwochen in den Live-Betrieb überführbar. Hilfreich sind dabei insbesondere der in der Evaluationsphase entwickelte Comala Workflow-Code ohne Scaffolding-Plugin sowie das entsprechende Dokumententemplate. Zusammen mit der Erstellung dieser Grundlagen belief sich der Gesamtaufwand zur Einführung am LRZ auf etwa 4-6 Personenwochen. Zusätzlich benötigte Migrationszeiten von bestehenden Systemen zur Verwaltung von SMS-Dokumenten waren ebenfalls überschaubar.

	Vorteile	Nachteile
Variante 1	+ weitestgehende Anforderungserfüllung durch minimale Plugin-Kombination + relativ geringe Kosten + leichter wartbar aufgrund geringerer Plugin-Anzahl	- anstehende Reviews sind neben Info-Mails nur im nicht nach Datum sortierbaren Seiteneigenschaftsbericht sichtbar - fehlende Validierung bestimmter Workflow-Parameter ermöglicht Fehler im Workflow
Variante 2	+ Reduzierung der Eingabefehler durch Eingabeprüfung/ -vorgabe + Definition von Pflichtfeldern ermöglicht vollständige Formulareingaben + Scaffolding-Felder ermöglichen datentypgetreue Filterung und Sortierung	- große Anzahl impliziter Zusatz-Plugins erschweren Wartung, Fehleranfälligkeit und Support der ServiceRocket-Plugins - Pflichteingaben umgehbar durch Nicht-Aufrufen des „edit content“-Dialogs - relativ hoher Preis
Variante 3	+ ServiceRocket Reporting-Plugin kann u.a. Review-Datumswerte aller Dokumente filtern, sortieren und als Bericht darstellen + sonst gleiche Vorteile wie bei Variante 2	- Filter/Sortierungen können nur einmal statisch bei Berichterstellung gesetzt werden - sonst gleiche Nachteile wie bei Variante 2

Tab. 3: Plugin-Alternativen mit Vor- und Nachteilen

Die Einführungserfahrung am LRZ zeigt, dass der Kreis der Leseberechtigten des Draft-Bereichs je nach Organisationsgewohnheiten wohl überlegt definiert werden sollte. Einerseits ist es insbesondere bei Organisationen mit offener Kommunikationspolitik zur Einführung von neuen Prozessen aber auch Veränderung bereits existierender Prozesse förderlich, wenn viele qualifizierte Mitarbeiter bereits bei der Entstehung der jeweiligen Dokumente Einblick in die aktuellen Planungen erhalten, andererseits kann dies jedoch für ungeübte Wiki-Nutzer bei der globalen Suche zur Verwechselung mit freigegebenen Dokumenten führen. Unterschiedliche Dokument-IDs von Draft- und Produktiv-Version sind leider nur für Eingeweihte erklärbar.

Ein zusätzlicher minimaler Workflow vereinfacht die Veröffentlichung von Strukturelementen wie Prozessportalseiten im produktiven Bereich bei gleichzeitiger Beibehaltung von identischer Struktur in Draft- und Produktiv-Bereich.

Für die Außerbetriebnahme empfiehlt es sich ebenfalls im Produktiv-Bereich einen reduzierten Workflow zu implementieren, der verhindert, dass ungültig gewordene Dokumente im produktiven Bereich sichtbar bleiben.

Gut gewählte Stichwörter ermöglichen die Erstellung von Übersichtsseiten mit dem Makro „Seiteneigenschaftenbericht“.

Die Bedienung von Confluence ist auch für IT-ferne Mitarbeiter ohne großen Schulungsaufwand leicht erlernbar. Mitarbeiter mit Rollen im Dokumentensteuerungsverfahren müssen jedoch mit dem für eine Verfahrenseinführung üblichen Aufwand geschult werden. Der implementierte Worflow verhindert grobe Fehlbedienungen.

6 Zusammenfassung und Ausblick

Die mit der anforderungsgemäßen Einführung eines SMS (wie ISO/IEC 20000) einhergehende Etablierung eines DMS ist oft aufwendig und langwierig. Die ohnehin schon knappen Ressourcen werden somit zusätzlich durch die parallele Einführung belastet. Das vorgestellte leichtgewichtige Dokumentensteuerungsverfahren zusammen mit der Implementierung mit Atlassian Confluence kann hierbei zu schnellem Umsetzungserfolg führen. Wesentliche Anforderungen der Norm wurden dabei angemessen berücksichtigt.

Obwohl das Dokumentensteuerungsverfahren letztendlich in der favorisierten Variante 1 umgesetzt wurde sind Verbesserungen mittels der nachträglichen Ergänzung zu Variante 2 und Variante 3 möglich.

Auch eine Anwendbarkeit des Konzeptes auf viele Dokumenttypen eines Information Security Management Systems (ISMS), wie Policies, Prozessbeschreibungen, Verfahrensbeschreibungen, Pläne, etc. ist ohne weiteres gegeben, jedoch erfordern die ebenfalls vorhandenen spezielleren Dokumente des ISMS weitere Untersuchungen.

Literaturverzeichnis

[11a] ITIL continual service improvement. TSO The Stationery Office, London, 2nd ed.. Auflage, 2011.

[11b] ITIL service design. TSO The Stationery Office, London, 2nd ed.. Auflage, 2011.

[11c] ITIL service operation. TSO The Stationery Office, London, 2nd ed.. Auflage, 2011.

[11d] ITIL service strategy. TSO The Stationery Office, London, 2nd ed.. Auflage, 2011.

[11e] ITIL service transition. TSO The Stationery Office, London, 2nd ed.. Auflage, 2011.

[Gö14] Götzer, Klaus; Schmale, Ralf; Maier, Berthold; Rehbock, Klaus: Dokumenten-Management: Informationen im Unternehmen effizient nutzen. dpunkt-Verl., Heidelberg, 5., vollst. überarb. und erw. aufl.. Auflage, 2014.

[IS11] ISO/IEC: , ISO/IEC 20000-1:2011 - Information Technology - Service Management, Part 1: Service Management system requirements, 2011.

[IS12] ISO/IEC: , ISO/IEC 20000-2:2012(E) Information technology — Service management — Part 2: Guidance on the application of service management systems, 2012.

[IT16] ITEMO e.V.: , FitSM - Teil 1: Anforderungen: Teil 1: Anforderungen, 2016.

[LK12] Lutz, Alexandra; Kemper, Joachim: Schriftgutverwaltung nach DIN ISO 15489-1: Ein Leitfaden zur qualitätssicheren Aktenführung. Kommentar. Beuth, Berlin [u.a.], 1. aufl.. Auflage, 2012.

[SMS10] Steinbrecher, Wolf; Müll-Schnurr, Martina: Prozessorientierte Ablage: Dokumentenmanagement-Projekte zum Erfolg führen. Gabler, Wiesbaden, 2., überarb. und erw. aufl.. Auflage, 2010.

Anhang

Comala Workflow-Code ohne Scaffolding-Plugin:

```
{workflow:name=ITSM Workflow|label=workflowable|updatestatus=true}
    {workflowproperties:chart|rankdir=LR|size=8,8}
    {workflowparameter:doco|type=user|edit=true}
    {workflowparameter}
    {workflowparameter:durationreview|type=duration|edit=true}
    {workflowparameter}
    {workflowparameter:durationreminder|type=duration|edit=true}
        P3D
    {workflowparameter}
    {state:Neu|updated=In Bearbeitung|taskable=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
    {state}
    {state:In Bearbeitung|submit=Warten auf Freigabe|taskable=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
    {state}
    {state:Warten auf Freigabe|approved=freigegeben|rejected=In Bearbeitung|expired=ErinnerungWartenAufFreigabe|
        taskable=true|duedate=@durationreminder@|
            changeduedate=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
        {approval:Dokument wartet auf Ihre Freigabe|user=@doco@}
    {state}
    {state:freigegeben|final=true|expired=In Review|taskable=true|duedate=@durationreview@|changeduedate=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
    {state}
    {state:In Review|approved=freigegeben|rejected=In Ueberarbeitung|expired=ErinnerungInReview|taskable=true|
        duedate=@durationreminder@|changeduedate=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
        {approval:Dokument benoetigt Ihren Review|user=@doco@}
    {state}
    {state:In Ueberarbeitung|submit=Warten auf Freigabe der Ueberarbeitung|taskable=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
    {state}
    {state:Warten auf Freigabe der Ueberarbeitung|approved=freigegeben|rejected=In Review|expired=
        ErinnerungWartenAufFreigabeDerUeberarbeitung|taskable=true|
            duedate=@durationreminder@|changeduedate=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
        {approval:Ueberarbeitung wartet auf Ihre Freigabe|user=@doco@}
    {state}
    {state:ungueltig|taskable=true}
        {state-selection:states=Neu, In Bearbeitung, Warten auf Freigabe, freigegeben, In Review, In
            Ueberarbeitung, Warten auf Freigabe der Ueberarbeitung,ungueltig}
    {state}
    {state:ErinnerungWartenAufFreigabe}
    {state}
    {state:ErinnerungInReview}
    {state}
    {state:ErinnerungWartenAufFreigabeDerUeberarbeitung}
    {state}
    {state:TaskingWartenAufFreigabe|taskable=true}
        {task:name=Dokument wartet auf Ihre Freigabe|assignee=@doco@|note=Das neue Dokument wurde fertiggestellt
            und wartet nun auf Ihre Freigabe.}
    {state}
    {state:TaskingInReview|taskable=true}
        {task:name=Dokument benoetigt Ihren Review|assignee=@doco@|note=Das Dokument muss ggf. neu ueberarbeitet
            werden und benoetigt daher Ihren Review.}
    {state}
    {state:TaskingWartenAufFreigabeDerUeberarbeitung|taskable=true}
        {task:name=Ueberarbeitung wartet auf Ihre Freigabe|assignee=@doco@|note=Das Dokument wurde ueberarbeitet
            und wartet nun auf Ihre Freigabe.}
    {state}
    {trigger:statechanged|state=TaskingWartenAufFreigabe}
        {set-state:Warten auf Freigabe}
    {trigger}
    {trigger:statechanged|state=TaskingInReview}
        {set-state:In Review}
    {trigger}
    {trigger:statechanged|state=TaskingWartenAufFreigabeDerUeberarbeitung}
        {set-state:Warten auf Freigabe der Ueberarbeitung}
    {trigger}
    {trigger:statechanged|state=ErinnerungWartenAufFreigabe}
        {send-email:user=@doco@|subject=Dokument @page@ wartet auf Ihre Freigabe}
        Erinnerung: Das neue Dokument @page@ wurde fertiggestellt und wartet nun auf Ihre Freigabe.
        {send-email}
        {set-state:Warten auf Freigabe}
    {trigger}
    {trigger:statechanged|state=ErinnerungInReview}
        {send-email:user=@doco@|subject=Dokument @page@ benoetigt Ihren Review}
        Erinnerung: Das Dokument @page@ muss ggf. neu ueberarbeitet werden und benoetigt daher Ihren Review.
        {send-email}
        {set-state:In Review}
    {trigger}
    {trigger:statechanged|state=ErinnerungWartenAufFreigabeDerUeberarbeitung}
        {send-email:user=@doco@|subject=Ueberarbeitetes Dokument @page@ wartet auf Ihre Freigabe}
        Erinnerung: Das Dokument @page@ wurde ueberarbeitet und wartet nun auf Ihre Freigabe.
        {send-email}
        {set-state:Warten auf Freigabe der Ueberarbeitung}
    {trigger}
    {trigger:pageupdated|state=Warten auf Freigabe}
        {complete-task:task=Dokument wartet auf Ihre Freigabe|comment=Task wurde durch Bearbeitung des Users
            automatisch geschlossen.}
        {set-state:In Bearbeitung}
    {trigger}
```

```
85      {trigger:pageupdated|state=freigegeben}
86          {set-state:In Ueberarbeitung}
87      {trigger}
88      {trigger:pageupdated|state=In Review}
89          {complete-task:task=Dokument benoetigt Ihren Review|comment=Task wurde durch Bearbeitung des Users
                automatisch geschlossen.}
90          {set-state:In Ueberarbeitung}
91      {trigger}
92      {trigger:pageupdated|state=Warten auf Freigabe der Ueberarbeitung}
93          {complete-task:task=Ueberarbeitung wartet auf Ihre Freigabe|comment=Task wurde durch Bearbeitung des
                Users automatisch geschlossen.}
94          {set-state:In Ueberarbeitung}
95      {trigger}
96      {trigger:pagestatechanged|state=Neu}
97          {set-metadata:taskandapprovalset}0{set-metadata}
98          {set-metadata:durationreview}@durationreview@{set-metadata}
99          {set-metadata:wfstate}@workflow:state > name@{set-metadata}
100         {set-metadata:docuowner}[^@doco@]{set-metadata}
101     {trigger}
102     {trigger:pagestatechanged|state=In Bearbeitung}
103         {set-metadata:taskandapprovalset}0{set-metadata}
104         {set-metadata:durationreview}@durationreview@{set-metadata}
105         {set-metadata:wfstate}@workflow:state > name@{set-metadata}
106         {set-metadata:docuowner}[^@doco@]{set-metadata}
107     {trigger}
108     {trigger:statechanged|state=Warten auf Freigabe|@taskandapprovalset@=!1}
109         {send-email:user=@doco@|subject=Dokument @page@ wartet auf Ihre Freigabe}
110         Das neue Dokument @page@ wurde fertiggestellt und wartet nun auf Ihre Freigabe.
111         {send-email}
112         {set-metadata:taskandapprovalset}1{set-metadata}
113         {set-metadata:durationreview}@durationreview@{set-metadata}
114         {set-metadata:wfstate}@workflow:state > name@{set-metadata}
115         {set-metadata:docuowner}[^@doco@]{set-metadata}
116         {set-state:TaskingWartenAufFreigabe}
117     {trigger}
118     {trigger:pagestatechanged|state=freigegeben}
119         {set-metadata:datepublished}@datetime@{set-metadata}
120         {set-metadata:publisheddocversion}@content:latest version>version@{set-metadata}
121         {set-metadata:taskandapprovalset}0{set-metadata}
122         {set-metadata:durationreview}@durationreview@{set-metadata}
123         {set-metadata:wfduedate}@workflow:state>duedate@{set-metadata}
124         {set-metadata:wfstate}@workflow:state>name@{set-metadata}
125         {set-metadata:docuowner}[^@doco@]{set-metadata}
126         {publish-page}
127     {trigger}
128     {trigger:statechanged|state=In Review|@taskandapprovalset@=!1}
129         {send-email:user=@doco@|subject=Dokument @page@ benoetigt Ihren Review}
130         Das Dokument @page@ muss ggf. neu ueberarbeitet werden und benoetigt daher Ihren Review.
131         {send-email}
132         {set-metadata:taskandapprovalset}1{set-metadata}
133         {set-metadata:durationreview}@durationreview@{set-metadata}
134         {set-metadata:wfstate}@workflow:state > name@{set-metadata}
135         {set-metadata:docuowner}[^@doco@]{set-metadata}
136         {set-state:TaskingInReview}
137     {trigger}
138     {trigger:pagestatechanged|state=In Ueberarbeitung}
139         {set-metadata:taskandapprovalset}0{set-metadata}
140         {set-metadata:durationreview}@durationreview@{set-metadata}
141         {set-metadata:wfstate}@workflow:state > name@{set-metadata}
142         {set-metadata:docuowner}[^@doco@]{set-metadata}
143     {trigger}
144     {trigger:statechanged|state=Warten auf Freigabe der Ueberarbeitung|@taskandapprovalset@=!1}
145         {send-email:user=@doco@|subject=Ueberarbeitetes Dokument @page@ wartet auf Ihre Freigabe}
146         Das Dokument @page@ wurde ueberarbeitet und wartet nun auf Ihre Freigabe.
147         {send-email}
148         {set-metadata:taskandapprovalset}1{set-metadata}
149         {set-metadata:durationreview}@durationreview@{set-metadata}
150         {set-metadata:wfstate}@workflow:state > name@{set-metadata}
151         {set-metadata:docuowner}[^@doco@]{set-metadata}
152         {set-state:TaskingWartenAufFreigabeDerUeberarbeitung}
153     {trigger}
154     {trigger:pagestatechanged|state=ungueltig}
155         {set-metadata:taskandapprovalset}0{set-metadata}
156         {set-metadata:durationreview}@durationreview@{set-metadata}
157         {set-metadata:docuowner}[^@doco@]{set-metadata}
158         {set-metadata:wfstate}@workflow:state > name@{set-metadata}
159     {trigger}
160     {trigger:pageapproved|approval=Dokument wartet auf Ihre Freigabe}
161         {complete-task:task=Dokument wartet auf Ihre Freigabe|comment=Task wurde durch Approval des Users
                automatisch geschlossen.}
162     {trigger}
163     {trigger:pagerejected|approval=Dokument wartet auf Ihre Freigabe}
164         {complete-task:task=Dokument wartet auf Ihre Freigabe|comment=Task wurde durch Reject des Users
                automatisch geschlossen.}
165     {trigger}
166     {trigger:pageapproved|approval=Dokument benoetigt Ihren Review}
167         {complete-task:task=Dokument benoetigt Ihren Review|comment=Task wurde durch Approval des Users
                automatisch geschlossen.}
168     {trigger}
169     {trigger:pagerejected|approval=Dokument benoetigt Ihren Review}
170         {complete-task:task=Dokument benoetigt Ihren Review|comment=Task wurde durch Reject des Users
                automatisch geschlossen.}
171     {trigger}
172     {trigger:pageapproved|approval=Ueberarbeitung wartet auf Ihre Freigabe}
173         {complete-task:task=Ueberarbeitung wartet auf Ihre Freigabe|comment=Task wurde durch Approval des Users
                automatisch geschlossen.}
174     {trigger}
175     {trigger:pagerejected|approval=Ueberarbeitung wartet auf Ihre Freigabe}
176         {complete-task:task=Ueberarbeitung wartet auf Ihre Freigabe|comment=Task wurde durch Reject des Users
                automatisch geschlossen.}
177     {trigger}
178 {workflow}
```

Müller, Neumair, Reiser, Dreo Rodosek (Hrsg.): 10. DFN-Forum Kommunikationstechnologien,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 11

Leichtgewichtiges Security Incident und Event Management im Hochschulumfeld

Jule Anna Ziegler[1], Bastian Kemmler[1], Michael Brenner[1] und Thomas Schaaf[2]

Abstract: Für viele IT-Organisationen im Hochschulumfeld aber auch in kleineren und mittleren Unternehmen (KMUs) gewinnt die Etablierung eines prozessorientierten Informationssicherheitsmanagementsystems (ISMS) zunehmend an Bedeutung. Für die Gestaltung eines solchen Systems existieren verschiedene Rahmenwerke wie ISO/IEC 27000 oder IT-Grundschutz. Deren komplette Umsetzung strapaziert aber häufig die Ressourcen kleinerer IT-Organisationen übermäßig. Für den ISMS-Teilprozess Security Incident und Event Management (SIEM) wird als Lösungsvorschlag ein leichtgewichtiges Modell vorgestellt, das die Anforderungen aus etablierten Rahmenwerken berücksichtigt und relevante Prozessbausteine abbildet. Der leichtgewichtige SIEM-Prozess unterstützt somit eine ressourcenschonende Einführung eines ISMS. Basierend auf einer Anforderungsanalyse entsteht zunächst ein allgemeiner SIEM-Prozess, der als Grundlage für den leichtgewichtigen SIEM-Prozess dient. Dieser verzichtet durch Entfernen redundanter Prozessbausteine und sinnvolles Zusammenfassen auf jede vermeidbare Komplexität und halbiert die Anzahl der allgemeinen Prozessbausteine. Eine Experten-Evaluation validiert die grundsätzliche Anwendbarkeit des leichtgewichtigen SIEM-Prozesses.

Keywords: Security Incident und Event Management, SIEM, Information Security Management, IT Service Management

1 Einleitung

Informationssicherheit wird in der heutigen digitalen Gesellschaft immer wichtiger. Dies zeigt sich unter anderem durch die Berichterstattung in den Medien über Security Incidents, aber auch in dem zunehmenden Umfang gesetzlicher Verpflichtungen. Beispielsweise verlangt das im Juli 2015 beschlossene IT-Sicherheitsgesetz von Betreibern sogenannter kritischer Infrastrukturen, dass sie *angemessene organisatorische und technische Vorkehrungen zur Vermeidung von Störungen der Verfügbarkeit, Integrität, Authentizität und Vertraulichkeit ihrer informationstechnischen Systeme, Komponenten oder Prozesse zu treffen*[Bu15].

Entsprechend steigt auch für IT-Organisationen im Hochschulumfeld und in kleineren und mittleren Unternehmen (KMU) der Druck, ein sogenanntes Informationssicherheitsmanagementsystem (ISMS) zu etablieren – sei es nun eigenständig oder als Teil eines bereits bestehenden prozessorientierten Service-Managementsystems (SMS). Für die Gestaltung eines wirksamen Managementsystems für Informationssicherheit existieren verschiedene Rahmenwerke wie ISO/IEC 27000 oder IT-Grundschutz. Jedoch ist der Aufwand für

[1] Leibniz-Rechenzentrum, Boltzmannstr. 1, 85748 Garching b. München, {vorname}.{nachname}@lrz.de
[2] Ludwig-Maximilians-Universität München, Oettingenstr. 67, 80538 München, schaaf@nm.ifi.lmu.de

Einführung und Betrieb eines ISMS erheblich und strapaziert oft die Ressourcen kleinerer IT-Organisationen. Dazu kommen noch weitere erschwerende Faktoren. Viele Mitarbeiter nehmen die Einführung eines Managementsystems zunächst vor allem als Verlust von Entscheidungsfreiheit und Flexibilität wahr. Die Reaktionen sind entsprechend erst mal negativ [ET02], was die Umsetzung solch eines *organisatorischen Change*, z.B. nach dem Vorgehen von Kotter [KR06], aufwändig macht. Umso tiefgreifender dabei die Veränderung und umso schlechter sie vermittelt werden kann (oder wird), umso größer sind die zu erwartenden Widerstände.

Auch für einen Security Incident und Event Management (SIEM) Prozess, als wesentlicher Baustein des ISMS, ist somit eine einfache Anwendbarkeit und Verständlichkeit für alle Mitarbeiter ein entscheidender Erfolgsfaktor für eine erfolgreiche und nachhaltige Einführung. Es leiten hieraus sich die folgenden Fragestellungen ab:

- Wie sieht ein zu den wichtigsten bestehenden Rahmenwerken konformer SIEM-Prozess aus?
- Wie kann die Komplexität und die Anzahl der Prozessbausteine reduziert werden, um einen leichtgewichtigen SIEM-Prozess zu erhalten?
- Welche Prozessbausteine des SIEM sind obligatorisch und welche sind optional?
- Welche Prozessbausteine des SIEM lassen sich zusammenfassen?

Leichtgewichtige Modelle, d.h. einfach anwendbare und leicht verständliche Modelle zum Management der Informationssicherheit existieren kaum und beantworten die oben gestellten Fragestellungen nur in geringem Umfang.

Vorarbeiten zu dieser Veröffentlichung finden sich in der Masterarbeit „Ein Fachkonzept für leichtgewichtiges Informationssicherheits-Management“[Zi16].

In Abschnitt 2 werden etablierte Rahmenwerke beschrieben, aus deren Anforderungen in Abschnitt 3 zunächst ein allgemeiner SIEM-Prozess entsteht. Im darauffolgenden Abschnitt 4 wird darauf aufsetzend ein leichtgewichtiger SIEM-Prozess abgeleitet. Abschnitt 5 beschreibt die Durchführung einer Experten-Evaluation, deren Ergebnisse die Anwendbarkeit des leichtgewichtigen SIEM-Prozesses untermauern. Das Paper endet mit einer Zusammenfassung und einem Ausblick.

