ISSN: 2182-2069 / E-ISSN: 2182-2077

A Comprehensive Study on the Dimensions of IaaS Security

Mihály Héder[1*], Domonkos Baczó[2], Tibor Kovács[3], and Ernő Rigó[4]

[1*]Department of Network Security and Internet Technologies at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN), Hungary.
mihaly.heder@sztaki.hun-ren.hu, https://orcid.org/0000-0002-9979-9101

[2]Department of Network Security and Internet Technologies at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN), Hungary.
domonkos.baczo@sztaki.hun-ren.hu, https://orcid.org/0009-0008-0580-8873

[3]Department of Network Security and Internet Technologies at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN), Hungary.
tibor.kovacs@sztaki.hun-ren.hu, https://orcid.org/0009-0006-6245-6523

[4]Department of Network Security and Internet Technologies at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN), Hungary.
erno.rigo@sztaki.hun-ren.hu, https://orcid.org/0000-0003-1044-7167

Received: March 02, 2024; Revised: April 30, 2024; Accepted: June 07, 2024; Published: August 30, 2024

Abstract

In this paper we conduct a systematic exploration of the security considerations of Infrastructure-as-a-Service (IaaS) cloud deployments. These deployments are very common in the landscape of Research and Ecudation as well as the corporate world. The aim of this work is to provide an exhaustive list of concerns that can help both researchers and deployers of such systems. The organizing principle of our study is the architecture of the typical IaaS cloud. Here we identify three major layers: physical infrastructure, cloud middleware and virtual infrastructure. We also consider auxiliary element, the operations centre. In the physical layer we explore the questions of disaster recovery, and covert channel and side channel attacks, mostly exploiting hardware vulnerabilities. In the middleware we investigate the general cryptography of the IaaS as well as Trusted Computing. In the virtual layer, we present the most common issues in Virtual Machine handling, handling of updates and handling of malicious insiders. Regarding the operations centre we investigate the questions of monitoring, network management, the issues with the API's and the user portal, with special attention to authorization and permission management. Running IaaS clouds in a secure way requires a large team comprised of people with various technical backgrounds. Based on our experience with running a national-level IaaS cloud we found that most of the security researchers focus on one specific issue. We believe an important value of our contribution is the synthesis of all the relevant dimensions, with special attention to various, often overlooked aspects of the IaaS deployment.

Keywords: IaaS, Trusted Computing, Virtual Machines, Cloud.

Journal of Internet Services and Information Security (JISIS), volume: 14, number: 3 (August), pp. 116-142.
DOI: *10.58346/JISIS.2024.I3.007*

*Corresponding author: Department of Network Security and Internet Technologies at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN), Hungary.

1 Introduction

The economic forces, as well as environmental concerns around modern ICT all point toward the same direction - the consolidation of ICT users to shared infrastructure. The reasons for this are the uneven patterns of usage, the economies of scale and the possibility of services being delivered over a network and the optimal leverage of skilled ICT operators. These factors mean that any cost analysis will indicate that big, multi-tenant infrastructures are the most effective.

This, however, means that valuable targets are created in the sense that many sensitive tenants and their data may be accumulated on a single system. Moreover, it also means that there is a need for strong separation between tenants.

These are unique security characteristics of cloud systems that are complemented with the security challenges that any system faces that is connected to the network.

In this article we aim to provide an overview of these challenges and the possible remedies for them. We aim to create an actionable review, therefore, our main goal is to create a checklist based on the review. Following the checklist, while does not guarantee ultimate security of a cloud, it may help to cover all the common issues.

2 Methodology

The literature analysis was conducted in several steps. First, the potential search keywords were determined, then with these keywords searches were made in multiple scientific databases. The search results were filtered, both automatically and manually, and for the relevant articles excerpts were written. The next step was an evaluation of the selected articles, based on how useful they are for this research. For the effective categorization of the checklist items, the keywords of the papers were gathered, and then they were categorized according to the cloud systems they were involved with. The final step was the concretization of the checklists, and the fine tuning of the created categories.