2 Betrachtete Rahmenwerke

Die **ISO/IEC 27000** [IS13b] ist eine Standardfamilie zum Informationssicherheits-Management, die die Möglichkeit zur Zertifizierung bietet. Innerhalb dieser Standardfamilie beschreibt ISO/IEC 27001 [IS13b] die Anforderungen an ein ISMS sowie, im Anhang A auch Maßnahmen, die für die Etablierung eines SIEM relevant sind. SIEM ist auch Gegenstand im **Information Security Management Toolkit** [UC 1] der Universities and Colleges Information Systems Association (UCISA), welches auf der ISO/IEC 27001 [IS13b] und 27002 [IS13a] aufsetzt. Ein weiteres Rahmenwerk zum Informationssicherheitsmanagement ist der durch das Bundesamt für Sicherheit in der Informationstechnik (BSI) beschriebene **IT-Grundschutz** [Bu16]. Dieser ist ebenfalls kompatibel mit

der ISO/IEC 27000 und besteht aus einem vierteiligen Standard sowie modular aufgebauten IT-Grundschutz-Katalogen mit bereits identifizierten Bausteinen, Maßnahmen sowie Gefährdungen.

Die **IT Infrastructure Library (ITIL)** [Ax11] ist demgegenüber eine Büchersammlung mit "Good Practices" zu einem SMS und gilt momentan als der De-facto Standard im IT-Service Management (ITSM) [Br11]. Der Fokus von ITIL liegt auf 25 Prozessen, die einem Service Lifecycle zugeordnet sind, darunter auch das Informationssicherheitsmanagement. Eine Standardfamilie für ein leichtgewichtiges SMS ist durch **FitSM** [Fi16] beschrieben, die aus sieben Teilen besteht. Insbesondere der Teil FitSM-2 [Fi22a] legt Ziele und Aktivitäten, u. a. zum Informationssicherheits-Management, fest.

Die mehrteilige Dokumentenfamilie **COBIT 5** [IS12b] rückt die Governance und das Management der Unternehmens-IT in den Vordergrund. Die Anforderungen zum SIEM sind im Handbuch COBIT 5: Enabling Processes [IS12a] innerhalb der Domäne der Management-Prozesse beschrieben.

Wesentlich für alle Rahmenwerke ist das Prinzip des Deming-Zyklus [De86] (Plan-Do-Check-Act, kurz PDCA) zur kontinuierlichen Verbesserung eines Managementsystems.

3 Harmonisierung der SIEM Prozessbausteine

Zur Sicherstellung der Kompatibilität mit etablierten Rahmenwerken dienen die Anforderungen der sechs vorgestellten Rahmenwerke als Grundlage für die Entwicklung eines leichtgewichtigen SIEM-Prozesses. Hierzu werden die Prozessbausteine zunächst in die grundlegenden Typen Aktivitäten und Outputs gruppiert. [3]

Diese Vorgehensweise ermöglicht in einem späteren Schritt eine Vergleichbarkeit der Prozessbausteine und kann auf beliebige Prozesse oder Themengebiete angewendet werden.

Nach Konsolidierung und Harmonisierung bedeutungsgleicher Begriffe ergeben sich, ausgehend von den Anforderungen der sechs vorgestellten Rahmenwerke, wie in Tabelle 1 ersichtlich, die folgenden Aktivitäten und Outputs:

3.1 Aktivitäten

A1 Entwickeln eines Incident Response Plans, Definieren von Eskalations- und Kommunikationswegen
A2 Überwachen und Aufzeichnen von Security Events
A3 Bewerten und Klassifizieren von Security Events
A4 Analysieren und Antworten auf Security Incidents und Events
A5 Definieren und Überwachen von Folgemaßnahmen
A6 Beseitigen der Ursachen, Untersuchen und Lindern der Konsequenzen
A7 Aufzeichnen aller Aktionen und Berichterstattung

[3] Inputs sind bereits in den Aktivitäten enthalten

A8 Durchführen eines Reviews nach einem Security Incident
A9 Durchführen von Tests zur Simulation von Security Incidents und deren Dokumentation

3.2 Outputs

O1 Dokumentierter Incident Response Plan zum Umgang mit Security Incidents
O2 Definierte Eskalations- und Kommunikationswege
O3 Bewertungs- und Entscheidungskriterien
O4 Berichte und Aufzeichnungen über Security Incidents und Events sowie Folgemaßnahmen
O5 Richtlinie für das Management von Security Incidents (oder vergleichbare beschreibende Dokumentation)

Tabelle 1: Gegenüberstellung der Rahmenwerke

	ISO/IEC 27001	**UCISA Toolkit**	**IT-Grundschutz**	**ITIL**	**FitSM**	**COBIT 5**
Aktivitäten						
A1		✓	✓			✓
A2	✓	✓	✓	✓	✓	(✓)
A3	✓	✓	(✓)		✓	✓
A4	✓	✓	✓	✓	✓	✓
A5			✓	✓	✓	
A6	(✓)	✓	✓	✓		✓
A7	(✓)	✓	(✓)	✓	(✓)	✓
A8	✓	✓	(✓)	(✓)	(✓)	(✓)
A9			✓			
Outputs						
O1		✓				✓
O2		✓	✓			✓
O3	(✓)	(✓)			(✓)	
O4	✓	✓	✓	✓	✓	✓
O5	✓	✓	✓	✓	✓	✓

Rahmenwerk enthält Aktivität bzw. Output ...
✓: ... explizit
(✓:) ... implizit

Aus den beschriebenen Aktivitäten lässt sich nun ebenfalls ein allgemeiner SIEM-Prozess ableiten (vgl. Abbildung 1).

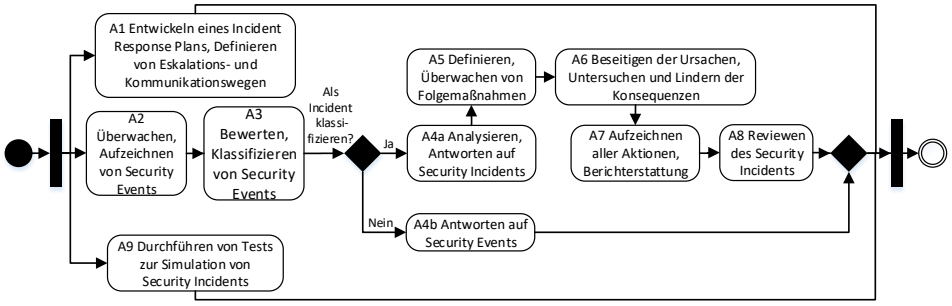


Abbildung 1: allgemeiner SIEM-Prozess

4 Der leichtgewichtige SIEM-Prozess

Zur Gestaltung eines einfach anwendbaren und leicht verständlichen Modells werden die in Abschnitt 3 vorgestellten Prozessbausteine (vgl. Tabelle 2) überprüft. Wo dies möglich ist ohne die Wirksamkeit des SIEM-Prozesses entscheidend zu gefährden, werden Bausteine zusammengefasst oder, in Einzelfällen, auch nicht essentielle Bausteine komplett gestrichen.

So lassen sich die Artefakte O1, O2 und O3 in einem leichtgewichtigen Modell in der Richtlinie (O5) (und der damit einhergehenden Prozessbeschreibung) für alle Security Incidents und Events auf einem gemeinsamen Niveau festschreiben. Denkbar sind hier ebenfalls verschiedene Verfahren für unterschiedliche Typen von Incidents bzw. Events, sodass je nach Klassifizierung in Aktivität A3 unterschieden werden kann. Jedoch sollte bei dieser Variante die Anzahl und Variabilität der Verfahren auf ein Minimum beschränkt werden. Somit werden beispielsweise Response Pläne, Eskalations- und Kommunikationswege, Bewertungs- und Entscheidungskriterien nur noch nach Typ des Incidents bzw. Events definiert. Benötigte Abweichungen werden an den Information Security Risk Manager eskaliert.

Mit ähnlicher Argumentation lässt sich die Aktivität A1 ebenfalls in die Richtlinie aufnehmen.

Weiter vereinfachend wirkt eine Zusammenfassung der Aktivitäten A5 und A6, da die Definition der Folgemaßnahmen und die damit einhergehende Durchführung und Beseitigung der Störung in kleineren und mittleren Organisation meist in einem Arbeitsschritt erledigt wird.

Verzichten kann man hingegen auf die Aktionen A7 und A8. Relevante Aufzeichnungen werden aufgrund der hohen Verbreitung von SMS-Tools meist ohnehin automatisch erstellt. Ein entsprechendes Erfolgs-/Nicht-Erfolgs-Review wird üblicherweise im Rahmen der Aktivitäten A5 und A6 implizit durchgeführt. Auch die Aktion A9 ist vernachlässigbar, da die Simulation von Security Incidents bei KMUs meist nicht organisationsintern sondern von entsprechend spezialisierten Unternehmen extern durchgeführt wird. Als Koordi-

Tabelle 2: Vergleich der Rahmenwerke mit dem leichtgewichtigen Modell

	ISO/IEC 27001	UCISA Toolkit	IT-Grundschutz	ITIL	FitSM	COBIT 5	leichtgewichtiges Modell
Aktivitäten							
A1		✓	✓			✓	R
A2	✓	✓	✓	✓	✓	(✓)	✓
A3	✓	✓	(✓)		✓	✓	✓
A4	✓	✓	✓	✓	✓	✓	✓
A5			✓	✓	✓		✓
A6	(✓)	✓	✓	✓		✓	
A7	(✓)	✓	(✓)	✓	(✓)	✓	(✓)
A8	✓	✓	(✓)	(✓)	(✓)	(✓)	(✓)
A9			✓				
Outputs							
O1		✓				✓	R
O2		✓	✓			✓	R
O3	(✓)	(✓)			(✓)		R
O4	✓	✓	✓	✓	✓	✓	✓
O5	✓	✓	✓	✓	✓	✓	✓

Rahmenwerk enthält Aktivität bzw. Output ...
✓: ... explizit | (✓:) ... implizit
R: als Verfahren in Richtlinie enthalten

nator dient hier ebenfalls der Information Security Risk Manager. Somit kann A9 ebenfalls als Eskalationszustand betrachtet werden.

Es ergibt sich somit der leichtgewichtige SIEM-Prozess aus Abbildung 2.

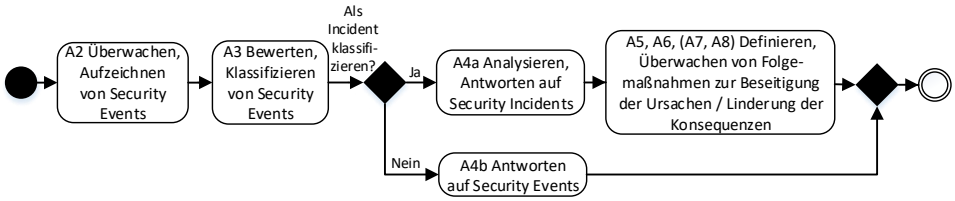


Abbildung 2: leichtgewichtiger SIEM-Prozess

Eine entsprechende RACI-Matrix (siehe Tabelle 3) bildet das zugehörige Mapping der Verantwortlichkeiten zwischen den jeweiligen Aktivitäten und den Rollen ab. Die Rollen

dazu entstammen aus dem FitSM-3 Rollenmodell [Fi22b]. Hierbei stellt der Prozessmanager ISM sicher, dass Security Incidents und Events effektiv erkannt, dokumentiert, klassifiziert und bearbeitet werden. Zusätzlich ist pro Asset bzw. Control ein Asset Owner bzw. Information Security Control Owner notwendig, dem die Verantwortung des jeweiligen Objektes (Asset/Control) obliegt.

Die im linken Teil der Tabelle dargestellten Outputs sind allen Aktivitäten zugeordnet und sind in diesem Rahmen zu erstellen.

Tabelle 3: Prozessbausteine des Security Incident und Event Managements

Security Incident und Event Management Prozess				
		Rollen		
Artefakte & Outputs	*Aktivitäten & Abläufe*	Prozessmanager ISM	Asset Owner	Information Security Control Owner
• O4, (O1, O2, O3, A1) Aufzeichnungen und Berichte über Security Events und Incidents sowie Folgemaßnahmen • O5 Richtlinie für das Management von Security Incidents	A2 Überwachen und Aufzeichnen von Security Events	A	I	R
	A3 Bewerten und Klassifizieren von Security Events	A	C	R
	A4a Analysieren und Antworten auf Security Incidents	A	C	R
	A4b Antworten auf Security Events	A / R		
	A5, A6, (A7,A8) Definieren und Überwachen von Folgemaßnahmen zur Beseitigung der Ursachen / Linderung der Konsequenzen	A	C	R

R = Responsible (Durchführungsverantwortlich)
A = Accountable (Verantwortlich im Sinne von Kostenverantwortung und Rechenschaftspflicht)
C = Consulted (wird um Rat gefragt)
I = Informed (wird informiert)

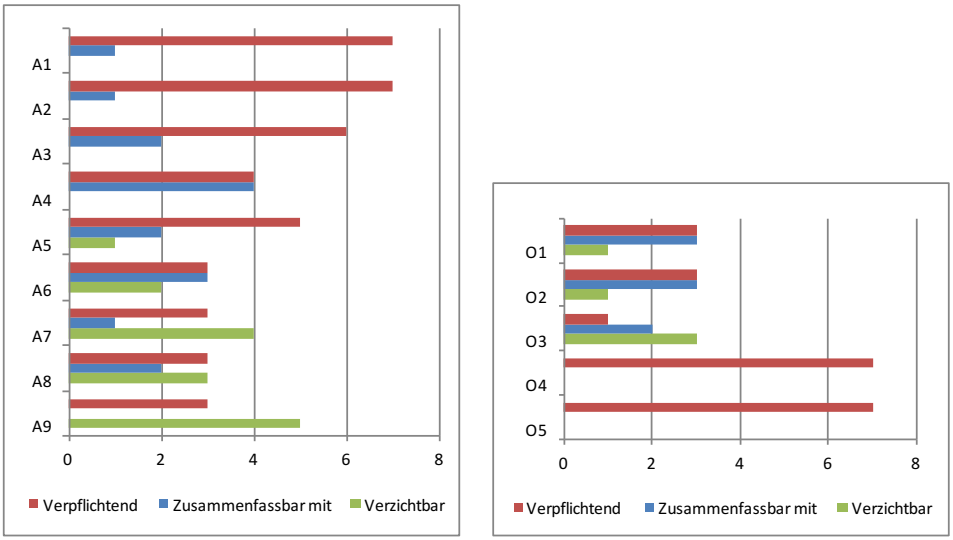
5 Experten-Evaluation

Zur Validierung des Modells beurteilten acht Experten aus Hochschulumfeld und Industrie (z. B. Berater/Trainer, Auditoren, Datenschutzbeauftragte) die Kritikalität und Rele-

vanz der einzelnen Prozessaktivitäten und -outputs. Durchgeführt wurde die Experten-Evaluation in Form eines Online-Fragebogens. Die unvoreingenommene Fragestellung (ohne Einsicht in das vorgestellte leichtgewichtige Modell) diente dazu, ein fundiertes Feedback über die Relevanz der Prozessbausteine zu erhalten.

Dabei bewerteten die Experten die Aktivitäten (A1 bis A9) und Outputs (O1 bis O4) aus Abschnitt 3 und klassifizierten diese entsprechend den Kategorien „Verpflichtend“, „Zusammenfassbar mit“oder „Verzichtbar“.

Abbildung 3 zeigt die Ergebnisse der Evaluation. Die X-Achse repräsentiert die Anzahl der Experten, die Y-Achse die entsprechenden Prozessbausteine.



(a) Aktivitäten

(b) Outputs [4]

Abbildung 3: Ergebnisse der Experten-Evaluation [5]

Abbildung 3a verdeutlicht, dass die Aktivitäten A1 bis A3 eine essentielle Rolle in einem leichtgewichtigen SIEM-Prozess spielen, wohingegen die Aktivitäten A7 bis A9 überwiegend als verzichtbar gelten. Die Tendenz zur Zusammenfassbarkeit sahen die Experten jeweils in den Aktivitäten A4 bis A6. Einer der befragten Experten begründete dies ebenfalls damit, dass die Aktivitäten A7 und A8 bereits implizit in A5 und A6 enthalten seien (vgl. Abschnitt 4).

Hinsichtlich der Outputs repräsentieren die Ergebnisse in Abbildung 3b eindeutig die Notwendigkeit des Outputs O4 in einem leichtgewichtigen SIEM-Prozess. Entsprechend unserer Einschätzung stuften die Experten die Outputs O1 bis O3 als zusammenfassbar ein.

[4] Bei Output O5 wurden die Experten zur Relevanz (sehr relevant, teilweise relevant, weniger relevant, gar nicht relevant) befragt. Fünf Experten halten O5 für sehr relevant. Zwei Experten halten O5 für teilweise relevant.

[5] Die Antwortoption „keine Angabe“ist in der Abbildung nicht aufgeführt.

In der Gesamtbetrachtung aller Prozessbausteine bestätigen die Experten den leichtgewichtigen SIEM-Prozess.

6 Zusammenfassung und Ausblick

Die Etablierung und der Betrieb eines ISMS für IT-Organisationen im Hochschulumfeld aber auch in kleineren und mittleren Unternehmen bedeutet für die betroffenen Unternehmen oft eine zeitaufwendige Einführung und hoher Ressourcenbedarf beim Betrieb. Bei der Betrachtung der Komplexität des dafür erforderlichen SIEM-Prozesses wird der Aufwand besonders deutlich. Eine Harmonisierung der SIEM-Prozessbausteine der in Abschnitt 2 vorgestellten Rahmenwerke und die folgende Extraktion eines allgemeinen SIEM-Prozesses (vgl. Abschnitt 3) verdeutlicht den entstehenden Aufwand.

Viele Prozessbausteine dieses zu Vorgaben mehrerer Rahmenwerke konformen Prozesses sind zusammenfassbar oder gar redundant (vgl. Abschnitt 4). Mittels sinnvollem Zusammenfassen und Entfernen lässt sich die Anzahl der Prozessbausteine des Prozesses reduzieren, ohne dass Funktion oder Wirksamkeit des Prozesses wesentlich eingeschränkt werden. Insbesondere die Verankerung von wenigen, spezifischen, kurzgefassten Verfahren in der SIEM-Richtlinie reduziert die allgemeine Prozesskomplexität.

Der Vorteil des leichtgewichtigen SIEM-Prozesses ist neben der kompakten Darstellung zusätzlich eine Vereinfachung der Anwendbarkeit im Hochschulumfeld und für kleine bis mittelgroße Organisationen. Mithilfe der Tabelle 3 sind die erforderlichen Prozessbausteine übersichtlich aufgelistet. Anhand des Mappings der Verantwortlichkeiten zwischen Rollen und Aktivitäten ist die Aufgabenverteilung leicht verständlich. Ein UML-Aktivitätsdiagramm unterstützt das Verständnis des Zusammenhangs und der Einordnung der Prozessbausteine. Der entstandene leichtgewichtige SIEM-Prozess ist dabei weiterhin kompatibel mit den etablierten Rahmenwerken, auf denen er ursprünglich basiert.

Ausblickend ist zur weiteren Bewertung des Modells im nächsten Schritt die praktische Umsetzung des leichtgewichtigen SIEM-Prozesses geplant.

Ebenso können analog zu dem hier beschriebenen Vorgehen weitere hilfreiche Artefakte, wie beispielhafte, leichtgewichtige Verfahrensbeschreibungen, zur Unterstützung der Umsetzung eines leichtgewichtigen Service Management Ansatzes (vgl. FitSM [Fi16]) entwickelt werden.

Danksagung

Die Autoren danken den Mitgliedern des MNM-Teams für hilfreiche Diskussionen und wertvolle Kommentare zu vorhergehenden Versionen des Papers. Das MNM-Team, unter der Leitung von Prof. Dr. Dieter Kranzlmüller und Prof. Dr. (em.) Heinz-Gerd Hegering, ist eine Forschungsgruppe mit Wissenschaftlern an den Münchner Universitäten, der Universität der Bundeswehr München und dem Leibniz-Rechenzentrum der Bayrischen Aka-

demie der Wissenschaften. Der Internetauftritt findet sich unter http://www.mnm-team.org.

Literaturverzeichnis

[Ax11] Axelos, Hrsg. ITIL service design. TSO The Stationery Office, London, 2nd ed.. Auflage, 2011.

[Br11] Brenner, Michael; Gentschen Felde, Nils; Hommel, Wolfgang; Metzger, Stefan; Reiser, Helmut; Schaaf, Thomas: Praxisbuch ISO-IEC 27001: Management der Informationssicherheit und Vorbereitung auf die Zertifizierung ; [mit 80 Prüfungsfragen zur Vorbereitung auf die Foundation-Zertifizierung]. Hanser, München, 2011.

[Bu15] Bundestag: , Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme (IT-Sicherheitsgesetz), 17.07.2015.

[Bu16] Bundesamt für Sicherheit in der Informationstechnik: , IT-Grundschutz Webseite. https://www.bsi.bund.de/DE/Themen/ITGrundschutz/itgrundschutz_node.html, Version: 2016. Abruf: 24.Mrz.2016.

[De86] Deming, William Edwards: Out of the Crisis. Massachusetts Institute of Technology Center for Advances Engineering Study, Cambridge, Massachusetts, 1986.

[ET02] Elrod, P. David; Tippett, Donald D.: The “death valley” of change. Journal of Organizational Change Management, 15(3):273–291, 2002.

[Fi16] FitSM: , FitSM - Standards for lightweight IT service management. http://fitsm.itemo.org/, Version: 2016. Abruf: 1.Apr.2016.

[Fi22a] FitSM: , FitSM-2 Objectives and activities. http://fitsm.itemo.org/sites/default/files/FitSM-2_Objectives_and_activities.pdf, Version: 2.2. Abruf: 15.Mrz.2016.

[Fi22b] FitSM: , FitSM-3 Role model. http://fitsm.itemo.org/sites/default/files/FitSM-3_Role_model.pdf, Version: 2.2. Abruf: 15.Mrz.2016.

[IS12a] ISACA: COBIT 5 - Enabling Processes. ISACA, Illinois, 2012.

[IS12b] ISACA: COBIT 5 - Rahmenwerk für Governance und Management der Unternehmens-IT. ISACA, Illinois, 2012.

[IS13a] ISO/IEC: Information technology - Security techniques - Code of practice for information security controls (ISO/IEC 27002:2013). 2013.

[IS13b] ISO/IEC: Information technology - Security techniques - Information security management systems - Requirements (ISO/IEC 27001:2013). 2013.

[KR06] Kotter, John P.; Rathgeber, Holger: Das Pinguin-Prinzip: Wie Veränderung zum Erfolg führt. Droemer, München, 2006.

[UC 1] UCISA: , UCISA Information Security Management Toolkit. https://www.ucisa.ac.uk/~/media/Files/members/activities/ismt/Complete%20with%20covers, Edition 1.0 Volume 1. Abruf: 20.Feb.2016.

[Zi16] Ziegler, Jule Anna: Ein Fachkonzept für leichtgewichtiges Informationssicherheits-Management. Masterarbeit, Ludwig-Maximilians-Universität München, 2016.

eInfrastrukturen für Forschung, Lehre und Forschungsdatenmanagement

Design eines FDM-fähigen Speichersystems

Dennis Wehrle,[1] Bernd Wiebelt,[2] Dirk von Suchodoletz[2]

Abstract: Es liegt in der Natur wissenschaftlicher Prozesse, dass viele Zwischenergebnisse, aber auch schlicht irrelevante Forschungsdaten gespeichert werden, die bei regelmäßigen Überprüfungen eigentlich gelöscht werden könnten. Weiterhin passiert es häufig, dass potentiell wertvolle Daten aufgrund von Platzmangel unwiederbringlich gelöscht werden. Ein nachhaltiges Forschungsdatenmanagement schafft den Spagat zwischen der Finanzierbarkeit einer wachsenden Datenmenge bei gleichzeitiger Optimierung der Qualität der Daten. Dieser Beitrag diskutiert, wie klassische technische Lösungen durch geeignete mit den einzelnen Wissenschafts-Communities abgestimmte Steuerungsrahmen ergänzt werden können. So ließe sich das hier beschriebene Speicherkonzept als FDM-Baustein im Nutzen verbessern, indem die Forschenden zu jedem Zeitpunkt qualifizierende Beschreibungen ihrer Daten hinzufügen, wobei die Angabe derselben eine Grundvoraussetzung für eine langfristige Speicherung darstellen. Für einen echten, den Rahmen der einzelnen Forschungsinstitution übergreifenden, Mehrwert können diese Metadaten über standardisierte Schnittstellen abgefragt und in bestehende und zu entwickelnde fachspezifische Workflows integriert werden. Ein FDM-fähiges Speichersystem muss berücksichtigen, dass Daten vieler Forschungsgruppen an verteilten Standorten liegen und von unterschiedlichen Wissenschafts-Communities verwendet werden.

Keywords: Forschungsdatenmanagement, Speichersystem, Storage, Governance, Data Life Cycle

1 Motivation

Sowohl die seit Jahrzehnten exponentiell steigende Verfügbarkeit von Rechenleistung, als auch die parallel dazu gewachsene Leistungsfähigkeit der Geräte zur wissenschaftlichen Datenerfassung hat zu einer Explosion der vorzuhaltenden Daten geführt. Mit gleichzeitig steigender Plattenkapazität wurden vielfach ad-hoc Lösungen etabliert, um für Arbeitsgruppen gemeinsamen Speicherplatz in der einfachst möglichen Form, als gemeinsam sichtbares Dateisystem, zur Verfügung zu stellen. Diese an sich vernünftige Vorgehensweise stößt inzwischen dort an ihre Grenzen, wo Kernleistungen der Wissenschaft tangiert sind, nämlich die erzielten Ergebnisse für andere Wissenschaftler nachvollziehbar und nachnutzbar zu machen (Review-Prozess) und für zukünftige Generationen zu konservieren (Archivierungsprozess). Das Thema Forschungsdatenmanagement (FDM) beschäftigt daher Wissenschaftler, Forschungseinrichtungen, Fördergeber und Politik seit geraumer Zeit. So hat die DFG bereits 1998 in ihrer Denkschrift „*Vorschläge zur Sicherung guter wissenschaftlicher Praxis*“ unter anderem Empfehlungen zur Sicherung und Aufbewahrung von Primärdaten veröffentlicht. Deshalb wird in zunehmendem Maße ein (institutionalisiertes) FDM zur Voraussetzung für die Bewilligung von Zuwendungen für neue

[1] Universität Freiburg, Professur für Kommunikationssysteme, Hermann-Herder-Str. 10, 79104 Freiburg, dennis.wehrle@rz.uni-freiburg.de

[2] Universität Freiburg, Rechenzentrum, Hermann-Herder-Str. 10, 79104 Freiburg, bernd.wiebelt@rz.uni-freiburg.de / dirk.von.suchodoletz@rz.uni-freiburg.de

Forschungsvorhaben. [3] Das IT-gestützte nachhaltige FDM [Ne12] [BHM11] rückt damit in den Fokus zukunftsorientierter Forschungsprozesse. [4] FDM muss sowohl starke organisatorische als auch technische Aspekte vereinen, die sich zudem durch unterschiedliche Zeiträume auszeichnen. Die mit einem wirkungsvollen und nachhaltigen FDM verbundenen Aufwendungen können kaum sinnvoll von einzelnen Forschungsgruppen oder Instituten geleistet werden, sondern sollten durch die Forschungseinrichtungen und ihre zentralen Einrichtungen wie Rechenzentren und Bibliotheken übergreifend koordiniert werden.

Nach einer Phase der Konzept- und Strategieentwicklung haben Wissenschaftseinrichtungen erste Richtlinien erlassen und damit begonnen, Repositorien für Forschungsdaten einzurichten, welche zunehmend in Verbünde integriert werden und einen weltweiten Nachweis ihrer enthaltenen Daten erlauben [EU17]. Forschungsdaten auf nachhaltige Weise zu verwalten und in standardisierte Arbeitsabläufe zu integrieren erzeugt organisatorische Herausforderungen auf mehreren Ebenen. Zum einen sind die Wissenschafts-Communities selbst gefordert, ihre Prozesse entsprechend vorzubereiten und anzupassen. Zum anderen sind die Forschungsinstitution in der Pflicht, in Abstimmung mit den Forschenden passende Soft- und Hardware-Lösungen unter Finanzierungs- und Kosten-Nutzen-Abwägungen zu finden. Wegen der Verschiedenartigkeit der Anforderungen in den einzelnen Disziplinen [5] und schon bestehenden einrichtungs- und fächerübergreifenden Kooperationen lässt sich die Herausforderung effektiv nur in Zusammenarbeit realisieren. Bei einer Kooperation mehrerer Beteiligter auf Ebene der Forschungseinrichtung und in föderierten Verbünden müssen die Interessen der einzelnen Akteure im Gesamtsystem geeignet gegeneinander abgewogen werden. [6] Es werden Vorschläge aus Sicht des Rechenzentrums gemacht, wie ein Konzept- und Speichersystem für ein FDM an einer konkreten Forschungseinrichtung – hier der Universität Freiburg – mit Einbindung in kooperative und föderative Strukturen umgesetzt werden könnte. Diskutiert werden organisatorische und technische Fragen, insbesondere wie die Anreicherung mit fachspezifischen Metadaten gemeinsam mit den Communities realisiert und gesteuert werden kann. Die technische Umsetzung basierend auf einer Hierarchical Storage Management Architektur sollte Nutzern verschiedene Zugangswege zum Speichersystem erlauben und gleichzeitig standardisierte Schnittstellen für Replikation und Austausch der Daten und Metadaten anbieten.

2 Vorüberlegungen in Freiburg

Der Blick auf die Aktivitäten an anderen Forschungseinrichtungen und Gespräche mit einzelnen Forschungs-Communities an der Universität Freiburg ergibt ein breit gefächertes Bild an Anforderungen. Diese beinhalten den kurzfristigen Bedarf an Speicher im zwei bis dreistelligen Terabyte-Bereich mit file- oder objektbasierter Anbindung ebenso wie den Zugriff mittels Versionierungssystemen oder die Ablage in einem Repositorium. Einige

[3] Leitfaden der DFG für die Antragstellung: Projektanträge, `http://www.dfg.de/formulare/54_01/54_01_de.pdf`, S. 5

[4] Vgl. Grundsätze zum Umgang mit Forschungsdaten. Allianz der dt. Wissenschaftsorganisationen. `http://www.allianzinitiative.de/de/handlungsfelder/forschungsdaten/grundsaetze` (2010)

[5] Ergebnisse einer qualitativen Befragung im Rahmen des bwFDM-Communities Projekts [Tr15].

[6] Vgl. „Überlegungen zu Steuerung und Governance von kooperativ betriebenen HPC-Infrastrukturen“, [vo16]

Communities können für ihre Datenerhebung auf bereits etablierte Forschungsinfrastrukturen zurückgreifen, benötigen jedoch Platz für eine langfristige Archivierung von Roh- oder Ergebnisdaten beispielsweise aus abgeschlossenen Promotionen. Im Bereich der Bioinformatik steigt der Bedarf allein durch immer höher auflösende Messinstrumente. Das Nutzungsprofil vieler Disziplinen lässt sich gut durch das Domain Model annähern. [7] Eine reine Speicherung von Forschungsdaten ist allerdings nicht ausreichend. Für ein effektives FDM-System werden zusätzliche Komponenten benötigt, wie in Abschnitt 3 erläutert.

Das Konzept für ein FDM bewegt sich im Spannungsfeld zwischen sehr vielfältigen und fachspezifischen Bedürfnissen der Wissenschafts-Communities einerseits und der praktischen Umsetzbarkeit einer technischen und organisatorischen Lösung andererseits. Um die Gefahr einer Insellösung zu vermeiden, ist auf einen möglichst standardisierten Ansatz zu achten und eine Umsetzung in föderativen Strukturen zu gewährleisten. [8] Ein implementiertes FDM einer Einrichtung muss im Einzelnen folgende Aspekte adressieren:
Forschungsdaten fallen durch die Vielfalt der Wissenschaftsdisziplinen in verschiedenen Formen an und sind vielfältiger Herkunft. Jede Disziplin hat eigene Vorstellungen ihrer beschreibenden Metadaten. Ein FDM deckt optimalerweise den gesamten Data-Lifecycle [Ba12] ab, von der Erhebung oder Erzeugung der Daten über die einzelnen Verarbeitungsschritte bis zu ihrer leichten Auffindbarkeit und langfristigen Bereitstellung für eine Nachnutzung durch Dritte.
Wissenschafts-Communities betreiben FDM auf unterschiedliche Weise [Kl13]. Viele Forschungsdaten laufen noch nicht in nationalen oder internationalen Datenzentren zusammen. Die Bedürfnisse unterscheiden sich durch die teilweise vor Ort oder weltweit verteilt schon vorhandenen Ressourcen, um Daten nach der Erzeugung oder Publikation geeignet abzulegen. Solche Strukturen sollen in Freiburg nicht dupliziert, sondern passend für die Forschenden vor Ort ergänzt werden und geeignete Schnittstellen für die Einbindung in übergeordnete Strukturen bieten.
Forschungseinrichtungen wie die Universität Freiburg müssen verbindliche Richtlinien erlassen und deren Umsetzung fördern. Hierzu zählen die Frage der Freigabe der Daten und Regelungen vergleichbar zu Publikationsverträgen mit der Universitätsbibliothek. Das Rechenzentrum der Universität hat sich aus diesen Gründen an Projekten wie bwFDM-Communities [Tr15] und „Landesweit koordinierte Strukturen für Nachweis und effiziente Nachnutzung von Forschungsdaten“ beteiligt, um schrittweise eine Policy für die Gesamteinrichtung zu entwickeln. Den Empfehlungen des RfII [Rf16] folgend liegt der Fokus auf der Einbindung in einrichtungsübergreifende Strukturen. Als Vorbilder [Er13] für eine konkrete Umsetzung dienen beispielsweise die Aktivitäten an der HU Berlin, der Universität Göttingen oder der RWTH Aachen [EMS].
Dienstanbieter müssen in die Lage versetzt werden, sowohl die kurz- und längerfristige Speicherung als auch den Nachweis der Forschungsdaten anzubieten. Da diese Aufgabe wegen der Vielfalt der Anforderungen und Disziplinen nicht von einer Einrichtung geleistet werden kann, sollte eine zuverlässige, verteilte Datenspeicherung in föderierten

[7] Daten entstehen oft durch Aktivitäten einzelner Forschender oder Gruppen (private Domäne), die in weiteren Schritten in Kooperationen geteilt (Gruppendomäne) und später je nach Art in die dauerhafte Domäne verschoben werden, wo eine Nachnutzung möglich wird (Zugriffsdomäne), vgl. [Kl13], S. 6

[8] Vgl. hierzu „Rahmenbedingungen einer disziplin-übergreifenden Forschungsdaten-Infrastruktur“, `http://www.forschungsdaten.org/index.php/Radieschen`

Verbünden erfolgen. [Kl13] [EU17]
Nachhaltige Finanzierung und Ausstattung des FDM sind wegen der erwartbaren erheblichen Zunahme der Datenmengen im Petabyte-Bereich pro Jahr eine nicht unerhebliche Herausforderung. Das FDM wird ebenso wie eine Bibliothek zu einer zentralen Infrastruktur (nicht nur am eigenen Standort) mit entsprechendem Finanzierungsbedarf [Rf16]. Da ein nicht unerheblicher Teil zukünftiger Kosten von der Datenmenge abhängt, sind Möglichkeiten zur Beteiligung der Nutzer insbesondere bei erheblichen Datenmengen vorzusehen. [9] Umso wichtiger werden geeignete Verfahren zur Qualifizierung der langfristig gespeicherten Daten, die ein ausgewogenes Verhältnis zwischen Quantität und Qualität der Forschungsdaten erzielen.

3 Vorschlag für eine FDM-Infrastruktur

Die für die Universität in der Diskussion befindliche FDM-Infrastruktur vereint eine Kombination aus Hard-, Software- und organisatorischen Bausteinen. Die bedarfsorientierte Realisierung orientiert sich gleichzeitig an den Lösungen von RADAR und am Service-Portfolio von EUDAT [10] sowie an aktuellen technologischen Umsetzungen der Storage-Hersteller für Massendaten. Das RZ koordiniert seine Aktivitäten mit Partnern in Baden-Württemberg. Für das Storage-System wird ein dreistufiges Konzept angestrebt, welches verschiedene Stadien des Data-Life-Cycle abdecken soll:

1. Layer I: Die oberste Ebene bietet verschiedene High-Level-Access-Varianten für Forschende an, über die sie direkt mit dem Storage-System interagieren können. Diese können lokale Filesystem-Caches an entfernten Standorten enthalten.
2. Layer II: Die mittlere Ebene kümmert sich um die primäre Datenhaltung für die aktuell im Zugriff befindlichen Daten.
3. Layer III: Die unterste Ebene übernimmt längerfristige Archivfunktionen. Hier werden Daten abgelegt, die nicht mehr aktiv bearbeitet werden und für bestimmte länger- und langfristige Zeiträume aufbewahrt werden sollen.

Die technische Umsetzung kann in einem hierarchischen Storage-Modell erfolgen, in dem die Daten je nach erwarteter Verfügbarkeit und Redundanzlevel abgelegt werden. Dieses wird durch einen Data-Mover orchestriert, der zusätzliche Informationen für seine Aktivitäten berücksichtigt, die aus den von den Forschenden gepflegten Metadaten gewonnen werden.

Layer I bietet verschiedene Dienste an, die von einem direkten Filesystem-Zugriff, über Versionierungsdienste wie beispielsweise GIT bis hin zum Repository- oder Object Store Zugriff reichen. Die Forschenden erhalten nach Beantragung Zugriff auf den Speicherplatz für einen gewissen Zeitraum. Für jedes Vorhaben wird hierzu ein (virtueller) Datencontainer im FDM-Storage-System erzeugt. Eine längerfristige Datenhaltung wird von

[9] Vgl. Vorteile der Nutzer in der Beteiligung und bestehenden Forschungsinfrastrukturen, Aufwuchsfinanzierung und andere Beteiligungsmodelle in [vo16].
[10] Vgl. `https://www.radar-projekt.org` bzw. in [EU17] Services und Support

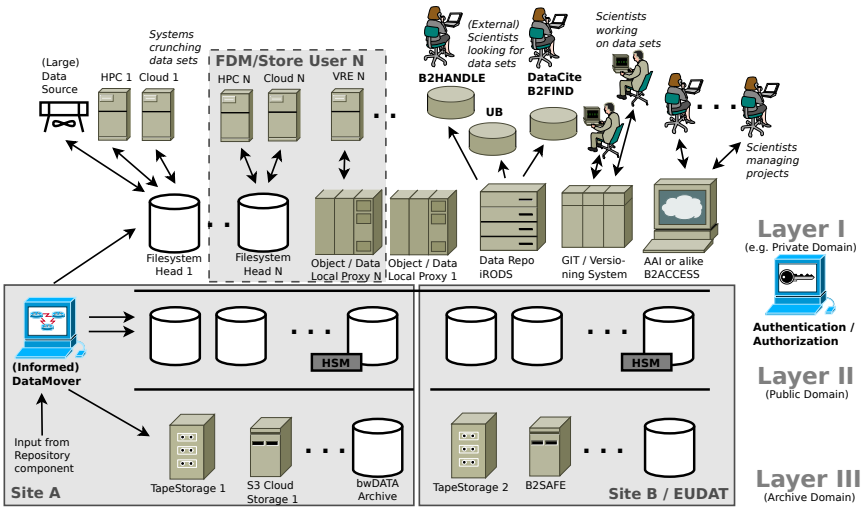


Abb. 1: Überlegungen zur FDM-Infrastruktur und Schnittstellen zu übergeordneten Systemen

der Qualifizierung der Daten und der Scientific Governance abhängig gemacht. Die technische Zugriffsebene besteht sowohl aus Hardwarekomponenten des FDM-Systems als auch aus einer Softwareschicht, die längerfristig angelegt ist und abstrakt auf der Hardwareschicht aufsetzt. Hierzu zählen die Bereitstellung von Versionierungsdiensten ebenso wie Repository- oder Object Store Komponenten. Hinzu kommt das Benutzerinterface (Beantragung, Metadaten) und Schnittstellen zu Management- sowie Authentifizierungskomponenten. Bausteine wie Versionierungssysteme oder Repositorien existieren bereits am Markt oder sind bereits in den beteiligten Einrichtungen im Einsatz. Sie sind jedoch geeignet in das Gesamtsystem zu integrieren.

Layer II – die mittlere Schicht – bildet die zentrale Komponente des FDM-Systems. Sie wird hauptsächlich durch die Hardwarekonzepte des zu beauftragenden Anbieters bestimmt. Sie bringt einen Data-Mover mit, der nach entsprechenden Kriterien die Ablage der Daten-Container steuert. Dieser könnte unterschiedliche Zahlen von Kopien ebenso wie die Lokalität der Daten berücksichtigen. Diese Schicht kümmert sich um die Integrität der Daten durch verschiedene technische Umsetzungen, wie beispielsweise Erasure Coding, RAID-Verbünde oder Kopien auf verschiedenen Ebenen des FDM-Systems. Ebenso könnte eine geografische Redundanz mit dem Partnerstandort in Tübingen in Betracht gezogen werden. Diese Elemente sind von verschiedenen Herstellern verfügbar, da eine breite Palette an Hierarchical Storage Management Systemen angeboten wird. Die Basis vieler solcher Systeme sind spezielle Filesysteme wie GPFS oder BeeGFS. Entsprechende Filesystem-Köpfe für SMB oder NFS werden üblicherweise angeboten.

Layer III implementiert die langfristige Datenpublikations- und Archivebene. Sie könnte in unterschiedlicher Form umgesetzt werden. So sollte mindestens eine S3-Schnittstelle für eine potenzielle Nutzung von internen und externen Cloud-Services bereitgestellt werden. Durch die Repository-Komponente des Gesamtsystems sollte ein durchgängig transparenter Zugriff (durchgängige Referenzierung) auf die Daten unabhängig vom tatsäch-

lichen physikalischen Lagerort sichergestellt werden. Hierbei sollte das jeweilige Nachweis- und Zugriffssystem damit umgehen können, dass die Daten unter Umständen erst mit Zeitverzögerung bereitgestellt werden.

Der Übergang von Daten in die Archivschicht sollte durch einen formalen Vorgang begleitet werden, an dem sowohl der Eigentümer die Daten abschließend durchsieht als auch ein Storage-Gremium der jeweiligen Community die Archivwürdigkeit bestätigt. Spätestens mit der Übernahme in den Layer III sind Daten mit geeigneten Persistent Identifiern [11] zu versehen und können damit permanent referenziert werden. Die Referenzierung könnte beispielsweise aus den Systemen der jeweiligen Universitätsbibliotheken heraus erfolgen. Ebenso wäre eine Einbindung in die DataCite-Infrastruktur [BD11] denkbar.

Data-Mover sind zentrale Bestandteile von modernen hierarchischen Storage-Systemen. Sie sorgen für eine geeignete Verteilung und Redundanz der Daten über das Gesamtsystem hinweg. Diese Komponente sollte dahingehend erweiterbar sein, dass sie zusätzliche Kriterien, die beispielsweise den Metadaten der Datensätze entnommen werden, in ihren Entscheidungen berücksichtigen kann. Ziel sollte es dabei sein, dem System Informationen in ausreichendem Umfang und Qualität bereit zu stellen, dass die geeignete Platzierung der Daten weiterhin weitgehend automatisch erfolgen kann. Manuelle Eingriffe sollten lediglich über die entsprechende Anpassung der Metadaten durch Nutzer oder Storage-Gremien laufen. Der Data-Mover muss Zugriff auf alle Layer des FDM-Storage-Stacks haben. Sollte die Archivebene nicht direkt Bestandteil des Systems sein, sollte er diese zumindest geeignet ansprechen können.

4 Mehrwert durch Qualifizierung der Daten

Die Speicherplatzbedürfnisse der einzelnen Forschungsprojekte können sehr verschieden ausfallen, weshalb einerseits unterschiedliche Arten des Zugriffs auf die FDM-Ressource als auch andererseits verschiedene Zustände der Datenhaltung [TGHR07] angenommen werden. Im Rahmen dieses Vorhabens wird grob von drei Schritten in der Datenhaltung ausgegangen (Abbildung 2). Der Einstieg (Schritt 1) erfolgt über die formale Anmeldung eines Forschungsvorhabens. Die Einstiegshürden hierfür werden bewusst niedrig gesetzt. An dieser Stelle werden eine Reihe grundsätzlicher Metadaten zum Projekt erhoben. Der beantragte Speicherplatz wird als (virtueller) Storage-Container mit den gewünschten Zugriffsmethoden für einen begrenzten Zeitraum [12] bereitgestellt. Dieses Verfahren stellt sicher, dass regelmäßig neuen Forschergruppen Speicher zugeteilt werden kann. Die Storage-Container werden mit den bereitgestellten wissenschaftlichen Metadaten und den vom System generierten technischen Metadaten verknüpft. Für den geforderten Workflow kann auf existierende Implementierungen im Land zurückgegriffen werden, die bereits seit einiger Zeit im bwHPC-Projekt produktiv im Einsatz sind. Die dort eingesetzte Plattform könnte für die Zwecke angepasst und erweitert werden, da sie viele notwendige Komponenten wie beispielsweise die Authentifizierung über bwIDM

[11] Vgl. [EU17], B2Handle

[12] Dieser Zeitraum könnte beispielsweise 100 Tage betragen und mit geringen formalen Hürden nochmal um diese Zeitspanne verlängert werden. Dieses wird im Rahmen der Governance vereinbart.

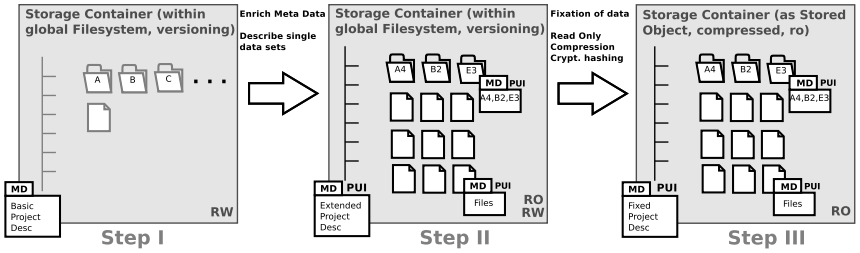


Abb. 2: Anreicherung von Metadaten, wenn Daten länger im System vorgehalten werden sollen (Schritt 1 zu 2) sowie Übergang zur Langzeiterhaltung (Schritt 2 zu 3)

bereits mitbringt. In diesem Stadium werden Daten als vorübergehender Arbeitsbereich („Scratch“) betrachtet. Auf diese Weise können Forschende mit wenig Aufwand Speicher einrichten. Dieses ist insbesondere für die erste Datenaufnahme und Verarbeitung relevant.

Der Data-Mover wird den Benutzer über den Ablauf der Haltefrist rechtzeitig informieren, so dass dieser die Chance hat, die Daten entweder geeignet zu klassifizieren oder selbst zu sichern. Ein Nutzer kann für einen bestimmten Zeitraum die Haltefrist der Daten durch einfachen Antrag verlängern, allerdings nicht beliebig oft. Dadurch wird sichergestellt, dass das System nicht mit Scratch-Daten überläuft, die keinen langfristigen Wert über den konkreten Arbeitsschritt des Nutzers hinaus besitzen.

Zu dem Augenblick, an dem Daten eine bestimmte Qualität erreicht haben und für einen längeren Zeitraum vorgehalten werden sollen, können sie für den längerfristigen Verbleib im FDM-System qualifiziert werden (Schritt 2). Dazu kann der Eigentümer sowohl weitere Projekt-Metadaten ergänzen als auch seine Daten mit zusätzlichen fachspezifischen Metadaten (als auch um Tags und Persistent Unique Identifiers) anreichern, so dass sie sich potenziell (öffentlich) referenzieren und zitieren lassen. Die Forschenden sollten im Sinne von Open Access verpflichtet werden, ihre Daten nach einer gewissen Zeit – welche vom Projekt, den Zuwendungsgebern oder der Community-Policy abhängen können – bereitzustellen. Dieses kann auch im Rahmen einer Publikation erfolgen. Auf diese Weise sollen die Forschenden ermutigt werden, eine hohe Qualität sowohl der Daten als auch der Metadaten sicherzustellen. Dieser Vorgang könnte beispielsweise Komponenten [13] analog zu einem „Publikationsvertrag“ für die Veröffentlichung eines Werkes über eine Universitätsbibliothek beinhalten.

Mit Erreichung bestimmter Projektziele beziehungsweise nach dem Projektabschluss ist eine längerfristige Datenhaltung vorzusehen (Schritt 3), so diese nicht in Community-eigenen Repositorien langfristig abgelegt werden. Eine geforderte, langfristige Datenhaltung könnten beispielsweise die von der DFG geforderten zehn Jahre für die Überprüfbarkeit von Forschungsergebnissen sein. Ebenso könnten die Daten wiederum für weitere Vorhaben (auch für externer Nutzer) von Interesse sein. Je nach Art und Häufigkeit des Zugriffs auf die Daten könnte der Data-Mover unterschiedliche Speichersysteme im Hintergrund vorsehen.

[13] Abtretung langfristiger Rechte, automatischer Rechteübergang nach einer bestimmten Zeit o.ä.

Jederzeit bei Bedarf, spätestens jedoch im Übergang von Schritt 2 zu 3 - dem Übergang in die dauerhafte Domäne - werden die Daten in der vorliegenden Form „eingefroren“ (Sicherstellung der Unveränderlichkeit). Sie werden dann üblicherweise aus dem System für den laufenden Zugriff in ein Object Store überführt. Sollte für eine spätere Anbindung und Nachnutzung der Daten ein Dateisystemzugriff erforderlich sein, ließe sich dieses über geeignete technische Maßnahmen [14] abbilden.

Weiterhin können beim Übergang in die dauerhafte Domäne automatische Prozesse im Schritt von Layer 2 zu 3 ausgelöst werden, die von Kompression über die Bildung von Prüfsummen bis hin zur Verschlüsselung der Daten reichen. Auf diese Weise sollen sowohl effiziente Datenhaltung als auch Integrität und Unveränderbarkeit der Datensätze sichergestellt werden. Die Daten bleiben weiterhin für die Nutzung verfügbar, werden jedoch dann als abgeleitete Versionen genutzt. [15] Hierdurch wird die notwendige Persistenz der Daten für ihre Zitier- und Referenzierbarkeit sichergestellt. In dieser Stufe wird der Data-Mover keine Daten automatisch löschen, sondern anhand der Metadaten (bspw. Hinweis auf eine Publikation), Zugriffsrechten und Zugriffshäufigkeit festlegen, wo die Daten geeignet lagern und wieviele (geo-)redundante Kopien erzeugt werden sollen.

Eine automatische Löschung könnte beispielsweise für Daten vorgesehen werden, deren Haltefrist abgelaufen ist und für die keine Zugriffe über einen bestimmten Zeitraum registriert wurden. Diese Entscheidung könnte an ein entsprechendes Governance-Gremium delegiert werden. Ebenso wären Fragen der Finanzierung langfristiger Datenhaltung mit den Heimat-Institutionen und einzelnen Communities abzustimmen. Unabhängig von der aktuellen Haltung der Daten sollten diese transparent mindestens ab Schritt 2 referenziert sein und beispielsweise via OAI-PMH oder entsprechender anderer geeigneter Systeme und Standards auffindbar sein. Die Referenzierbarkeit der Daten sollte unabhängig von der konkreten Ablage (auch bei externen Diensten wie bwDATA-Archiv, EUDAT) erhalten bleiben und je nach Festlegung durch das FDM-System gesteuert werden.

Qualifizierende Metadaten an Storage-Container und später den Datensätzen dienen der Steuerung des FDM-Gesamtsystems, der Wiederauffindbarkeit und einer guten Nachnutzbarkeit. Metadaten geben üblicherweise die jeweiligen Fach-Communities oder auch die Standards der Bibliotheken und (Daten-)Archive vor. Für die Zwecke des FDM-Systems kommen einige technische Metadaten hinzu, die entweder automatisch ermittelt oder vom Nutzer abgefragt werden. Hierzu zählen (insbesondere für Schritt 1): *Projektbeschreibung* die sich an etablierten Standards der Communities und Fördergebern wie der DFG orientiert und die Zuordnung von Forschungsprojekten erleichtern; *Eigentümer des Projekts*, üblicherweise Leitung einer Arbeitsgruppe oder eines Forschungsprojekts. Dies soll sicherstellen, dass Daten auch nach Weggang von Arbeitsgruppenmitglieder zugreifbar bleiben. Diese sind zudem Ansprechpartner für das Steuerungsgremium der betroffenen Wissenschafts-Community; *Erwartete Laufzeit des Projekts und erwarteter Umfang der Daten*, dient einerseits der Steuerung und andererseits der Abschätzung zukünftigen Speicherbedarfs im Zeitablauf; *Bevorzugter Typ der Nutzung des Speicher-Containers* (File-

[14] An dieser Stelle werden häufig sog. Fuse-Layer eingesetzt.
[15] Dieses wird technisch beispielsweise über Copy-on-Write Mechanismen umgesetzt.

system, Object Store, Versioning, ...) wird von der Workflow-Engine genutzt, um nach formaler Freigabe die notwendigen Ressourcen und Schnittstellen bereitzustellen.

Die spätere Qualifizierung der Daten in Schritt 2 wird insbesondere durch die Standards und Vereinbarungen der jeweiligen Communities bestimmt. Hier sollte das FDM-System in der Lage sein, verschiedene Metadaten-Standards [16] zu unterstützen.

Governance Zwischen den verschiedenen Wissenschafts-Communities, den Betreibern und den Zuwendungsgebern muss ein sinnvoller Ausgleich der Interessen und Kosten organisiert werden. [17] Storage-Kapazität unterscheidet sich insofern fundamental von Compute-Kapazität, dass nach Abschluss eines Projekts die Ressource nicht wieder automatisch frei wird und an weitere Nutzer weitergegeben werden kann. Es muss sichergestellt werden, dass auch später hinzukommende Forschungsgruppen das System für ihre Zwecke nutzen können. Während es für kurze Zeiträume kein Problem darstellt, auch größere Datenmengen vorzuhalten, wird dieses für lange Fristen kostenintensiv. Um einen guten Kompromiss aus Menge und Qualität der Daten zu erhalten, sind verschiedene Herangehensweisen denkbar. Eine Variante setzt auf einen höheren Grad des Selbstmanagements der Nutzer-Community vor dem Hintergrund vorliegender Forschungsdatenmanagementkonzepte. [18]

Der erste Schritt der Datenhaltung setzt weitgehend auf Self-Service und orientiert sich an den eingeführten Prozessen vergleichbarer Verbundprojekte. [19] Mit der Verfügbarkeit einer gewissen Basisberechtigung soll es einfach möglich sein, für zunächst begrenzte Zeit Speicherplatz für ein Forschungsprojekt zu beantragen. Um eine effiziente Nutzung insbesondere bei längerfristiger Belegung der Ressource zu erreichen, werden Schwellen eingebaut, die schrittweise eine hohe Qualität der Daten sicherstellen sollen. Diese beinhalten automatisierbare Anteile (Nutzung der Informationen aus den Metadaten) und Review-Prozesse durch (gewählte) Gremien, die beispielsweise fest in der akademischen Selbstverwaltung der einzelnen Fakultäten angesiedelt sein könnten. Diese könnten so Einfluss nehmen, für welche langfristige Datenhaltung Fakultätsbeiträge für ein einrichtungsweites FDM eingesetzt würden.

5 Vorläufiges Fazit

Der zunehmende Bedarf, Forschungsdaten zu speichern, zu qualifizieren und in wissenschaftliche Arbeitsabläufe einzubinden, muss adressiert werden. Dieser sollte nicht mehr wie bisher durch einzelne dezentrale Storage-Lösungen ohne wirkliche Langfristperspektive und nachhaltige Nutzung beziehungsweise Austausch der Daten bedient werden. So würden lediglich vorhandene Lösungen und Infrastrukturen mit einem hohen Kosten- und Management-Aufwand dupliziert. Die Diskussions- und Abstimmungsprozesse zwischen Universitätsleitung, zentralen Einrichtungen und den Wissenschafts-Communities laufen.

[16] Vgl. beispielsweise [EU17], Dienste des B2FIND
[17] Vgl. hierzu Darstellungen in [vo16], S. 281ff., S. 344ff.
[18] Vgl. Stand in Baden-Württemberg, `www.forschungsdaten.info`
[19] Vgl. Darstellungen in [vo16] bzw. „Zentrale Antragsseite“ (ZAS), `https://www.bwhpc-c5.de/ZAS`

Weitgehender Konsens besteht, dass ein (abstrakt) übergreifendes FDM-System angestrebt wird, welches sich in Verbundstrukturen einordnet und Schnittstellen und Austausch mit europäischen Lösungen sucht. Das System soll mit einer Kernkomponente, welche die wichtigen Dienste bietet, starten. Es soll durch regelmäßige Erneuerungen unter Beteiligung der Communities ausgebaut werden. Um sowohl eine hohe Qualität der Daten als auch eine effiziente Nutzung des Systems zu erreichen, werden die Wissenschafts-Communities in den Fakultäten und Instituten in den Aufbau der Workflows zur Qualifizierung der Daten und den Aufbau der Governance-Strukturen eingebunden. Das System wird gleichfalls versuchen, durch einen hohen Selfservice-Anteil den Forschenden weitgehende Freiheiten beispielsweise bei der Wahl der Metadaten und der Qualitätskriterien einzuräumen. Eine frühestmögliche Beteiligung der Communities ist vor allem im Hinblick auf die Akzeptanz und des Nutzen des FDM-Systems notwendig. Gleichzeitig müssen die Hürden für die Nutzung des Systems so gering wie möglich gehalten werden.

Literaturverzeichnis

[Ba12] Ball, Alexander: Review of Data Management Lifecycle Models. February 2012.

[BD11] Ball, Alex; Duke, Monica: How to cite datasets and link to publications. Digital Curation Centre, 2011.

[BHM11] Büttner, Stephan; Hobohm, Hans-Christoph; Müller, Lars: Handbuch Forschungsdatenmanagement. Bock+ Herchen, 2011.

[EMS] Eifert, Thomas; Muckel, Stephan; Schmitz, Dominik: Introducing Research Data Management as a Service Suite at RWTH Aachen. In: Ges. für Informatik eV (GI). S. 55.

[Er13] Erway, Ricky: Starting the Conversation: University-Wide Research Data Management Policy. ERIC, 2013.

[EU17] EUDAT Collaborative Data Infrastructure, 2017. `https://www.eudat.eu/`.

[Kl13] Klar, Jochen; Enke, Harry: Report „Organisation und Struktur“, 2013.

[Ne12] Neuroth, Heike; Strathmann, Stefan; Oßwald, Achim; Klump, Jens; Ludwig, Jens: Langzeitarchivierung von Forschungsdaten. Eine Bestandsaufnahme. 2012.

[Rf16] RfII – Rat für Informationsinfrastrukturen: Empfehlungen zu Strukturen, Prozessen und Finanzierung des Forschungsdatenmanagements in Deutschland, 2016. `http://www.rfii.de/?wpdmdl=2075`.

[TGHR07] Treloar, Andrew; Groenewegen, David; Harboe-Ree, Cathrine: The data curation continuum: Managing data objects in institutional repositories. D-Lib magazine, 13(9):4, 2007.

[Tr15] Tristram, Frank; Bamberger, Peter; Cayoglu, Ugur; Hertzer, Jörg; Knopp, Johannes; Kratzke, Jonas; Rex, Jessica; Schwabe, Fabian; Shcherbakov, Denis; Svoboda, Dieta-Frauke; Wehrle, Dennis: Öffentlicher Abschlussbericht von bwFDM-Communities, 2015. `http://bwfdm.scc.kit.edu/downloads/Abschlussbericht.pdf`.

[vo16] von Suchodoletz, Dirk; Schulz, Janne; Leendertse, Jan; Hotzel, Hartmut; Wimmer, Martin: Kooperation von Rechenzentren Governance und Steuerung – Organisation, Rechtsgrundlagen, Politik. de Gruyter, 2016.

Skalierbare virtuelle Netz-Testbeds für Lehr- und Forschungsumgebungen mit VIRL

Sebastian Rieger [1]

Abstract: Lehrveranstaltungen und Forschungsprojekte, in denen Experimente im Netzwerkbereich durchgeführt werden, verwenden häufig virtuelle Netzwerkumgebungen. Diese können je nach erforderlicher Praxisnähe unterschiedliche Software-Lösungen bzw. Experimentierumgebungen verwenden. Eine hohe Praxisnähe in Bezug auf die Funktionalität des virtuellen Netzes bietet die Emulation von Netzen. Allerdings erfordert sie aufgrund der Nachbildung realer Netzfunktionen und -komponenten z.B. im Vergleich zu Simulationen weitaus mehr Ressourcen, was die Skalierbarkeit zugunsten der Praxisnähe erschwert. Das Paper beschreibt die im Umfeld des Netzwerk-Labors (NetLab) an der Hochschule Fulda eingesetzten Lösungen für emulierte virtuelle Netz-Testbeds und zeigt Möglichkeiten für die Bewertung von deren Leistungsfähigkeit und Skalierbarkeit auf.

Keywords: Netz-Virtualisierung, Simulation, Emulation, OpenStack, VIRL.

1 Einleitung

Für die Realisierung von experimentellen Netzwerkumgebungen für Forschungsprojekte und Lehrveranstaltungen haben sich unterschiedliche Möglichkeiten etabliert. Diese können anhand ihrer Ausrichtung auf Praxis oder Theorie differenziert werden. Abbildung 1 zeigt unterschiedliche Ausrichtungen und einige Beispiele für passende Werkzeuge.

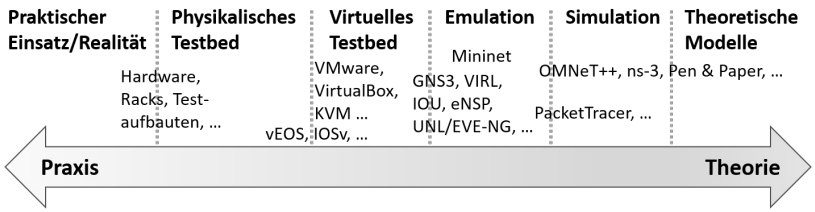


Abb. 1: Klassifizierung von Umgebungen für Netzwerkexperimente.

Auf der einen Seite können praxisnahe Testumgebungen, wie in der linken Seite der Abbildung gezeigt, direkt in realen Netz- und IT-Infrastrukturen realisiert werden. Dies ist allerdings aufgrund der Beeinflussung des Produktivbetriebs durch sowie auf das Testbed häufig nicht praktikabel. Um dieses Problem zu umgehen, können physikalische Testbeds isoliert vom Produktivsystem betrieben werden. In der Regel ist dies jedoch aufwändig und kann nur bedingt an schnell veränderliche Anforderungen angepasst werden. Virtuelle Testbeds reduzieren diesen Aufwand erheblich. Netzstrukturen (Links, Netzwerkkomponenten)

[1] Hochschule Fulda, Fachbereich Angewandte Informatik, Leipziger Straße 123, 36037 Fulda, sebastian.rieger@informatik.hs-fulda.de

müssen darin allerdings häufig manuell nachgebildet werden. Auf der anderen Seite können komplexe Netze, wie im rechten Teil der Abbildung 1 dargestellt, auch in rein theoretischen Modellen abgebildet werden. Eine Umsetzung der formalen Definition von Netzen und Erkenntnissen aus diesen Modellen in reale Umgebungen kann jedoch aufgrund der fehlenden Praxisnähe eine große Herausforderung bilden. Simulationen nähern sich der Praxis an und bieten den Vorteil, dass das Verhalten der zugrundeliegenden theoretischen Modelle exakt (z.B. bzgl. des Timings) auf Anforderungen aus der Praxis ausgerichtet werden kann. Zusätzlich kann innerhalb deterministischer Simulationen die Zeit im Gegensatz zu realitätsnäheren Ansätzen angehalten sowie vor- oder zurückgestellt werden. Simulationen verwenden ein abstraktes Netz, das das Verhalten in der Praxis nicht vollständig nachbilden kann. Entsprechend ist der Realitätsbezug abhängig von der konkreten Fragestellung, für die die Simulation entworfen wurde.

In [Ha12] wird basierend auf einer ähnlichen Klassifizierung eine Emulation statt Simulation der Netzkomponenten und Links empfohlen. Die Autoren des Papers bewerten dabei die Ziele bzw. Vorteile der Emulation von Netzumgebungen in den Bereichen: *Functional Realism*, *Timing Realism*, *Traffic Realism*, *Topology Flexibility*, *Easy Replication* und *Low Cost*. Emulatoren erfüllen gemäß der Bewertung in [Ha12] lediglich die Anforderung *Timing Realism* nicht ohne Weiteres exakt. Dies deckt sich auch mit den Erfahrungen aus den im Netzwerk-Labor (NetLab) an der Hochschule Fulda eingesetzten Lösungen. Entsprechend bildet die Emulation derzeit eine flexible Grundlage für die Realisierung von praxisnahen Netz-Experimentierumgebungen. Allerdings entsteht durch die realitätsnahen virtuellen Maschinen (VMs) im Vergleich zu Simulationen etc. ein hoher Ressourcenbedarf. Für große virtuelle Netztopologien sind zudem die lokalen Ressourcen von Laborrechnern schnell erschöpft. Dieses Paper zeigt eine Möglichkeit für die Verbesserung der Skalierbarkeit und die Verwendung einer zentral verwalteten Virtualisierungsplattform auf.

Die nächsten beiden Abschnitte stellen Beispiele für Netzwerkexperimentierumgebungen, die im NetLab verwendet werden, vor und treffen eine Auswahl für die in diesem Paper vorgestellte skalierbare Plattform für virtuelle Netz-Testbeds. Abschnitt 2 beschreibt die in diesem Paper verwendeten Netztopologien. Im Abschnitt 3 wird eine Testumgebung für die anschließend in Abschnitt 4 bewertete Skalierbarkeit von Netz-Testbeds unter Verwendung des Virtual Internet Routing Lab (VIRL) von Cisco realisiert. Abschließend zieht Abschnitt 5 ein Fazit und gibt einen Ausblick auf weitergehende Betrachtungen.

1.1 Virtuelle Umgebungen für Netzwerkexperimente

Die Emulation stellt gewissermaßen einen Mittelweg zwischen Theorie und Praxis dar und realisiert häufig ein virtuelles Testbed. So können reale Systeme (vgl. Linux VMs) und Netzwerkmanagementwerkzeuge (Wireshark etc.) innerhalb der Testbeds eingesetzt werden. Für Lehrveranstaltungen (insb. Übungen und Praktika) sowie Forschungsprojekte im Umfeld des NetLab werden *physikalische Testbeds*, *virtuelle Testbeds*, *Emulation* und *Simulation* eingesetzt. Analog zu den in [Ha12] diskutierten Ergebnissen hat sich dabei die Emulation als besonders flexibel herausgestellt. *Physikalische Testbeds* werden im NetLab z.B. für praxisnahe Projekte der Studierenden beispielsweise in Form von mobilen

Experimentier-Racks für studienbegleitende Cisco-Zertifizierungen (vgl. CCNA, CCNP) eingesetzt [PS11]. Sie bieten teilweise einen höheren *Functional Realism*, *Traffic Realism* und *Timing Realism*, jedoch zugunsten weitaus größerer Einschränkungen bei *Topology Flexibility*, *Easy Replication* und *Low Cost*. Für den Einsatz in Übungen, in denen die in der Vorlesung vermittelte Theorie veranschaulicht werden soll, sind komplexe Aufbauten mit *physikalischen Testbeds* häufig zu zeitintensiv. Auch die Realisierung von *virtuellen Testbeds* (z.B. dezentral mit VMware Workstation oder zentral mit VMware vSphere) erfordert einen hohen Aufwand bzgl. der Vorbereitung und Wartung der VMs und Netze. Sie werden daher im NetLab insb. für praxisnahe Client-/Server-Anwendungen sowie in isolierten Umgebungen z.B. für Experimente im Bereich der IT-Sicherheit eingesetzt. Die VMs müssen hierbei über die Semester hinweg überarbeitet (Updates, Änderungen am Setup bzw. der Netz- und IT-Infrastruktur) und verwaltet werden, was *Topology Flexibility* sowie *Easy Replication* einschränkt. Simulatoren wie ns-3 [ns17] und OMNeT++ [OM17] werden in Master-Veranstaltungen angesprochen. Für praxisnahe Übungen werden sie im NetLab jedoch neben ihrer Komplexität insb. aufgrund des fehlenden *Functional Realism* und *Traffic Realism* nicht verwendet.

Für die o.g. Cisco-Zertifikate wird PacketTracer [Pa17] eingesetzt. In Übungen zu Lehrveranstaltungen kam jedoch Kritik an dessen fehlender Praxisnähe auf. Beispielsweise werden Clients lediglich simuliert und bieten keine vollständige Implementierung gängiger Werkzeuge (z.B. reduzierter Funktionsumfang bei arp, ping, traceroute). Zusätzlich müssen Eigenheiten der Simulation erklärt werden. Beispielsweise verwirft PacketTracer das erste ICMP-Paket eines Pings am Router im Ziel-Subnetz, da dieser zunächst per ARP die MAC-Adresse des Ziel-Rechners ermitteln muss. Dieses für eine Simulation durchaus korrekte Verhalten tritt jedoch in der Praxis nicht auf, da der Client unmittelbar nach dem Boot-Vorgang bereits Pakete über den Router gesendet hat, und seine MAC-Adresse somit bereits im ARP-Cache des Routers liegt. PacketTracer zeigt daher Schwächen im Bereich *Functional Realism*, *Traffic Realism* und *Timing Realism*.

1.2 Begründung der Auswahl und Verwendung von VIRL

Im vorherigen Abschnitt wurde die Auswahl basierend auf den Anforderungen des NetLab bereits auf den Bereich „Emulation“ der Abbildung 1 begrenzt. Das z.B. in [LHM10] und [Ha12] in diesem Bereich verwendete Mininet [Mi17] wird im NetLab für SDN-Übungen eingesetzt. Aufgrund der Container-basierten Emulation besitzt es einen geringen Ressourcenbedarf. Allerdings erfordert die Realisierung von eigenen Netztopologien eine Einarbeitung in die Python-API von Mininet und es können nur bedingt reale Netz-Infrastrukturen/-Software integriert werden. Für eine praxisnahe Emulation von Netzumgebungen im NetLab wurden daher in den vergangen Jahren GNS3 [GN17], EVE-NG [Em17], eNSP [eN17] und VIRL [VI17] evaluiert. Alle setzen primär auf die Integration von VMs und erlauben so den virtuellen Einsatz realer Netz-Soft- und Hardware. eNSP ist auf Netzkomponenten von Huawei ausgerichtet. Die Clients sind analog zu PacketTracer simuliert und eingeschränkt. Wie auch GNS3 und EVE-NG kann eNSP allerdings lizenzkostenfrei verwendet werden. In Bezug auf die genannten Emulatoren besitzt GNS3 die größte Verbreitung und ist am längsten auf dem Markt, bietet jedoch kein cluster-basiertes scale-out und ist somit bzgl.

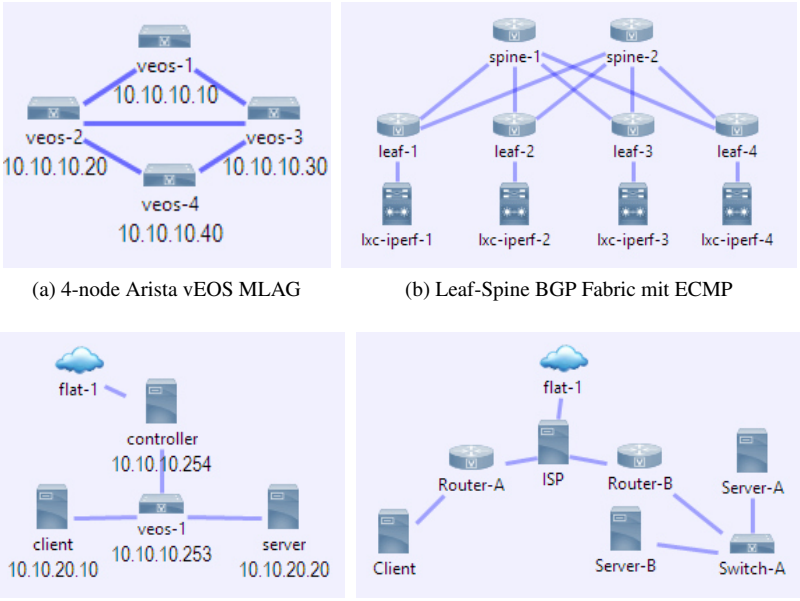
der Skalierbarkeit einzelner Topologien limitiert bzw. erlaubt keine Zusammenarbeit von Studierenden an laufenden Topologien. EVE-NG bietet hierfür eine Lösung und setzt darüber hinaus auf eine fortschrittliche web-basierte Konfiguration. Allerdings befindet sich EVE-NG noch im Alpha-Stadium. Alle bislang genannten Funktionen bietet auch VIRL. Aufgrund der auf OpenStack basierenden Architektur ist bei VIRL zusätzlich eine zentrale und skalierbare Cluster-Installation möglich. Emulierte Netztopologien können daher von den Studierenden gemeinsam und ortsunabhängig (im NetLab sowie auf dem eigenen Notebook bzw. von zu Hause) verwendet und mittels GIT versioniert verwaltet werden. Die offene Architektur von OpenStack und VIRL (auf der Basis von Ubuntu 14.04, LXC, linux-bridge, VXLAN) ermöglicht außerdem eigene Erweiterungen für die Realisierung von skalierbaren Netz-Testbeds für Lehr- und Forschungsumgebungen. Außerdem können Studierende im Vergleich zu den anderen Alternativen offizielle Cisco-Images nutzen, die bei GNS3 und EVE-NG separate Lizenzen erfordern.

VIRL erlaubt die direkte Verwendung von Betriebssystemen, die auch auf physikalischen Netzwerkkomponenten laufen (vgl. Arista EOS, Cisco IOS), in Form von VMs, was die im vorherigen Abschnitt genannte Anforderung *Functional Realism* nahezu vollständig erfüllt. Durch die virtuelle Vernetzung dieser VMs kann realer Traffic eingespielt (Capture) oder nachgebildet werden, was sowohl *Traffic Realism* als auch in Teilen *Timing Realism* bietet. Die VMs können dabei nach Bedarf flexibel zu virtuellen Netzwerktopologien zusammengeschaltet werden, um so im Vergleich zur Realisierung mit realer physikalischer Hardware eine bessere *Topology Flexibility* sowie *Easy Replication* zu erfüllen. Nachteilhaft sind in Bezug auf das Kriterium *Low Cost* im Vergleich zu den anderen genannten Alternativen die Kosten, die durch die Einzelplatz-Lizenzen von VIRL entstehen und maximal 20 Cisco-Nodes in laufenden Topologien erlauben. Trotz dieses Nachteils stellt VIRL derzeit in Bezug auf die anderen genannten Anforderungen eine praktikable Lösung für die im NetLab durchgeführten Experimente dar. Es erlaubt zusätzlich als einzige Lösung derzeit eine zentral verwaltete Umgebung bei der die Studierenden gemeinsam auf verteilt laufenden Topologien im Cluster arbeiten können. Der nächste Abschnitt zeigt einige Beispiele für die Realisierung von Topologien mit VIRL in Kombination mit praxisnahen Architekturen und Werkzeugen auf.

2 Beispiele für emulierte Netz-Testbeds und -Experimente im NetLab

Die Abbildung 2 zeigt einige Beispiele der im NetLab für Übungen und Praktika mit VIRL verwendeten Netztopologien. Abbildung 2a zeigt eine Topologie mit 4 Arista vEOS (4.16.9) Nodes. Die Links zwischen den vier Nodes sind jeweils redundant ausgelegt. Master-Studierende experimentieren im Team in unterschiedlichen Szenarien z.B. mit dem Einsatz des Multiple Spanning Tree Protocol (MSTP), Link Aggregation Control Protocol (LACP) und Multi-chassis Link Aggregation Group (MLAG), um die Auswirkungen des Spanning-Tree-Protokolls und dessen Erweiterungen auf die Nutzung der verfügbaren Links z.B. in Data Center Networks kennenzulernen. Aufbauend auf dieser Übung arbeiten die Master-Studierenden in der in Abbildung 2b gezeigten Topologie an Leaf-Spine-Architekturen mit BGP Fabrics, wie sie z.B. in Cloud-Rechenzentren von Facebook und Microsoft eingesetzt werden. Dabei kommen für die Endpunkte Ubuntu 14.04 Container (LXC)

zum Einsatz. Die Spine und Leaf Switches werden mit IOSv (15.6(2)T) umgesetzt. In der Vergangenheit wurden ähnliche Topologien mit vEOS (ebenfalls BGP) und NX-OSv (FabricPath) umgesetzt. Auf den LXC Containern können die Studierenden per SSH reguläre Linux-Kommandos verwenden. Die Nodes und Links können während einer laufenden Emulation gestoppt bzw. getrennt werden. Außerdem kann der Traffic auf jedem Link aufgezeichnet und mit Wireshark ausgewertet, sowie QoS-Parameter (Delay, Jitter, Packet Loss) auf den Links angepasst werden. Für SDN-Experimente mit OpenDaylight (als OpenFlow Controller) und Arista vEOS (als OpenFlow 1.3 Switch) wird die in Abbildung 2c gezeigte Topologie verwendet. Hierfür wurde zuvor eine Mininet-basierte Übung eingesetzt. Durch die Verwendung von VIRL können die Studierenden praxisnah unter Verwendung von vEOS, das sich analog zum EOS auf physikalischen Arista Switches verhält, den Einsatz von OpenFlow testen.



(a) 4-node Arista vEOS MLAG (b) Leaf-Spine BGP Fabric mit ECMP

(c) vEOS/OpenDaylight SDN Beispiel (d) IOSv/netem Routing/Switching Toubleshooting

Abb. 2: Beispiele für emulierte Netz-Testbeds und Netzwerkexperimente.

Ein Beispiel für die Verwendung weniger komplexer Netztopologien zeigt die Abbildung 2d. In dieser Übung ermitteln Bachelor-Studierende Fehlkonfigurationen (ARP, Routing, Delay/Packet Loss, Port Status/Shutdown), fertigen durch Verwendung realer Netzwerk-Tools (ping, traceroute/mtr, Wireshark) Netztopologien an und realisieren den Zugriff auf einen auf Server-B installierten Apache Web-Server. Client, Server und ISP bilden Ubuntu 14.04 VMs. Beim ISP wurde zusätzlich mit netem ein künstliches Delay und Packet Loss hinzugefügt (WAN Emulation). Der ISP Node bindet die Topologie als Standard-Gateway an das reale lokale Netz (flat-1) an. Damit ist ein Zugriff auf das Internet (ping auf google.com etc.) aus der Emulation möglich. Aus dem Labor können die Studierenden zusätzlich per OpenVPN auf Nodes im emulierten Netz (z.B. den Web-Server) zugreifen. Ein Zugriff außerhalb

des Labors (z.B. von zu Hause) ist über ein VPN-Profil der Hochschule möglich. Neben IOSv und Ubuntu Nodes kommt für den Switch IOSvL2 (15.2(4.0.55)E vgl. Catalysts) zum Einsatz. Alle genannten Beispiele werden per GIT unter [HS17a] bereitgestellt. Für den Einsatz von VIRL im NetLab wurden einige Erweiterungen implementiert. Diese umfassen z.B. eine an der Hochschule entwickelte Anpassung von Arista vEOS und CumulusVX Images für den Einsatz in VIRL [Ri17].

3 Realisierung einer VIRL Benchmark-Umgebung

Im Vergleich zu Simulationen, die in der Regel in wenigen Sekunden gestartet sind, benötigen die realitätsnahen virtuellen Testbeds in VIRL teilweise mehrere Minuten für den Start. Um diesen Nachteil zu reduzieren, wurde eine Cluster-Umgebung (VIRL 1.2.83) realisiert und evaluiert, um die Startzeit der Topologien zu optimieren. Da die Topologien direkt einsatzfähig konfiguriert sind und nach dem Start sofort laufen bzw. Daten übertragen, ist aus der Startzeit auch ein Rückschluss auf die Leistungsfähigkeit der Emulation während des Betriebs möglich. Die für die VIRL Umgebung verwendete Hardware wurde bereits in [PTR15] beschrieben. Als VIRL Hosts wurden hierbei vier VMs mit je 64 GB RAM und 32 vCPUs verwendet. Die VMs bilden eine Nested Virtualization innerhalb einer VMware vSphere 6.0 Umgebung, wobei jede der vier VIRL VMs durch eine DRS-Beschränkung explizit auf einem der vier physikalischen ESXi Hosts läuft. Die ESXi Hosts bieten 2 8-Kern Intel(R) Xeon(R) E5-2650v2 2.60GHz CPUs sowie 256GB RAM und nutzen als Storage-Backend zwei NetApp E2700. Die Knoten sind mit zweimal 1 GbE an einen Cisco C3850 und mit zweimal 10 GbE an einem Arista 7150S-24 und Arista 7050S-52 Switch angeschlossen. Wie auch in [Ha12] diskutiert wurde, hängt die Anforderung *Timing Realism* in Bezug auf die Emulation insb. in virtualisierten Umgebungen stark von der Isolation der Emulationsumgebung ab. Daher wurden für die nachfolgenden Benchmarks isolierte Resource Pools definiert und die Tests bewusst während der vorlesungsfreien Zeit durchgeführt sowie die Gesamtauslastung der VMware vSphere-Umgebung überwacht. Der VIRL Benchmark bildete zum jeweiligen Testzeitpunkt die einzigen VMs, die eine signifikante Last auf der vSphere-Umgebung erzeugten. Um zusätzlich etwaige Ausreißer in den Performance-Messungen für die in diesem Paper betrachtete Skalierbarkeit von VIRL zu reduzieren, wurde ein Skript entwickelt, das einen reproduzierbaren Test ermöglicht.

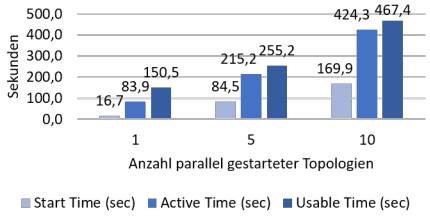
Für den Test wurde die in Abbildung 2a gezeigte Topologie verwendet, da sie über eine vergleichsweise kleine und damit gut skalierbare Node-Anzahl verfügt. Je Testzyklus werden die emulierten Netztopologien zunächst via VIRL REST-API gestartet, die Zeit bis zur Bestätigung des Starts gemessen und anschließend der Zeitpunkt ermittelt, zu dem alle Nodes *ACTIVE* waren. *ACTIVE* bedeutet in diesem Zusammenhang, dass die VMs vom OpenStack Nova Scheduler auf die VIRL Hosts verteilt wurden, die notwendigen virtuellen Netze und Ports angelegt wurden, das vEOS Image bereitgestellt ist und der Boot-Vorgang der VMs beginnt. Abschließend werden in dem entwickelten Benchmark-Skript Verbindungen zur Konsole der Nodes hergestellt und dadurch sowohl das interaktive Delay für Eingaben auf der Konsole als auch die Zeit gemessen, bis die Nodes tatsächlich durch die Benutzer verwendet werden können. Dafür wurde ein Python-Skript geschrieben, das eine WebSocket-Verbindung zur seriellen Konsole der Nodes auf den jeweiligen VIRL Hosts

herstellt, dreimal ENTER übermittelt und die Zeit bis zu den erwarteten resultierenden drei Prompt-Zeilen mit dem Hostnamen misst. Der geschilderte Testablauf wurde zunächst 10x nacheinander für jeweils eine emulierte Netztopologie gestartet. Dann wurden 10x nacheinander 5 parallel emulierte Netztopologien und schließlich 10x nacheinander 10 parallel Netztopologien gestartet und die o.g. Messungen bis zur vollständigen Nutzbarkeit der Topologien ermittelt. Jeder vEOS Node belegt 1 VCPU und 2 GB RAM. Somit waren für die Testzyklen mit 10 parallel gestarteten Topologien insg. 10 x 4 vEOS Nodes x 2 GB = 80 GB RAM erforderlich, die ab dem Boot-Vorgang sukzessive von KVM für die Prozesse reserviert wurden. Das Skript für den Benchmark und die dafür entwickelten Werkzeuge können unter [HS17b] abgerufen werden. Der komplette Testabauf wurde mit unterschiedlicher Cluster-Node-Anzahl wiederholt, um Rückschlüsse auf die Skalierbarkeit zu erhalten. Dabei wurde zunächst von einem einzelnen VIRL Host auf einen 2-Node Cluster und schließlich auf einen 4-Node VIRL Cluster erweitert. Der Testlauf wurde zu unterschiedlichen Tageszeiten wiederholt, um etwaige Seiteneffekte auf die Messungen zu minimieren. Die gemittelten durchschnittlichen Testergebnisse werden im nächsten Abschnitt präsentiert und ausgewertet.

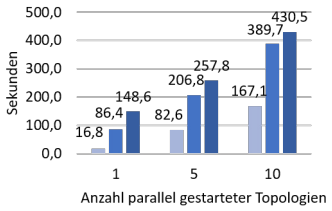
4 Bewertung der Skalierbarkeit

Abbildung 3 zeigt die Ergebnisse des im vorherigen Abschnitt erläuterten Testablaufs. Im Hinblick auf die Auswirkungen der Anzahl parallel gestarteter Topologien lässt sich z.B. in Abbildung 3a erkennen, dass die *Start Time* (Zeit bis alle per REST-API an VIRL übermittelten Topologien gestartet wurden) erwartungsgemäß linear ansteigt. Mit zunehmender Anzahl parallel gestarteter Topologien steigt jedoch die *Active Time* (Zeit bis alle VMs in den gestarteten Topologien booten) im Vergleich zur *Start Time* stärker an. Die *Usable Time* (Zeit bis die Konsole der gestarteten vEOS Nodes nutzbar ist) liegt mit zunehmender Node-Anzahl näher an der *Active Time*. Bis zu einer Anzahl von 20 parallel gestarteten Topologien sinkt der Zuwachs von *Active Time* und *Usable Time* zunächst. Zusätzliche Cluster Nodes steigern diesen Effekt, sofern die Anzahl parallel gestarteter Topologien größer als 5 ist. Bei mehr als 20 parallel gestarteten Topologien (vgl. Abbildung 3c) steigt die Wartezeit bis zur vollständigen Verfügbarkeit der Topologien allerdings bedingt durch die komplette Auslastung der Ressourcen des 4-Node Clusters wieder an. Wie in Abbildung 3c erkennbar ist, nimmt der Zuwachs an Wartezeit auf 1 oder 5 Topologien erwartungsgemäß durch den Aufwand für die Verteilung und Anbindung der zusätzlichen Cluster Nodes leicht zu. Diese Effekte lassen auf die Skalierbarkeit des OpenStack Schedulers schließen. Mit zunehmender Anzahl parallel gestarteter Nodes sinkt der Overhead durch das Scheduling. Neben dem Overhead durch das Scheduling resultieren die vergleichsweise langen Wartezeiten auf das Starten der VMs (Abstand zwischen *Start Time* und *Active Time*) auch aus dem aufwändigen Prozess der Anlage von virtuellen Netzen. Jeder Link in Abbildung 2a ist redundant und erfordert die Anlage von zwei VXLAN-Segmenten und dazugehörigen Ports in den Linux Bridges etc. Dies äußert sich auch in einer hohen Auslastung der Neutron Prozesse auf dem Controller Node des OpenStack Clusters. Testweise wurde die Anzahl der neutron-server und nova-api sowie nova-conductor Worker Prozesse auf 10 erhöht. Aufgrund der hierfür zusätzlich erforderlichen Ressourcen konnten

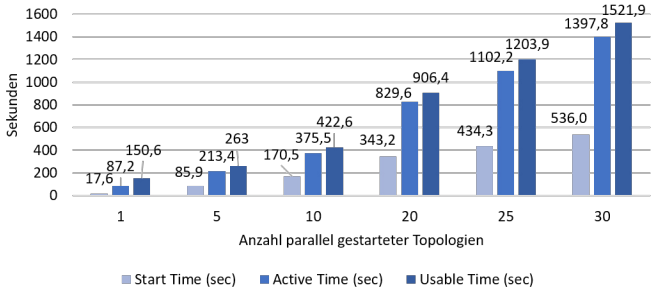
danach nur noch maximal 20 Topologien gestartet werden. Entsprechend wurde der auf dem Controller des VIRL Clusters benötigte Arbeitsspeicher reserviert. Dies reduziert die Wartezeit auf die 20 parallel gestarteten Topologien gemäß Abbildung 3d jedoch nur um ca. 11%. Hier zeigt sich noch Verbesserungspotenzial bzgl. der Anpassung der Konfiguration des OpenStack Managements (Scheduler, Message Bus) sowie dem nachgelagerten KVM.



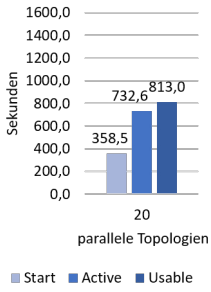
(a) Startzeit Topologie Abb. 2 (a) - 2-Node Cluster



(b) 4-Node Cluster mit ramdisk



(c) Startzeit Topologie Abb. 2 (a) - 4-Node Cluster



(d) 4-Node - mehr Worker

Abb. 3: Benchmark-Ergebnisse für die Startzeit der emulierten Netztopologien.

Limitierende Faktoren bilden in Bezug auf den Benchmark eher RAM und I/O als die CPU-Leistung. CPU-Last entsteht neben den OpenStack- und VIRL-Management-Prozessen primär beim Booten der vEOS Instanzen, wobei die zugewiesene VCPU für durchschnittlich 60 Sekunden zu 100% ausgelastet ist. Von Cisco wird für VIRL die Option angeboten eine ramdisk für die Instanzen (Nova VMs) zu nutzen. Die Abbildung 3b zeigt den testweisen Einsatz einer ramdisk. Da dies von Cisco nur für den Controller unterstützt wird, wurden die Compute Nodes manuell angepasst. vEOS Instanzen nutzen allerdings nur jeweils 213 MB ephemeral storage. Daher ist der positive Effekt einer ramdisk gering bzw. wird bei zunehmender Anzahl paralleler Topologien durch den Performance-Verlust durch das zusätzlich belegte RAM (vgl. Abbildung 3b) zunichte gemacht.

5 Fazit und Ausblick

Durch Cisco VIRL wird für Lehrveranstaltungen wie insb. Praktika und Übungen sowie für Forschungsprojekte eine praxisnahe und skalierbare Plattform für virtuelle Netz-Testbeds bereitgestellt. Gegenüber Alternativen wie GNS3 und EVE-NG bietet VIRL den Vorteil, dass offizielle Cisco-Images ohne zusätzliche Lizenzen verwendet werden können. Der

weitaus wichtigere Vorteil gegenüber den Alternativen ist jedoch die OpenStack-basierte Cluster-Lösung von VIRL, die mit einigen manuellen Anpassungen, wie im Paper präsentiert, eine Skalierbarkeit auch für große Topologien sowie eine zentral verwaltete Lösung für große Benutzergruppen ermöglicht. Für VIRL Hosts wird so ein scale-out für virtuelle Testbeds möglich. Durch die Verwendbarkeit von Standard-Netzwerk-Management-Werkzeugen (vgl. Wireshark, ping, traceroute, usw.) bzw. Ubuntu Containern und VMs in den emulierten Netztopologien sowie die Anbindung an reale externe Netze und das Internet wird im Vergleich zu Netz-Simulatoren eine hohe Flexibilität und Realitätsnähe erreicht. Die damit verbundene im Vergleich zu Netz-Simulatoren längere Startzeit, wird durch einen für das NetLab in PHP und Python entwickelten VIRL-Scheduler zusätzlich kompensiert. Dieser wird verwendet, um zeitgesteuert vor einer Übung Topologien für die Studierenden zu laden und so eine direkte Verwendung in der Lehrveranstaltung zu erlauben. In einem Bachelor-Projekt ist zudem derzeit von mehreren Studierenden ein VIRL-Kursmanagement in Entwicklung. Dies soll neben dem VIRL-Scheduler auch die automatisierte Anlage von Kursen (Benutzern und Gruppen) für VIRL sowie ein Reservierungssystem für automatisierte Starts von Topologien außerhalb der Zeiten der Lehrveranstaltungen erlauben. Das Dev/Innovate/Research Program, über das die Hochschule Fulda die Cluster-Lizenz für dieses Paper erhalten hat, soll 2017 auslaufen. Derzeit wird evaluiert auf die offizielle Mehrplatz-Version von VIRL (Cisco Modeling Labs (CML)) umzustellen, oder eine Einzellizenz für den Cluster zu verwenden. Die dadurch entstehende Limitierung auf max. 20 Cisco-Nodes wäre im Labor vertretbar, zumal andere VMs (z.B. Arista, Cumulus, Linux) trotzdem unbeschränkt verwendet werden können. Die aktuelle Entwicklung bei GNS3 und EVE-NG läuft ebenfalls in Richtung skalierbarer Cluster-Lösungen, in denen Topologien gemeinsam von mehreren Nutzern verwendet werden können, so dass hierbei ggf. eine kostengünstige Alternative zu VIRL für das NetLab entsteht.

Bzgl. weiteren Performance-Steigerungen sind Optimierungen des OpenStack Scheduling und der Netz-Infrastruktur, wie in der Bewertung in diesem Paper angesprochen, denkbar. Im Bereich SDN bleibt Mininet durch die extrem schnelle Startzeit von Switches und Containern, trotz geringerer Praxisnähe, eine interessante Alternative. Gleiches gilt für einige Übungen, in denen virtuelle Testbeds basierend auf lokalen VMware Workstation/VirtualBox Installationen oder physikalische Racks aufgrund einer geringen Komplexität der Netzexperimente besser einsetzbar sind. Zukünftig ist durch neue OpenStack Versionen mit einer Steigerung der Performance des Scheduling (nova-scheduler) sowie der Prozesse für die Verwaltung der virtuellen Netz-Infrastruktur (neutron-server) zu rechnen. Derzeit werden diese in VIRL nur mit jeweils einem Prozess ausgeführt und limitieren daher die Startzeit der emulierten Netztopologien. Zudem wird nur ein sequentielles Scheduling ermöglicht, was einen hohen Traffic auf der verwendeten Message Queue und Datenbank sowie den Log-Files etc. zur Folge hat. Dies bietet ebenfalls Ansatzpunkte für zukünftig im Umfeld des NetLab betrachtete Verbesserungen.

6 Danksagung

Für das NetLab wurde von Cisco eine VIRL Lizenz im Rahmen des Dev/Innovate/Research Program zur Verfügung gestellt.

Literaturverzeichnis

[Em17] Emulated Virtual Environment Next Generation (EVE-NG) / Unified Networking Lab (UNL), http://www.unetlab.com, Stand: 3.1.2017.

[eN17] eNSP - Enterprise Network Simulator, http://support.huawei.com/enterprise/en/network-management/ensp-pid-9017384, Stand: 3.1.2017.

[GN17] GNS3 - The software that empowers network professionals, https://www.gns3.com, Stand: 3.1.2017.

[Ha12] Handigol, Nikhil; Heller, Brandon; Jeyakumar, Vimalkumar; Lantz, Bob; McKeown, Nick: Reproducible network experiments using container-based emulation. In: Proceedings of the 8th international conference on Emerging networking experiments and technologies. ACM, S. 253–264, 2012.

[HS17a] HS-Fulda NetLab VIRL Topologien, https://gogs.informatik.hs-fulda.de/srieger/git-virl-hs-fulda, Stand: 3.1.2017.

[HS17b] HS-Fulda NetLab VIRL Utilities, https://gogs.informatik.hs-fulda.de/srieger/virl-utils-hs-fulda, Stand: 3.1.2017.

[LHM10] Lantz, Bob; Heller, Brandon; McKeown, Nick: A network in a laptop: rapid prototyping for software-defined networks. In: Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks. ACM, S. 19, 2010.

[Mi17] Mininet - An Instant Virtual Network on your Laptop (or other PC), http://mininet.org, Stand: 3.1.2017.

[ns17] ns-3, https://www.nsnam.org, Stand: 3.1.2017.

[OM17] OMNeT++ Discrete Event Simulator, https://omnetpp.org, Stand: 3.1.2017.

[Pa17] Packet Tracer - A free network simulation and visualization tool for the IoT era., https://www.netacad.com/about-networking-academy/packet-tracer, Stand: 3.1.2017.

[PS11] Pape, Christian; Seifert, Christoph: Adaption and improvement of an industry-developed IP Telephony curriculum. In: 7th Annual International Conference on Computer Science and Education in Computer Science. Sofia/Dobrinishte, Juli 2011.

[PTR15] Pape, C.; Trommer, R.; Rieger, S.: Energieverbrauch von Live-Migrationen in OpenStack-basierten Private-Cloud-Umgebungen. In: 8. DFN-Forum - Kommunikationstechnologien, 8.-9. Juni 2015, Lübeck, Germany. 2015.

[Ri17] Arista vEOS image on VIRL, https://learningnetwork.cisco.com/thread/99040, Stand: 3.1.2017.

[VI17] VIRL - Virtual Internet Routing Lab, http://virl.cisco.com, Stand: 3.1.2017.

Müller, Neumair, Reiser, Dreo-Rodosek (Hrsg.): 10. DFN-Forum Kommunikationstechnologien,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 11

Eine vollautomatisierte e-Learning Plattform am Beispiel eines Universitäspraktikums

Jan Schmidt[1], Nils gentschen Felde[1]

Abstract:

Das Skalierungsverhalten vieler universitärer Praktika ist den derzeitig stetig wachsenden Teilnehmerzahlen häufig nicht gewachsen. Insbesondere die zumeist sehr zeitaufwändige Korrekturarbeit durch das Betreuungspersonal verhindert größere Teilnehmerzahlen, was zu einem Mangel an verfügbaren Praktikumsplätzen führt. Ziel dieser Arbeit ist es, die notwendigen Konzepte und ein System zu entwickeln und zu implementieren, das es ermöglicht, Praktikumsaufgaben vollautomatisiert zu überprüfen. Außerdem soll den Studenten die Möglichkeit gegeben werden, ihre Lösungen bereits während der Implementierung zu überprüfen, um ein eigenständiges Lernen zu unterstützen. In dieser Arbeit werden anhand beispielhafter Praktika Anforderungen an ein solches Prüfungssystem erhoben und daraus ein Systementwurf abgeleitet. Der Systementwurf wird beispielhaft für das Praktikum IT-Sicherheit implementiert. Die Implementierung wird durch eine Kombination aus dem Python-Framework *Flask* und der Task-Queue *Celery* realisiert, um eine skalierbare Web-Anwendung zu erhalten. Die Praxistauglichkeit wird direkt durch den produktiven Einsatz der erhaltenen Lösung an der LMU München belegt.

Keywords: e-Learning, Virtuelle Lehr- & Forschungsumgebungen, Automated Grading

1 Einleitung

In Zeiten stetig steigender Studentenzahlen haben sich im Laufe der Zeit bereits eine Menge technischer Entwicklungen ergeben, die das Skalierungsverhalten von Lehrveranstaltungen (insbesondere von Praktika) erheblich positiv beeinflussen. Im Falle der LMU bedeutet das z. B., dass viele der technisch notwendigen Arbeitsumgebungen und IT-Infrastrukturen virtualisiert wurden [DGK16; Li05; Li06; LRg08; Sc11], so dass eine immer größere Zahl an Studenten vollkommen zeit- und ortsungebunden an einem Praktikum teilnehmen kann. Die Kehrseite der Medaille jedoch ist, dass die persönliche Betreuung und das inhaltliche Anleiten der Studenten – vor allem bedingt durch die exzessive Nutzung der neu gewonnenen Freiheiten bei der Orts- und Zeitwahl und der folglich geringeren Präsenzzeit an der Universität – immer schwieriger wird.

Aus diesem Grund wird in dieser Arbeit ein System entworfen, das die Übungsaufgaben im Rahmen eines Praktikums vollautomatisiert korrigieren kann und so den Korrekturaufwand

[1] MNM-Team, Ludwig-Maximilans-Universität München, Oettingenstr. 67, 80538 München, Germany, Email: {schmidtja,felde}@nm.ifi.lmu.de

für die Betreuer verringert. Auch soll den Studenten die Möglichkeit geboten werden, dass sie ihre Lösungsansätze bereits während der Bearbeitung der Praktikumsaufgaben testen können. Dazu werden im folgenden Kapitel themenverwandte Arbeiten analysiert, bevor Kapitel 3 durch eine Anwendungsfall-getriebene Anforderungsanalyse einen Kriterienkatalog für einen Systementwurf hervorbringt. Auf Basis dieses Kriterienkatalogs wird in Kapitel 4 eine Systemarchitektur abgeleitet und deren Implementierung kurz umrissen, bevor Kapitel 5 die erhaltene Lösung bewertet und Kapitel 6 die Ergebnisse zusammenfasst und einen Ausblick auf weiterführende Arbeiten gibt.

2 Themenverwandte Arbeiten

Es existiert eine Menge Arbeiten und Ansätze zu automatisierten e-Learning Umgebungen, im Rahmen derer diverse Übungsaufgaben durch die Teilnehmer / Studenten bearbeitet und gelöst werden müssen. Die Überprüfung der gelösten Aufgaben erfordert jedoch häufig Humaninteraktion, zumeist von freiwilligen Helfern, was das Skalierungsverhalten der angebotenen Initiativen stark von der Anzahl der Betreuer abhängig macht. Ziel dieser Arbeit ist, genau diese Lücke zu schließen.

Eine bekannte Initiative zur Ausbildung im Bereich IT-Sicherheit ist das sog. *Hacking-Lab*[2]. Das *Hacking-Lab* ist eine Online-Plattform, die eine vernetzte IT-Infrastruktur bereitstellt, innerhalb welcher verschiedene Aufgaben und Herausforderungen zur IT-Sicherheit gelöst werden können. Ein *Hacking-Lab* besteht aus sog. „challenges“, die jeweils aus einer existierenden Schwachstelle bestehen, die es zu finden und beheben gilt. Eine Lösung besteht aus der Beschreibung einer gefundenen Schwachstelle und der Dokumentation einer entsprechenden Absicherung. Eine Bewertung der Lösungen wird von den Betreuern des *Hacking-Lab*s vorgenommen und per Email an die Teilnehmer versendet. Das *Hacking-Lab* basiert technisch auf dem *Open Web Application Security Project*[3] und stellt u. a. auch die Missionen im Rahmen der *European Cyber Security Challenge*[4] bereit. Ähnliche Arbeitsumgebungen, in denen insbesondere Pen-Testing gelehrt und geübt werden kann, sind das *Hacking Dojo*[5], das *Virtual Hacking Lab*[6] sowie *LAMPSecurity*[7].

Viele Vorarbeiten zur automatischen Korrektur von Programmieraufgaben finden sich unter dem Stichwort *Automated Grading* [Bl04; EP08; HPK11; KLC01; SHS15]. Sämtliche Arbeiten beschränken sich allerdings auf die Korrektur von Programmieraufgaben und eignen sich nicht dafür, beliebige praktische Aufgaben, die innerhalb eines universitären Praktikums oder *Hacking-Labs* existieren, zu korrigieren. Ein großer Teil der Lösungen ist jedoch web-basiert und kann als gute Basis für das angestrebte System dienen.

[2] https://www.hacking-lab.com/index.html
[3] https://www.owasp.org/index.php/OWASP_Hacking_Lab
[4] http://www.europeancybersecuritychallenge.eu/
[5] http://hackingdojo.com/lab/
[6] https://sourceforge.net/projects/virtualhacking/
[7] https://sourceforge.net/projects/lampsecurity/

Ein Prüfungssystem für ein virtualisiertes Praktikum ist in [BR13] dokumentiert. Es werden virtuelle Maschinen auf den Systemen der Teilnehmer durch eine Sammlung von Shell-Skripten überprüft und eine entsprechende Bewertung generiert. Zur Überprüfung werden die jeweiligen Tests auf die zu überprüfenden Systeme der Teilnehmer kopiert und ausgeführt. Das führt folglich leider dazu, dass Testroutingen potentiell durch den Teilnehmer eingesehen und im schlimmsten Fall sogar manipuliert werden könnten.

3 Anforderungsanalyse

Auch an der LMU existieren bereits Vorarbeiten und voll-virtualisierte Arbeitsumgebungen im Bereich der Lehre [DGK16; Li05; Li06; LRg08; Sc11], an deren Beispiel eine möglichst weitreichende automatische Korrektur von zu lösenden Übungsaufgaben realisiert werden soll. Diese Arbeiten dienen im Folgenden als Ausgangsbasis für die vorliegende Arbeit. In einem ersten Schritt werden die an den Praktika beteiligten Akteure und ihre Rollen identifiziert. Erwartungsgemäß sind dies die studentischen Teilnehmer, die Lehrstuhlmitarbeiter in ihrer Rolle als inhaltliche Betreuer und Prüfer des Praktikums sowie die technischen Administratoren der unterstützenden IT-Infrastrukturkomponenten.

Die unterschiedlichen Sichtweisen der drei beteiligten Rollen (Student, Betreuer, Administrator) dienen als Ausgangsbasis für eine Anwendungsfall-getriebene Anforderungsanalyse. Die dabei angewandte Methodik folgt der aus dem objektorientierten Software-Entwurf bekannten Vorgehensweise der Analyse von Anwendungsfällen (sogenannte *Use Cases*). Der Ausgangspunkt der Anwendungsfallanalyse ist die knappe und informelle Beschreibung ausgewählter Szenarien. Zur Ableitung von Anwendungsfällen werden die vier Phasen des Lebenszyklus einer Praktikums-Instanz (Planung, Aufbau / Inbetriebnahme, Betrieb inkl. Prüfung, De-Kommissionierung) und zusätzlich die Evolution (Änderung, Erweiterung etc.) des Praktikums über die Jahre hinweg betrachtet. Die Beschreibung der verschiedenen Lebenszyklusabschnitte aus den jeweils unterschiedlichen Blickwinkeln wird als Grundlage für die Analyse der verschiedenen Anwendungsfälle verwendet. Insgesamt leiten sich nach diesem Schema 29 Anwendungsfälle ab [Sc16], die sich auf die vier Lebenszyklusphasen verteilen. Beispiele für Anwendungsfälle sind das Gruppieren von Übungsaufgaben zu Aufgabenblättern, die Zuordnung von Studenten zu Arbeitsgruppen, Änderungen an den Arbeitsgruppen während des Semesters (z. B. Wechsel eines Studenten in eine andere Gruppe, frühzeitiges Ausscheiden eines Studenten, o. ä.), die Archivierung und Reproduzierbarkeit von Testergebnissen oder schlicht die Integration des angestrebten Systems in die bereits existierende Betriebsumgebung. Aus der Summe der Anwendungsfälle werden Anforderungen abgeleitet und in einem Anforderungskatalog gesammelt.

Zusammenfassung der Ergebnisse der Anforderungsanalyse

Zusammenfassend ergeben sich 29 funktionale und 15 nicht-funktionale Anforderungen. Neben den durch die Anwendungsfälle abgeleiteten Anforderungen ergibt sich noch eine weitere Menge an Anforderungen aus formalen und rechtlichen Vorgaben, die z. B. in Prü-

fungsordnungen und sonstigen Studienregularien sowie der Gesetzgebung festgeschrieben sind. Die meisten der hierdurch gegebenen Anforderungen sind Duplikate der bereits im Rahmen der Anwendungsfall-getriebenen Anforderungsanalyse erhobenen Anforderungen (z. B. Reproduzierbarkeit der Ergebnisse, Archivierung etc.). Einige andere Anforderungen hingegen erweisen sich im weiteren Verlauf der Arbeit als nur sehr kompliziert oder gar nicht IT-gestützt umsetzbar (z. B. die eindeutige Zurechenbarkeit von Ergebnissen zu Studenten, insbesondere zum Zwecke der Notengebung).

4 Systementwurf und Implementierung

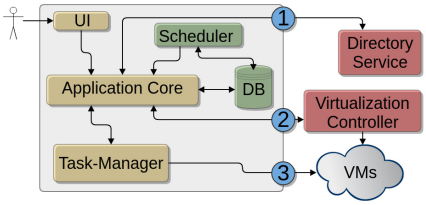


Abb. 1: erster Systementwurf

Um die Bestandteile des Systems und ihre Beziehungen untereinander festzulegen, werden in einem ersten Schritt die erhobenen Anforderungen gruppiert und daraus Komponenten abgeleitet. Abb. 1 zeigt einen ersten groben Überblick über die abgeleiteten Komponenten, Kommunikationsbeziehungen und ihren Schnittstellen zu Bestandssystemen aus der Vogelperspektive.

Aus den Anforderungen an die Datenhaltung (wie z. B. die Verwaltung von Teilnehmerdaten, Terminen und Fristen oder die Zuordnung von Aufgaben zu Übungsblättern) folgt, dass eine zentrale Komponente zur Speicherung relevanter Daten und Informationen (also eine Datenbank (*DB*)) benötigt wird. Zudem beinhaltet der Anforderungskatalog einige Anforderungen an die Interaktionen mit bestehender Infrastruktur, Diensten und den bestehenden virtuellen Maschinen (vgl. rechte Seite in Abb. 1). Der Systementwurf wird diesen Anforderungen in Form von drei Schnittstellen (①, ② und ③) gerecht:

Schnittstelle ① stellt die Kommunikation mit einem existierenden Verzeichnisdienst (*Directory Service*) sicher, der zur Authentifizierung und Autorisierung registrierter Nutzer verwendet werden soll.

Schnittstelle ② realisiert die Kommunikation mit der bestehenden Virtualisierungsumgebung unter Nutzung existierender Management-Komponenten (*Virtualization Controller*). Der notwendige Funktionsumfang der Schnittstelle ergibt sich aus einer Gruppe nichtfunktionaler Anforderungen, die fordern, dass z. B. Testergebnisse reproduzierbar sind, keine Spuren auf den getesteten Systemen hinterlassen werden, und dass Tests nicht durch Dritte beeinflussbar sind. Daraus folgt ebenfalls, dass der Funktionsumfang dieser Schnittstelle zumindest das Sichern des Systemzustandes einzelner VMs sowie die Zugriffssteuerung und das Management von VMs unterstützen muss.

Schnittstelle ③ stellt die eigentliche Kommunikation zwischen dem Prüfungssystem und den virtuellen Maschinen sicher. Zum einen müssen die VMs vor einem Test vorbereitet werden (z.B. einrichten geeigneter Firewall-Regeln), zum anderen müssen Testroutinen Zugriff auf das System erhalten, um dort ausgeführt werden zu können.

Die Kommunikation unter Nutzung der vorgesehenen Schnittstellen sowie die Verwaltung der zentralen Abläufe und das Auflösen potentieller Abhängigkeiten von Aufgaben (*Task*s) untereinander soll durch einen zentralen Applikations-Kern (*Application Core*) übernommen werden. Teil dieses Kerns ist auch ein *Scheduler*, der u. a. den Forderungen nach Skalierbarkeit, Fehlertoleranz und Nebenläufigkeit (z. B. durch die parallele Testausführung mehrerer Übungsgruppen) gerecht wird. Zudem ist im Systementwurf ein eigener *Task-Manager* als *Producer* nach dem *Producer-Consumer*-Schema vorgesehen, der die auszuführenden *Task*s erstellt, die von *Consumern* als eigenständige Prozesse abgearbeitet werden.

Aus einer weiteren Gruppe von Anforderungen ergibt sich schlussendlich die Notwendigkeit einer Sichtenbildung und eines Rollenkonzepts. Das bedeutet für den Systementwurf, dass eine Nutzerschnittstelle (*User Interface (UI)*) mit unterschiedlichen Sichten entsprechend einem festgelegten Rollenkonzept benötigt wird. Zudem beeinflussen viele nicht-funktionale Anforderungen (vor allem an die Sicherheit) die Festlegung von Protokollen und bedingen die Wahl geeigneter kryptographischer Verfahren, wie auch im weiteren Verlauf dieses Kapitels genauer beschrieben.

4.1 Detaillierter Systementwurf am Beispiel des Praktikums IT-Sicherheit

Auf Basis der einleitenden Überlegungen lässt sich eine verfeinerte Systemarchitektur als Grundlage für eine spätere Implementierung leicht an einem Beispiel darstellen. Abb. 2 stellt das Ergebnis graphisch dar.

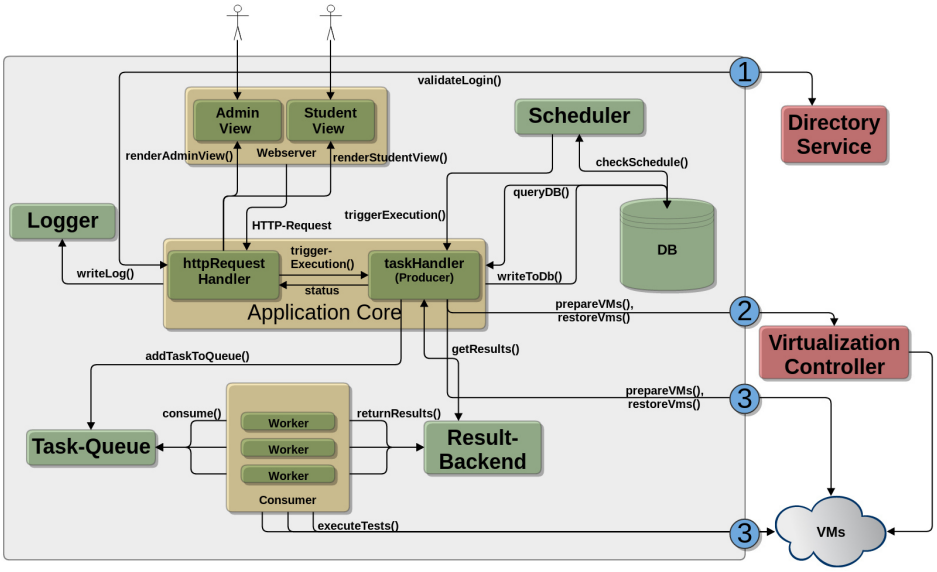


Abb. 2: Verfeinerte Systemarchitektur [Sc16]

Webserver. Die geforderte Sichtenbildung des Systementwurfs wird durch einen Webserver mit unterschiedlichen Sichten für administrative Aufgaben und für Studenten realisiert.

Application Core. Der *Application Core* stellt die zentrale Komponente des Gesamtsystems dar. Es werden alle Anfragen des Webservers verarbeitet, aus denen sogenannte *Task*s erzeugt werden, die wiederum jeweils von einem *Worker* verarbeitet werden. Außerdem wird die Verarbeitung der *Task*s überwacht. Dementsprechend teilt sich der *Application Core* in zwei Teilmodule auf, die die beiden Teilaufgaben funktional abdecken:

`httpRequestHandler`. Neben der Authentifizierung und Autorisierung von Benutzern unter Nutzung eines externen Verzeichnisdienstes (*Directory Service*, wobei „extern“ lediglich bedeutet, dass der Verzeichnisdienst bereits existiert und nicht unter der Ägide des hier entwickelten Systems steht), verarbeitet der *Request Handler* die Anfragen des *Webserver*s.

`taskHandler`. Es gibt zwei verschiedene Quellen, von denen *Task*s vom *taskHandler* entgegen genommen werden können. Zum einen können Nutzer Anfragen stellen, die vom *httpRequestHandler* weitergereicht werden. Zum anderen können geplante oder regelmäßige Aufgaben, die durch einen *Scheduler* zur Ausführung gebracht werden, eintreffen. Regelmäßig ablaufende *Task*s können z. B. das automatische Kontrollieren von Praktikumsaufgaben nach Ablauf der Bearbeitungsfrist für ein Aufgabenblatt sein.

Neben den verschiedenen Initiatoren können ebenfalls zwei verschiedene Arten von *Tasks* unterschieden werden: 1.) Die Überprüfung von Praktikumsaufgaben, 2.) die Kommunikation mit der virtuellen Infrastruktur unter Nutzung des externen *Virtualization Controllers*.

Alle *Task*s werden mit ggf. notwendigen Zusatzinformationen aus der Datenbank angereichert und an eine *Task-Queue* weitergegeben. Dort werden sie von *Worker*n abgearbeitet. Der *taskHandler* überwacht den Status der *Task*s und gibt diesen regelmäßig an den *httpRequestHandler* weiter, um ihn im *User Interface* anzeigen zu können. Nachdem ein *Worker* seinen *Task* abgearbeitet hat, wird das Ergebnis in ein *Result-Backend* geschrieben und kann vom *taskHandler* gelesen und verarbeitet werden.

Scheduler. Um z. B. das automatische Kontrollieren von Praktikumsaufgaben nach Ablauf der Bearbeitungsfrist für ein Aufgabenblatt zu realisieren, kommt ein *Scheduler* zum Einsatz. In regelmäßigen Abständen werden die in der Datenbank erfassten Zeitpläne überprüft und alle ggf. anstehenden Aufgaben als *Task*s zur Ausführung an den *taskHandler* weitergegeben.

Task-Queue und Result-Backend. Die *Task-Queue* dient als Warteschlange für *Task*s, die noch von keinem *Worker* verarbeitet werden. Außerdem werden über die *Task-Queue* Informationen über den Status zwischen *Worker* und *taskHandler* ausgetauscht. Nachdem ein *Worker* seinen *Task* abgeschlossen hat, wird das Ergebnis in ein *Result-Backend* eingetragen, aus dem es vom *taskHandler* ausgelesen und weiterverarbeitet kann.

Worker und Testroutinen. Die *Worker* sind eigenständige Prozesse, die die *Task*s aus der *Task-Queue* abarbeiten. Sie führen die eigentlichen Testroutinen aus. Eine Testroutine

ist eine ausführbare Datei, die lediglich mit ihrem Pfad als Referenz in der Datenbank gespeichert ist und eigenständig (ohne weitere Abhängigkeiten) ausgeführt werden kann.

Bevor eine Testroutine ausgeführt wird, werden die zu testenden virtuellen Maschinen durch den *taskHandler* vorbereitet. Dies beinhaltet, dass den Teilnehmern die Rechte zur Verwaltung der VMs entzogen werden, der aktuelle Zustand der VMs durch Snapshots gesichert und auf jeder VM eine Firewall eingerichtet wird, die sicherstellt, dass nur das Testsystem Zugang während der Testausführung erhält. Nach dieser Vorbereitung kann jeder Test von einem *Worker* ausgeführt werden.

Die Kommunikation mit den Testroutinen erfolgt über wohldefinierte Ein- und Ausgaben. Die Testroutinen erhalten als Eingabeparameter die IP-Adressen der zu testenden VMs. Ein Test muss so entwickelt werden, dass er bestimmte Kommandozeilenparameter (wie z. B. `--eth0-ipv6`) verarbeiten kann. Mit diesen Parametern kann sich der Test (meist per SSH) mit der VM verbinden. Anschließend wird der eigentliche Test durchgeführt. Da die Tests eigenständig sind und auch keine Verbindung zur Datenbank haben, ist der Rückgabewert nicht die erreichte Punktzahl einer Aufgabe, sondern ein entsprechender Prozentsatz der erreichten Punkte, der später mit der für die Aufgabe erreichbaren Punktzahl multipliziert wird. Zusätzliche Ausgaben der Testroutinen beinhalten Hinweise und Fehlermeldungen. Diese werden anschließend vom *taskHandler* an das *User Interface* weitergereicht, um die Teilnehmer bei ihrer Arbeit und Fehlersuche zu unterstützen.

Nach Abschluss aller auszuführenden Testroutinen werden die betroffenen VMs durch Wiedereinspielen der zuvor angelegten Snapshots in ihren ursprünglichen Zustand versetzt und sämtliche Berechtigungen der Teilnehmer werden erneut zugewiesen.

Datenbank (DB). Aus operativen Gründen ist es notwendig, dass die angemeldeten Studenten eines Kurses sowie deren Gruppenzugehörigkeit verwaltet werden können. Außerdem müssen die Aufgaben und Übungsblätter des Praktikums verwaltet werden. Jedes Übungsblatt hat eine Bearbeitungsfrist (*Deadline*) und mehrere Aufgaben, die diesem Übungsblatt zugeordnet werden. Nach Ablauf der Bearbeitungsfrist werden die entsprechenden Testroutinen der jeweiligen Aufgaben ausgeführt und die Bewertung jeder Gruppe in der Datenbank gespeichert. Ein dafür geeignetes und hier eingesetztes Datenmodell ist in [Sc16] zu finden.

4.2 Implementierung

Das entworfene System entspricht im Grundsatz einer Web-Anwendung mit Task-Management. Es bietet sich für die Implementierung an, auf bestehende Bibliotheken und Rahmenwerke zurückzugreifen, um die Grundfunktionalität des Entwurfs zu gewährleisten. Die Implementierung erfolgt in Python auf Basis von *Flask*[8]. *Flask* ist aufgrund seiner Flexibilität und unkomplizierten Entwicklung besonders gut geeignet.

[8] http://flask.pocoo.org/

Der *httpRequestHandler* sowie der *Web-Server* mit seinen *Views* ist in *Flask* realisiert. Das Datenmodell ist durch das SQL-Toolkit *SQLAlchemy*[9] und seinen *Object Relational Mapper* umgesetzt. Für den *taskHandler*, die *Worker* sowie den *Scheduler* wird *Celery*[10] verwendet. *Celery*-Tasks werden über die *Task-Queue* an die *Worker* verteilt, und *Celery* übernimmt die Kommunikation mit der *Task-Queue* und dem *Result-Backend*. *Celery* unterstützt verschiedene *Task-Queues*, *Message-Broker* und *Result-Backends*. *Redis*[11] wird sowohl für die *Task-Queue* als auch für das *Result-Backend* verwendet.

Die Implementierung erfolgt beispielhaft für das Praktikum IT-Sicherheit an der LMU. Dieses Praktikum verwendet als Virtualisierer *VMware ESXi*. Die Kommunikation mit dem entsprechenden *Virtualization Controller* (in diesem Fall also dem *vCenter*) wird durch Powershell-Skripte gelöst. Diese werden unter Nutzung der Python-API von *Ansible*[12] ausgeführt. Die Powershell-Skripte abstrahieren somit die Funktionalität VMs steuern und Snapshots verwalten zu können.

5 Bewertung der Ergebnisse

Zur Bewertung der erzielten Ergebnisse bietet es sich an, a) einen Abgleich mit den abgeleiteten Anforderungen zu führen und b) die (subjektiven) Eindrücke zum produktiven Einsatz des Systems in der Lehre zu betrachten.

Bei einer Überprüfung des Anforderungskatalogs zeigt sich, dass von 29 funktionalen Anforderungen 26 erfüllt werden konnten, 3 nur teilweise. Von den 15 nicht-funktionalen Anforderungen wurden 9 erfüllt, 4 wurden nur teilweise erfüllt. 2 nicht-funktionale Anforderungen konnten gar nicht erfüllt werden. Aus Platzgründen werden nachfolgend lediglich die beiden nicht erfüllten Anforderungen kurz beleuchtet.

Wie aus der Systemarchitektur ersichtlich (vgl. Abb. 2), setzt der Entwurf einen existierenden und gepflegten Verzeichnisdienst voraus. Im konkreten Fall der hier angeführten Implementierung existiert zwar ein solcher Verzeichnisdienst in Form eines Verwaltungssystems für Lehrveranstaltungen, jedoch ist keine Schnittstelle für den automatisierten Datenaustausch vorhanden. Als notwendiges Übel muss also ein Datenexport des Verzeichnisdienstes manuell importiert werden, was inhärent zu einer doppelten Datenhaltung und damit potentiell zu Inkonsistenzen führt. Zwar ist die Nichterfüllung der Konsistenzanforderung nicht durch das System bedingt, aber in der vorliegenden Implementierung unumgänglich.

Weiter konnte die Anforderung, falsch-positive und falsch-negative Ergebnisse der Testroutinen zu vermeiden, nicht erfüllt werden. Dies liegt vor allem daran, dass die Ergebnisse einer Testroutine von der Aufgabenstellung und der konkreten Implementierung der Testroutinen

[9] http://www.sqlalchemy.org/
[10] http://www.celeryproject.org/
[11] http://redis.io
[12] https://www.ansible.com/

abhängig sind. Ein möglicher Lösungsansatz, der auch umgesetzt ist, versucht dieses Problem durch Kontrollgruppen zu verringern. Es existiert je eine Kontrollgruppe mit der korrekten Lösung und eine Kontrollgruppe ohne Lösung, so dass zumindest grobe Fehler in den Testroutinen erkannt werden können. Eine vollumfängliche Erfüllung der Anforderung scheint nur unter Anwendung formaler Methoden machbar, was bisher nicht geschehen ist.

Die neue Arbeitsumgebung und die neuartige Art und Weise, ein Praktikum anzubieten, wurde von den Studenten gut angenommen und hat für eine gute Grundlage für Fragen im Tutorium oder per E-Mail gesorgt. Die Korrekturzeit der Aufgaben wurde durch die automatische Korrektur auf ein Minimum reduziert. Durch die Automatismen und die Ausprägung der Testroutinen sind zudem zusätzliche (Verständnis-) Fragen eingegangen, da die Lösungen nicht mehr nur den eigenen Ansprüchen der Studenten, sondern auch den Testroutinen genügen müssen.

Kritik hingegen gab es vorwiegend bei der Interpretation der Statusmeldungen von Testroutinen, wenn eine Aufgabe nicht mit voller Punktzahl bewertet worden ist. Hier ist die größte Herausforderung, dass auf der einen Seite vielsagende Rückmeldungen wünschenswert sind, aus der Rückmeldung aber auf der anderen Seite nicht direkt auf die richtige Lösung der Aufgabe geschlossen werden können soll.

6 Zusammenfassung & Ausblick

Im Rahmen dieser Arbeit ist ein Automatismus entstanden, der die Korrektur praktischer Übungsaufgaben im Rahmen eines universitären Praktikums übernimmt. Als Ausgangsbasis der Arbeit dienen bereits virtualisierte e-Infrastrukturen zur Ausbildung im Bereich vernetzter Systeme. Durch die Virtualisierung bedingt skalieren die eingesetzten Infrastrukturen bereits sehr gut, jedoch entwickelt sich schnell ein Flaschenhals durch die noch immer notwendige (persönliche) Betreuung von Teilnehmern sowie die Korrektur bearbeiteter Aufgaben. Eine Anwendungsfall-getriebene Anforderungsanalyse auf Basis existierender Praktika bringt einen Katalog an Anforderungen hervor, der systematisch zu einer Systemarchitektur führt, die im Nachgang implementiert und produktiv in Betrieb genommen wurde. Die Erfahrungen im operativen Einsatz sind sehr gut – sowohl objektiv durch einen hohen Grad der Erfüllung existierender Anforderungen, als auch subjektiv im Sinne positiver Rückmeldung der Praktikumsteilnehmer.

Neben Erweiterungen der Implementierung, wie z. B. die Erweiterung des derzeitigen Aufgabenpools oder der Anbindung an existierende Verzeichnisdienste und Dienste zur Veranstaltungsverwaltung, steht die Weiterentwicklung zu einer Art „Praktikum as a Service“ an. Studenten sollen den Startzeitpunkt ihres Praktikums sowie die Bearbeitungsgeschwindigkeit vollkommen frei wählen können. Dazu bedarf es einiger Modifikationen, insbesondere des zugrundeliegenden Datenmodells sowie der prozessualen Abläufe.

Literatur

[Bl04] Blumenstein, M.; Green, S.; Nguyen, A.; Muthukkumarasamy, V.: GAME: a Generic Automated Marking Environment for programming assessment. In: International Conference on Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. Bd. 1, 212–216 Vol.1, Apr. 2004.

[BR13] Baumstark, L.; Rudolph, E.: Automated Online Grading for Virtual Machine-based Systems Administration Courses. In: Proceeding of the 44th ACM Technical Symposium on Computer Science Education. SIGCSE '13, ACM, Denver, Colorado, USA, S. 477–482, 2013.

[DGK16] Danciu, V.; Guggemos, T.; Kranzlmüller, D.: Schichtung virtueller Maschinen zu Labor- und Lehrinfrastruktur. In: 9. DFN Forum Kommunikationstechnologien. GI–Edition Lecture Notes in Informatics, Rostock, Deutschland, Juni 2016.

[EP08] Edwards, S.; Perez-Quinones, M.: Web-CAT: Automatically Grading Programming Assignments. In: Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education. ITiCSE '08, ACM, Madrid, Spain, S. 328–328, 2008.

[HPK11] Hull, M.; Powell, D.; Klein, E.: Infandango: Automated Grading for Student Programming. In: Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education. ITiCSE '11, ACM, Darmstadt, Germany, S. 330–330, 2011.

[KLC01] Kurnia, A.; Lim, A.; Cheang, B.: Online Judge. Comput. Educ. 36/4, Mai 2001.

[Li05] Lindinger, T.: Machbarkeitsanalyse zur Virtualisierung des IT–Sicherheit Praktikums, Techn. Ber., LMU, Okt. 2005.

[Li06] Lindinger, T.: Virtualisierung einer Praktikumsinfrastruktur zur Ausbildung im Bereich Sicherheit vernetzter Systeme, Diplomarbeit, LMU, Mai 2006.

[LRg08] Lindinger, T.; Reiser, H.; gentschen Felde, N.: Virtualizing an IT–Lab for Higher Education Teaching. In: Tagungsband zum 1. GI/ITG KuVS Fachgespräch „Virtualisierung". Paderborn, Deutschland, S. 97–104, Feb. 2008.

[Sc11] Schmidt, B.: Konzeptionelle und praktische Erweiterung des Praktikums IT–Sicherheit rund um das Thema IPv6, Techn. Ber., LMU, Nov. 2011.

[Sc16] Schmidt, J.: Vollautomatisiertes Prüfungssystem am Beispiel des Praktikums IT-Sicherheit, Masterarbeit, LMU München, Aug. 2016.

[SHS15] Schlarb, M.; Hundt, C.; Schmidt, B.: SAUCE: A Web-Based Automated Assessment Tool for Teaching Parallel Programming. In: Euro-Par 2015: Parallel Processing Workshops. Springer, Cham, S. 54–65, Aug. 2015.

Virtualisierte wissenschaftliche Forschungsumgebungen und die zukünftige Rolle der Rechenzentren

Konrad Meier[1], Björn Grüning[2], Clemens Blank[3], Michael Janczyk[4], Dirk von Suchodoletz[5]

Abstract: Virtualisierungs- und Cloud-Technologien haben den Wandel der IT-Landschaft der letzten Jahre erheblich geprägt und werden von der Wissenschaft zunehmend für ihre eigenen Zwecke genutzt. Rechenzentren sollten auf den steigenden Bedarf in Forschung und Lehre mit der Bereitstellung geeigneter Infrastrukturen antworten. Es wird untersucht, wie Cloud-, Compute- und Lehrpool-Umgebungen so bereitgestellt werden können, dass eine bestmögliche Versorgung der Wissenschaft bei gleichzeitig effizienter Nutzung vorhandener Ressourcen erreicht werden kann. Durch die Erzeugung geeigneter virtualisierter Forschungs- und Lehrumgebungen können verschiedene Wissenschafts-Communities ihre Vorhaben durchführen, ohne hierfür weiterhin eigene Hardware-Infrastrukturen betreiben zu müssen. Es wird anhand der Umsetzung zweier prototypischer virtualisierter Forschungsumgebungen (VFU) im Bereich der Elementarteilchenphysik und der Bioinformatik diskutiert, wie zukünftige Schnittstellen zwischen Forschenden und Infrastrukturbetreibern aussehen sollten. Als Basis kommen die baden-württemberg-weit angebotenen kooperativen Forschungsinfrastrukturen des Hybrid-Clusters NEMO, der bwCloud und für den wissenschaftlichen Desktop bwLehrpool zum Einsatz. Die auf diesen Systemen eingesetzte Abstraktion durch Virtualisierung ermöglicht eine Skalierung der Ressourcen in den VFUs.

Keywords: VFU, virtualisierte Forschungs-, Lehr- und Lernumgebung, Cloud, Rechenzentrum

1 Einleitung

Digitale Datenerhebung, -verarbeitung, -verteilung und Archivierung in fast allen Bereichen der Forschung und Lehre an den Universitäten sind Kernbestandteile jeder wissenschaftlichen Tätigkeit. Diese Entwicklungen spiegeln sich in den veränderten Erwartungen an Rechenzentren wider. Mit der Verbreitung IT-gestützter Forschung und der abnehmenden Stellung von Großrechnern fand eine Verschiebung von IT-Personal in die Fakultäten statt. Sie sollten die Freiheit bei der Wahl der eingesetzten elektronischen Werkzeuge sicherstellen und Unterstützung für Forschungstätigkeiten leisten. Die Komplexität und Vielfalt der eingesetzten Systeme und Software ist in den vergangenen Jahren jedoch erheblich gestiegen und daher nicht mehr mit dem ursprünglichen Personaleinsatz zu bewältigen. Zusätzlich dazu entwickeln sich interdisziplinärer Austausch und fächerübergreifende Kollaboration zu zentralen Aspekten an heutigen Fakultäten. Entwicklungen wie virtualisierte Forschungsumgebungen ([Bu14], [Sc15]) können helfen, eine Aufgabenverteilung zwischen Forschenden und Infrastrukturanbietern zu erreichen.

[1] Physikalisches Institut, Albert-Ludwigs-Universität Freiburg, konrad.meier@rz.uni-freiburg.de
[2] Technische Fakultät, Albert-Ludwigs-Universität Freiburg, gruening@informatik.uni-freiburg.de
[3] Zentrum für Biosystemanalyse, Albert-Ludwigs-Universität Freiburg, blankc@informatik.uni-freiburg.de
[4] Rechenzentrum, Albert-Ludwigs-Universität Freiburg, michael.janczyk@rz.uni-freiburg.de
[5] Rechenzentrum, Albert-Ludwigs-Universität Freiburg, dirk.von.suchodoletz@rz.uni-freiburg.de

2 Von der Virtualisierung zur Science-Cloud

Der physikalische Ort, an dem eine Software ausgeführt wird, wird durch schnelle Netze und Virtualisierung von Rechenleistung und Storage zunehmend unwichtiger. Ein Zugriff auf geographisch entfernte Standardanwendungen, die am besten in der Nähe zu den zu verarbeitenden Daten laufen, kann heutzutage sehr performant erfolgen. Die Voraussetzungen an die lokale Hardware des Anwenders sinken. Diese Entwicklung verbirgt sich hinter dem Begriff Cloud. Kommerzielle Rechenzentren setzen bei Cloud-Diensten massiv auf die „Economies of Scale"-Effekte und übersteigen die Kapazitäten – auch großer Universitäten – inzwischen um Größenordnungen. Nach dem Schritt von der Mainframe auf den ubiquitären PC stellt die Cloud die nächste Iteration der wesentlichen IT-Innovationen dar. Nachdem eine Hardware-Innovation den ersten Umbruch auslöste, wird die zweite Umwälzung nun durch eine Software- und Service-Innovation angestoßen und stellt die Alternativlosigkeit von Hochschulrechenzentren in Frage. Auf diese externen Vorgänge sollten Rechenzentren reagieren und ihr Service-Portfolio geeignet anpassen.
Virtualisierte Forschungsumgebungen (VFU) beginnen, sich Cloud-Technologien zunutze zu machen und in den verschiedenen Wissenschaftsdisziplinen zu etablieren. Sie schaffen (virtuelle) Arbeitsplattformen, die eine kooperative Forschungs- oder Lehrtätigkeit durch mehrere Wissenschaftlerinnen und Wissenschaftler an unterschiedlichen Orten zu gleicher Zeit ohne Einschränkungen erlauben. Eine VFU unterstützt in verschiedenen Phasen des wissenschaftlichen Prozesses und kann je nach Disziplin von der Erhebung, Berechnung oder Verarbeitung von Eingangsdaten, der Simulation von Prozessen bis hin zur Generierung von Ergebnissen reichen. Vielfach sind Werkzeuge für Recherche und Bearbeitung von Texten und Referenzen zur Erstellung der finalen Publikation enthalten. VFUs konzentrieren sich auf die Softwareseite, technologisch basieren sie vielfach auf Virtualisierung oder Containerisierung. Sie können so die gesamte Breite von virtuellen Desktops über Cloud-Instanzen bis hin zu abgeschotteten Umgebungen im High Performance Computing abdecken. Sie vereinigen das disziplinspezifische Know-How in einem Paket, welches aus Sicht eines RZs als Black-Box auf geeigneten Forschungsinfrastrukturen ausgeführt werden kann. Einen ähnlichen Weg gehen Virtual Desktop Infrastructures für den Arbeitsplatz mit der Bereitstellung von Lehr- und Lernumgebungen [Ri16].
Durch die so erreichbare Trennung von Inhalt und Infrastruktur muss der wissenschaftliche Nachwuchs nicht mehr Zeit auf die Installation von Betriebssystem und Softwarepaketen ohne Garantie auf Erfolg verwenden. Die Aufgabenteilung ist hochgradig standardisierbar, vermeidet Fehler bei der Installation und ist eine Grundvoraussetzung für eine zuverlässig archivierbare und nachnutzbare VFU. Durch standardisierbare und wiederholbare Prozesse wird die Dokumentation der Arbeit der Forschenden vereinfacht. Zwar können unbemerkte Veränderungen durch Systemupgrades nicht komplett verhindert werden, sie sind jedoch zumindest nachvollziehbar dokumentiert und bei Bedarf rückholbar.
Wird eine wissenschaftliche Forschungsumgebung von Beginn an virtualisiert, wird lokale Hardware-Abhängigkeit eher vermieden und späteres Verschieben der Umgebung auf eine neue Virtualisierungsplattform deutlich erleichtert. So könnte ein Wissenschaftler eine Simulationssoftware auf dem lokalen Desktop bereits in einem Container oder virtuellen Maschine entwickeln und in diesem Schritt das Debugging erleichtern, da Algorithmen und Prozesse interaktiv mit direktem Feedback getestet werden können. Die erfolgreich

getestete Simulation kann auf eine Cloud oder in einem Cluster berechnet werden, da der Arbeitsrechner üblicherweise mit der Anzahl der Simulationsläufe überfordert ist. Die Ressourcen dafür können durch RZs in Form einer Compute-Cloud bereitgestellt werden. Virtualisierung von Beginn an stellt sicher, dass das Image der VM direkt in die Cloud-Umgebung kopiert und eine oder mehrere der Problemstellung angemessene virtuelle Instanzen gestartet werden können. Gleichzeitig kann das Image Kooperationspartnern zur direkten Verwendung oder Anpassung an die eigene Fragestellung wie auch Dritten zur Verifikation des Workflows zur Verfügung gestellt oder im Rahmen einer Lehrveranstaltung genutzt werden.
Die Vorteile solcher Ansätze sind erheblich, so dass die Abstraktion der darunterliegenden Infrastruktur (Hardware) sowie die Bereitstellung von standardisierten Zugriffsschnittstellen (APIs) von der Europäischen Kommission gefördert werden [Eu15]. Die interdisziplinäre Kooperation der Wissenschaftsbereiche durch die Bereitstellung von Diensten und Werkzeugen soll durch speziell bereitgestellte Ressourcen aller Infrastrukturbereiche (Netzwerk, Berechnungen, Daten, Software, Benutzer-Schnittstellen) vorangebracht werden. Die Kommission erwartet durch den Einsatz von virtualisierten Forschungsumgebungen eine effizientere Kooperation sowie eine gesteigerte Produktivität durch den einfacheren Zugang zu Daten.

2.1 Veränderte Rolle der Rechenzentren

Rechenzentren (und neuerdings dedizierte E-Science-Abteilungen) sollen als zentrale Einrichtungen Hilfestellung leisten und den Betrieb technischer Infrastruktur sicherstellen. Doch sind sie gleichfalls mit diesen neuen Aufgabenstellungen überfordert, da aufgrund der Breite der technischen Anforderungen, der zahllosen fachspezifischen Werkzeugen und deren Abhängigkeiten eine fallbezogene Hilfestellung nur noch in Ausnahmefällen geleistet werden kann. Beispielsweise sind die von RZs angebotenen Standarddienste wie Storage oder Server-Hosting oft zu wenig passgenau, um eine ausreichende Entlastung der einzelnen Fachdisziplinen zu gewährleisten.
VFUs müssen technisch dem Stand der Entwicklung entsprechen, bedürfen zur Entfaltung ihrer vollen Wirksamkeit jedoch der Akzeptanz in den jeweiligen Fach-Communities. Der direkte Kontakt der RZs zu den Forschenden ist notwendige Voraussetzung dafür, ihre Angebote besser zu planen und auf die Anforderungen der Forschenden bestmöglich einzugehen. Das Standardisieren und Zentralisieren von Infrastruktur am RZ entlastet Wissenschaftler von IT-administrativen Aufgaben, die beim Betreiben eigener Infrastrukturen zwangsläufig anfallen. Auf Basis der angestrebten RZ-Infrastruktur ist es möglich, VFUs zu betreiben, die speziell auf die unterschiedlichen Anforderungen der verschiedenen Arbeitsgruppen ausgelegt sind. Die Bereitstellung der Ressourcen muss dabei flexibel und zeitnah erfolgen können.
Um die notwendigen Abstimmungen und Entwicklungen gemeinsam durch Wissenschaft und Infrastrukturanbieter RZ voranzutreiben, wurde das Projekt „ViCE – Virtual Open Collaboration Environment“ ins Leben gerufen. Es entwickelt nachhaltige Geschäfts- und Steuerungsmodelle für die Kooperation von unterschiedlichsten Fach-Communities mit Rechenzentren auf Basis von VFUs, die dem dynamischen Charakter der Wissenschaften

und ihren wechselnden Anforderungen angepasst sind. Es schafft hierzu eine RZ-übergreifende Kollaborations- und Austauschplattform für virtualisierte Forschungsumgebungen, die versioniert, annotiert und geteilt werden können. Die Beschreibung der enthaltenen Tools und Workflows erlaubt die einfache Nachnutzung für neue Forschungsfragestellungen, eine schnelle Einbindung des wissenschaftlichen Nachwuchses und den Einsatz in der Lehre. Notwendige Basisinfrastrukturen der RZs werden so aufbereitet, dass sie abstrakt von verschiedenen Disziplinen einfach und ohne Startverzögerung eingebunden werden können. Hierzu wird Know-How aufgebaut, das die Wissenschaft in ihren Bedürfnissen unterstützt und eine einfache Ausdehnung auf weitere Communities erlaubt (Abb. 1).

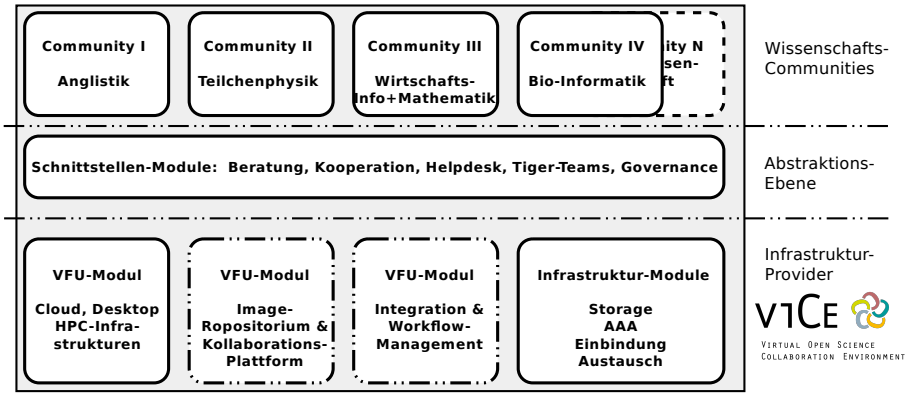


Abb. 1: ViCE sucht gemeinsam mit Infrastruktur-Providern und Wissenschafts-Communities nach einer guten Aufgabenverteilung in Nutzung und Betrieb aktueller Forschungsinfrastrukturen.

Schrittweise entstehen Best-Practices für den Betrieb von VFUs, die die Grundlagen für einen Austausch innerhalb und zwischen Standorten erlauben und perspektivisch in gemeinsamen Standards für die teilnehmenden IT-Zentren beziehungsweise deren Basisinfrastrukturen münden können. Diese orientieren sich an den Erfordernissen und Schnittstellen nationaler und internationaler Projekte und stellen so sicher, dass die Verschiebbarkeit auf breiter Ebene für weitere Fach-Communities nachnutzbar ist. Die Rekalibrierung der Aufgabenverteilung erlaubt es den Rechenzentren, zentrale und skalierende Infrastrukturen zu betreiben, und entlastet gleichzeitig die Wissenschaftler vom Betrieb eigener Infrastrukturen.

3 Virtualisierte Forschungs-, Lern- und Lehrumgebungen

Grundlegende Aufgabe beim Bereitstellen von VFUs ist eine flexible und schnelle Provisionierung von Infrastruktur-Ressourcen. Die bei der Bereitstellung entstehenden Herausforderungen gilt es zu lösen, um eine einheitliche Schnittstelle für die Ressourcenverwaltung zu etablieren. Dabei ist vor allem die Skalierung der Infrastruktur für bestehende Systeme eine Herausforderung, die durch neue Konzepte angegangen wird. Die sich schnell ändernden Anforderungen aus der Wissenschaft können sonst nicht adäquat beantwortet werden.

Abbildung 2 demonstriert, wie die Vereinheitlichung der Schnittstellen darauf aufbauende virtualisierte Forschungsumgebungen erlaubt. Das Beispiel zeigt eine HPC- und Cloud-Ressource eines Rechenzentrums. Auf beiden Umgebungen wird die OpenStack-Virtualisierungsplattform aufgesetzt, die eine OpenStack-API als Schnittstelle bereitstellt. Auch wenn die beiden Ressourcen im Wesentlichen unterschiedliche Anforderungen abbilden und unterschiedliche Betriebskonzepte aufweisen, sind sie aus Sicht der VFU homogen und ermöglichen einen einfachen Wechsel zwischen den Ressourcen ohne aufwändige Schnittstellen-Anpassungen. Ergänzt wird die HPC- und Cloud-Ressource um eine Desktop-Virtualisierung, um dem Wissenschaftler das Arbeiten in speziell angepassten, virtualisierten Desktop-Umgebungen am Arbeitsplatz zu ermöglichen. Wird eine wissenschaftliche Forschungsumgebung von Beginn an virtualisiert aufgesetzt, ist sie nicht abhängig von der darunterliegenden Hardware. Ein späteres Verschieben der Umgebung auf eine neue Virtualisierungsplattform wird somit erleichtert.

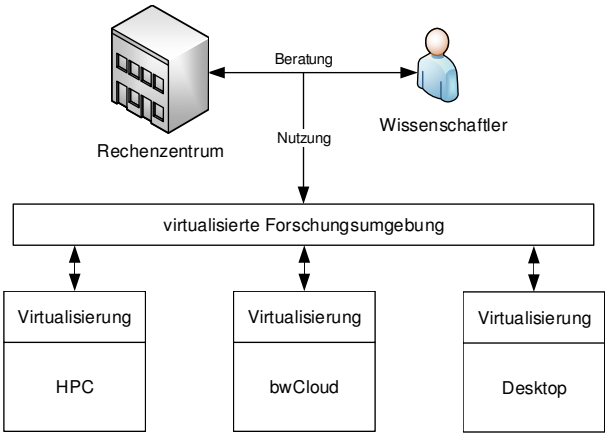


Abb. 2: Eine VFU entsteht auf dem Desktop, wird dort getestet und später zur besseren Skalierung je nach Ressourcenprofil auf das Hybrid-Cluster oder auf die bwCloud zur Berechnung verschoben.

bwCloud: Das Landesprojekt bwCloud[6] betreibt eine Infrastruktur-Cloud (IaaS) auf Basis von OpenStack für Wissenschaftler und Studierende. Das Betriebskonzept umfasst eine standortübergreifende Cloud-Plattform, die es den Benutzern erlaubt, virtuelle Maschinen selbst zu provisionieren. Dies ermöglicht es im Rahmen von Forschung und Lehre, die benötigten Ressourcen zeitnah zur Verfügung zu stellen. Die dafür notwendige Hardware wird von den Projektpartnern im Verbund an mehreren Rechenzentren betrieben [DSS15].

bwHPC: Für rechenintensive VFUs sind die Ressourcen einer Cloud-Umgebung nicht immer ausreichend. Rechencluster aus dem High Performance Computing (HPC) stellen die nächste Skalierungsstufe dar und sind für diese Fragestellungen ausgelegt. Den Wissenschaftlern in Baden-Württemberg stehen im Rahmen des bwHPC-Konzepts hochmoderne HPC-Systeme aller Leistungsklassen zur Verfügung [HWC13]. Das vom Rechenzentrum

[6] bwCloud Projektwebseite: http://www.bw-cloud.org [letzter Aufruf 11.3.2017]

der Universität Freiburg entwickelte neuartige Hybrid-Cluster-Konzept[7] gestattet es, in virtuellen Maschinen auf einem HPC-System zu rechnen. Das ermöglicht es VFUs auf einem HPC-Cluster auszuführen, ohne die darunterliegende Hardware des HPC-Systems zu partitionieren [Me16].

bwLehrpool: Die durch bwLehrpool[8] angebotene Desktop-Virtualisierung kombiniert auf Grundlage eines einzigen Basissystems ein flexibles Angebot verschiedenster Betriebssystemumgebungen mit einer einfachen Administrierbarkeit großer PC-Landschaften [Ri16]. Labor-, Lehr- und Lernumgebungen müssen auf diese Weise nicht mehr auf den Arbeitsplatzrechnern installiert sein, was den Wartungsaufwand der IT-Administratoren erheblich reduziert sowie Lehrenden und Forschenden vollkommen neue Gestaltungsmöglichkeiten der Lehre einräumt. Sie können Arbeitsumgebungen in einem weiten Spektrum selbst gestalten und aufgrund der Abstraktion des Systems sogar einrichtungsübergreifend austauschen. Das erlaubt Hochschulen, auf aktuelle Entwicklungen schnell zu reagieren, eine deutlich flexiblere Nutzung ihrer vorhandenen Ressourcen zu erreichen und die zentralen IT-Dienste von repetitiven Standardaufgaben zu entlasten.

3.1 VFU – Anwendungsfall Experimentelle Teilchenphysik

Aktuelle sowie zukünftige Experimente der Hochenergiephysik (HEP) generieren enorme Datenmengen. Um diese Daten in angemessener Zeit zu speichern, zu verarbeiten und zu analysieren, sind viele Rechenressourcen notwendig. Zusätzlich zu den aufgezeichneten Daten sind Monte-Carlo-Simulationen in gleichem Umfang erforderlich, um Beobachtungen aus dem Teilchenbeschleuniger mit theoretischen Modellen zu vergleichen.
In der Vergangenheit wurde die benötigte Rechenleistung von speziellen HEP-Clustern bereitgestellt. Die jeweils erforderlichen Betriebs- und Softwareumgebungen konnten auf den Clustern ohne Probleme installiert werden, da diese exklusiv zur Verfügung standen. Die Rechenaufgaben wurden aufgrund ihrer Unabhängigkeit in kleine Compute-Jobs aufgeteilt und gleichmäßig auf die HEP-Cluster verteilt. Eine gleichzeitige Ausführung oder eine Interprozesskommunikation (IPC) war nicht erforderlich. Durch die lose Koppelung der Compute-Jobs zählen sie zum High Throughput Computing (HTC). Die HEP-Cluster wurden dementsprechend mit geeigneten Schedulern und Infrastrukturen betrieben, um diese Compute-Jobs effizient zu verarbeiten.
Demgegenüber stehen Forschungs- und Universalcluster, die eine größere Nutzergruppe und damit andere Anforderungen bedienen müssen. Diese Cluster müssen Compute-Jobs ausführen, die auf mehreren Rechenknoten parallel laufen und damit einen Cluster-Interconnect mit geringer Latenz für die IPC benötigen. Diese Cluster zählen zur Klasse des High Performance Computings (HPC). Betriebssystem- und Softwarekonfigurationen unterscheiden sich teilweise deutlich voneinander. Sie sind darauf optimiert, die spezialisierte Hardware für IPC optimal zu unterstützen. So werden beispielsweise spezielle Compiler und Bibliotheken für das HPC-Cluster angeboten. Im Gegensatz dazu setzen die meisten

[7] Das Konzept des Hybrid-Clusters wurde erstmals als Poster auf dem 2. bwHPC Symposium 2015 in Ulm vorgestellt; http://www.uni-ulm.de/index.php?id=69869 [letzter Aufruf 3.4.2017]
[8] bwLehrpool Projektwebseite: https://www.bwlehrpool.de [letzter Aufruf 11.3.2017]

HEP-Experimente auf eine genau definierte Softwareumgebung inklusive Betriebssystem und Software aus einem zentralen Software-Repositorium. Nur so kann die Vergleichbarkeit der Ergebnisse garantiert werden, die auf unterschiedlichen Clustern berechnet werden. Darüber hinaus wird die HEP-Softwareumgebung in regelmäßigen Abständen aktualisiert und muss zeitnah auf den Clustern verfügbar sein. Das kann auf Clustern, die nicht dediziert von der HEP-Community administriert werden, nicht gewährleistet werden.
Diese Ausgangssituation macht es für HEP-Anwender schwierig, die Ressourcen von HPC-Clustern zu verwenden. Da diese Cluster jedoch einen signifikanten Anteil zu den benötigten Rechenressourcen beitragen können, wurde ein Betriebskonzept entwickelt, das es erlaubt, VFUs auf einem HPC-System zu starten. Die Ressourcen stehen für mehrere wissenschaftliche Fachbereiche und damit nicht exklusiv für einen Forschungsschwerpunkt bereit.
Der entwickelte Ablauf ist in Abbildung 3 dargestellt und zeigt, wie die VFU basierend auf dem Hybrid-Cluster-Konzept umgesetzt wurde. Dabei übermitteln Wissenschaftler ihre Compute-Jobs wie gewohnt an HTCondor. Daraufhin skaliert der Resource-Broker ROCED bedarfsabhängig und automatisiert die VFUs. HTCondor ist dabei ein Scheduling-System, das Priorisierung sowie Ressourcen-Monitoring und -Management erlaubt. ROCED (Rapid On-Demand Cloud-enabled Deployment) wird wiederum verwendet, um HTCondor zu überwachen, um je nach Ressourcenbedarf dynamisch virtuelle Maschinen zu starten oder zu stoppen.

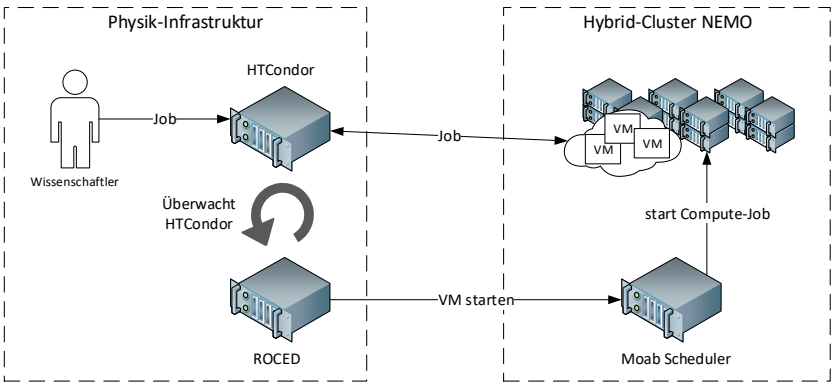


Abb. 3: Ein Resource-Broker skaliert bedarfsorientiert das Hybrid-Cluster nach Job-Übermittlung.

Das Setup zeigt eine Möglichkeit, wie spezielle Softwareumgebungen auf einem HPC-System ausführt werden können und wie die Aufgabenverteilung zwischen Wissenschaftlern und Rechenzentren in der Praxis aussehen kann. Das Bereitstellen der virtualisierten Infrastruktur (Hybrid-Cluster NEMO) wird vom Rechenzentrum übernommen. Die VFU wird vom Wissenschaftler erzeugt und nutzt die vom Rechenzentrum bereitgestellten Ressourcen. Im Fall der Teilchenphysik handelt es sich dabei um eine komplexe VFU, die unterschiedliche Ressourcen gleichzeitig verwendet. So laufen rechenintensive virtuelle Maschinen auf dem Hybrid-Cluster, während die für die Umgebung benötigten Dienste (HTCondor und ROCED) auf der bwCloud-Infrastruktur ausgeführt werden.

3.2 VFU – Anwendungsfall Bioinformatik

In der der Bioinformatik in Freiburg wird seit vier Jahren eine fortwährend wachsende Galaxy-Instanz betrieben, wodurch nicht nur die für die Wissenschaftler notwendige Hard- und Software verfügbar gemacht wird, sondern ebenso umfangreiche Metadaten über die technischen Anforderungen gesammelt werden können. Job-Logs mit detaillierter Aufzeichnung von Job-Metriken wie dem Speicherbedarf, der CPU-Leistung und der Laufzeit erlauben eine umfangreiche Analyse des Bedarfs sowie des Optimierungspotentials. Durch stetig wachsende Nutzerzahlen, Datenmengen und wachsender Komplexität der Software zeigen sich hierdurch Grenzen des bisherigen Setups, welches auf einem eigenen Cluster der Bioinformatik läuft und über die Sun Grid Engine (SGE) angesprochen wird.
Die Galaxy-Plattform ist ein auf Unix-Maschinen lauffähiges Software-Framework für das Management von wissenschaftlichen Workflows und Pipelines, für die Datenverarbeitung und für die Durchführung von Rechenjobs. Motiviert insbesondere durch die technischen Probleme, die mit dem Next Generation Sequencing (NGS) und der damit einhergehenden High-Throughput-Datenerzeugung aufkommen, ist es das Ziel, aus den großen Datenmengen möglichst viel wissenschaftliche Information zu gewinnen. Dies bedarf komplexer statistischer und algorithmischer Methoden, die nur aufgrund großer zur Verfügung stehender Rechenleistung realistisch geworden sind. Das führt zwangsweise zu Problemen in biologischen Fachbereichen, da hierdurch Wissenschaftler ohne einen informationstechnischen Hintergrund auf hochkomplexe computergestützte Analysen angewiesen sind. Das seit 2005 bestehende Galaxy-Projekt nutzt ein einfach zugängliches Webinterface, um unabhängig vom IT-Hintergrund Zugang zu umfangreichen Tools und den benötigten technischen Ressourcen zu ermöglichen und um komplexe Analysen durchzuführen.
Am Beispiel von Galaxy wird in der Softwarelandschaft der Bioinformatik eine Infrastruktur aufgebaut, die für den Nutzer das Handling auf Seiten der Software enorm verbessert. Darüber hinaus werden mittels Virtualisierung und Containerisierung die Hardware-Abhängigkeiten auf ein Minimum reduziert. Dadurch gelingt es, Forschenden abgestimmte, virtualisierte Forschungsumgebungen anzubieten, welche sowohl auf einem Desktop als auch in der Cloud und im HPC laufen [Af16].
Hierzu gibt es zusammen mit der Bioinformatik-Community Bestrebungen, verfügbare und zukünftige Software in einer Paketverwaltung (Conda) einzugliedern. Diese läuft im Userspace und erlaubt es, automatisiert, transparent und reproduzierbar vorkompilierte Pakete zu erstellen, welche sich in einem weiteren automatisierten Schritt zusätzlich in verschiedene Container-Formate übersetzen lassen. Dies macht sowohl eine Versionierung, als auch einen Austausch von Forschungsdaten inklusive deren Reproduzierung und die automatische Erfassung von Metadaten bezüglich eines verwendeten Workflows und dessen Tools einfach.
Aufgrund von wachsender Nachfrage und der angestrebten Nutzung, den bisher auf Freiburg und Umgebung beschränkten Dienst national und international zu öffnen, ist eine flexible Nutzung bereitgestellter Ressourcen notwendig. Dies erlaubt, auch auf zukünftige Veränderungen reagieren zu können und mit dem Dienst weiter zu skalieren.

In Abbildung 4 ist das entwickelte Setup der Freiburger Galaxy-Installation zu sehen. Zusätzlich zum bisherigen Cluster der Bioinformatik können nun über HTCondor Rechenjobs verwaltet werden, um sie unter anderem auf dem Hybrid-Cluster NEMO oder

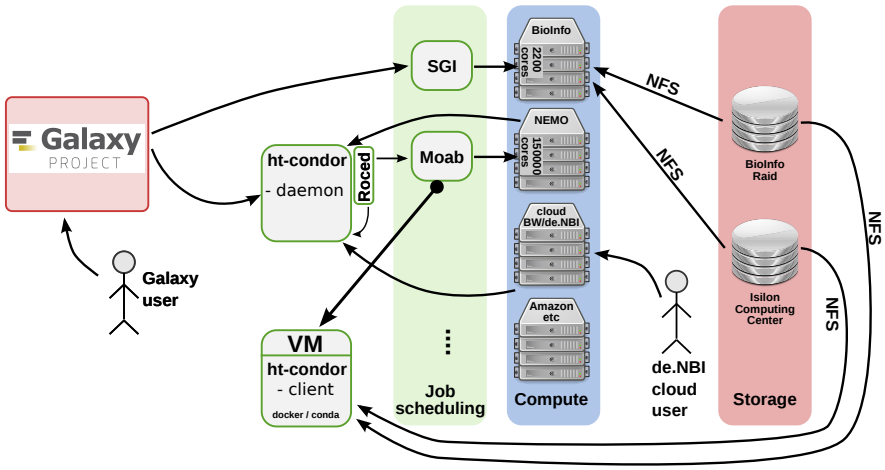


Abb. 4: Die Freiburger Galaxy-Instanz entscheidet je nach Workflow, ob eine Berechnung auf dem Bioinformatik-Cluster (SGI) oder in der Bioinformatik-VFU erfolgen soll.

auf Compute-Clouds wie der bwCloud oder de.NBI-Cloud zu starten. Dies erlaubt ein Setup, das weitgehend unabhängig von der zugrundeliegenden Hardware-Infrastruktur ist. Dadurch können grundsätzlich auch weitere und kommerzielle Anbieter genutzt werden (Eucalyptus, OpenNebula, OpenStack, Amazon EC2, etc.). Anhand der in Galaxy gesammelten Job-Metriken wurden Anwendungsprofile erstellt, die Berechnungen automatisch auf geeignete Hardware schicken. Für die Nutzer ändert sich dadurch nichts. Sie können weiterhin mit dem Webfrontend arbeiten, das sie auch von einer lokalen Galaxy-Instanz auf dem Desktop kennen. Die Verwaltung der Tools in einem Paketmanager und die Virtualisierung oder Containerisierung stellen sicher, dass das wissenschaftliche Arbeiten flexibel auf jeglicher Hardware transparent und mit reproduzierbaren Ergebnissen durchgeführt werden kann.

4 Fazit

Eine zukünftige IT-Strategie moderner Hochschulrechenzentren konzentriert sich neben der Grundversorgung auf notwendige Forschungsinfrastrukturen. Da diese einerseits von „Economies of Scale“-Effekten profitieren und damit andererseits eine attraktive Vielfalt an Diensten angeboten werden kann, bieten sich hier kooperative Ansätze an. Virtualisierte Forschungsumgebungen helfen dabei, die Ortsabhängigkeit aufzulösen und gemeinschaftlich zentrale Dienste anzubieten. Mit dieser Sichtweise erlauben Virtualisierung und Compute-Clouds eine neue Aufteilung der Aufgaben zwischen Wissenschaftlern, Instituts-IT und dem Rechenzentrum. Das Rechenzentrum stellt notwendige Basis- und Forschungsinfrastrukturen, angefangen vom Netz bis hin zu hochwertigen Storage-, Servervirtualisierungs- und HPC-Diensten zur Verfügung. Auf Basis dieser können Forschende ihre eigenen Umgebungen und Methoden frei realisieren. Sie werden dabei zusätzlich durch Angebote zu automatischem Backup und Beratung zum Forschungsdatenmanagement un-

terstützt.
Der Übergang vom klassischen wissenschaftlichen Rechenzentrum der 1990er zu einem Rechenzentrum für die Wissenschaft kann oft nicht aus den vorhandenen Ressourcen gestemmt werden. Um solche Angebote erfolgreich weiterentwickeln zu können, muss sich ein Rechenzentrum auch an Forschungs- und Infrastrukturbegleitprojekten beteiligen. Diese Neuausrichtung ist längst nicht abgeschlossen und wird anhalten, so lange es gesellschaftliche, politische, hochschulspezifische und technische Veränderungsprozesse gibt. Rechenzentren müssen sich diesen Herausforderungen stellen und Weiterentwicklungen meistern: „Ohne ständige Innovationen an einem IZ/RZ wird es obsolet“[9].

Literaturverzeichnis

[Af16] Afgan, Enis; Baker, Dannon; van den Beek, Marius; Blankenberg, Daniel; Bouvier, Dave; Čech, Martin; Chilton, John; Clements, Dave; Coraor, Nate; Eberhard, Carl; Grüning, Björn; Guerler, Aysam; Hillman-Jackson, Jennifer; Von Kuster, Greg; Rasche, Eric; Soranzo, Nicola; Turaga, Nitesh; Taylor, James; Nekrutenko, Anton; Goecks, Jeremy: The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. Nucleic Acids Research, 44(W1):W3–W10, 2016.

[Bu14] Buddenbohm, Stefan; Enke, Harry; Hofmann, Matthias; Klar, Jochen; Neuroth, Heike; Schwiegelshohn, Uwe: A life cycle model für collaborative research environments. In (Müller, Paul; Neumair, Bernhard; Reiser, Helmut; Rodosek, Gabi Dreo, Hrsg.): 7. DFN-Forum - Kommunikationstechnologien, 16.-17. Juni 2014, Fulda, Germany. Jgg. 231 in LNI. GI, S. 1–10, 2014.

[DSS15] Dulov, Oleg; Scheibenberger, Klaus; Schulz, Janne Chr.: bwCloud – Standortübergreifende Servervirtualisierung. SCC-News, 01/2015, 2015.

[Eu15] European Commission: EINFRA-9-2015 - e-Infrastructures for virtual research environments (VRE). http://cordis.europa.eu/programme/rcn/664625_en.html, 2015.

[HWC13] Hartenstein, Hannes; Walter, Thomas; Castellaz, Peter: Aktuelle Umsetzungskonzepte der Universitäten des Landes Baden-Württemberg für Hochleistungsrechnen und datenintensive Dienste. Praxis der Informationsverarbeitung und Kommunikation, 36(2):99–108, 2013.

[Me16] Meier, Konrad; Fleig, Georg; Hauth, Thomas; Janczyk, Michael; Quast, Gürnter; von Suchodoletz, Dirk; Wiebelt, Bernd: Dynamic provisioning of a HEP computing infrastructure on a shared hybrid HPC system. Journal of Physics: Conference Series, 762(1):012012, 2016.

[Ri16] Ritter, Steffen; Trahasch, Stephan; Slotosch, Sven; von Suchodoletz, Dirk; Münchenberg, Jan: bwLehrpool: Durchführung von elektronischen Prüfungen in virtualisierten Umgebungen. In (Lucke, Ulrike; Schwill, Andreas; Zender, Raphael, Hrsg.): DeLFI 2016 - Die 14. E-Learning Fachtagung Informatik, 11.-14. September 2016, Potsdam. Jgg. P-262 in LNI. GI, S. 149–154, 2016.

[Sc15] Schwarzkopf, R.: Virtual machine lifecycle management in grid and cloud computing. Philipps-Universität, 2015.

[9] Stefan Wesner, Keynote auf der 17. ZKI-Herbsttagung, https://www.uni-ulm.de/fileadmin/website_uni_ulm/zkiherbst2016/praesentationen/dienstag/wesner_gestaltung_von_innovationsprozessen_an_rechen_und_informationszentren.pdf, S. 41