The selection of the search keywords was a critical aspect of the research. The volume of potential results were immense, thus it was important not to use broad search terms or a low number of keywords. We agreed that “cloud”, “security” and “IaaS” were the most essential search terms, and we used them in every query. In the first phase of the literature gathering we used additional terms that directly correlated with the topic of this paper, e. g. “checklist”, “guideline”, “guide”, “policy” and “assessment”. In the next phase, we used additional keywords that are more specific to different subparts of the IaaS cloud e. g. “web interface”, “API”, “validation”, “mitigation”, “nullification”, “log”, “virtual machine”, “virtual image”, “testing”, etc. This way we could find cloud security measures that are more specific to a process or part of the cloud infrastructure, while being useful for a large number of IaaS systems.

The queries were made using multiple scientific databases. The majority of the queries were conducted using Scopus, the abstract and citation database of Elsevier. The reason for this was that Scopus provided a reasonable number of results that were mostly relevant and could be reviewed manually. We also used Web of Science (from Clarivate) for a number of queries, however the number of results were lower, and most of them were included in the set of results provided by Scopus. Another possibility was using Google Scholar, however it provided a vast amount of results that were impossible to manually review, while being efficient. Thus we only used Google Scholar in cases where the other databases did not provide the source for a result.

A limitation of our work is that we did not engage with the user-side and psychological aspects of IaaS security as it would have required a significant amount of theoretical background and explanations, making the article too long.

The most efficient tool for the research was Elicit, developed by Ought. Elicit is an AI research assistant that can find articles related to a question, or even provide alternative questions that might help the research. Elicit does not require perfect keyword matches, and the questions asked in natural language provide more context than a set of keywords. For example, Scopus often discarded one or two keywords from the query that lead to irrelevant results or "misinterpreted" polysemous terms. Elicit can also summarize articles that helped in deciding whether a paper is relevant for the research.

The query results were filtered in the databases, only articles from the last 5 years were used, in order that the checklist does not contain any obsolete information. Then the papers were manually filtered. In most cases the abstract summarized the contents of the article sufficiently, thus the relevancy of the papers could be determined by examining it. There were precedents where the abstract was broad or ambiguous, in these cases we reviewed the article itself. We excluded articles that were not free to access due to resource limitations. After filtering, we read the selected articles (92 in total), and created brief excerpts to summarize their content. The excerpts included the most important concepts, and ideas relevant to the research. We also provided evaluation for the usefulness of the papers and noted if the quality was in the extremes to aid future categorization.

The next step was to quantify the evaluation. We marked the articles on a scale of 1 to 5, in two subjects - cloud security checklist and cloud security threats. The latter became a parallel research, since we found a large number of articles that contained detailed information about vulnerabilities, threats and attacks against cloud computing systems. The marks were given according to the following guide:

- 5 - the article is entirely about the subject.
- 4 - the article contains multiple concepts that are relevant to our article.
- 3 - the article contains at least one concept that is relevant to our article.
- 2 - the article describes a framework or practical system, but it is a specific solution (can be featured as an example).
- 1 - the article is irrelevant or has extremely low quality - this means that the article should not be used.

In some cases, the evaluation was modified according to the level of detail, the generality of the subject and the overall quality of the article. Where these alterations occurred, we provided comments to justify them. We also flagged articles that contained topics that needed further investigation e. g. one article mentioned the MITRE ATT&CK matrix, that we also reviewed later.

With this method we could get a general idea of the most covered topics of the research. However, this method was insufficient for a logical categorization of the sources (and the checklists). We concluded that using the architecture model of the IaaS we could achieve a systematic overview of the most important security measures. The next step was to establish the main components of the architecture (will be detailed in this paper later), which were also the categories for the processed articles. Then we categorized the sources, allowing that one paper could fit into multiple categories. We added more categories for a more accurate classification, such as articles about checklists or general countermeasures, papers that are useful for the introduction and sources for interesting, albeit not strictly relevant, related topics. After this categorization, we could finally establish thematic checklists, which we fine tuned multiple times in order that every item is located in the most relevant

category. We also provided a list of practical solutions for each checklist that we found in the reviewed articles. We also based our results on research done in 2016 titled *Security checklist for IaaS cloud deployments (*unpublished*, available on arXiv).*

1. IaaS Cloud Architecture

Figure 2. presents the typical IaaS cloud architecture. This diagram shows only one data center in a cloud setup for brevity. In reality there might be several interconnected data centers. The nature of the connections between them is important when one data center is compromised, as it will be explained later.

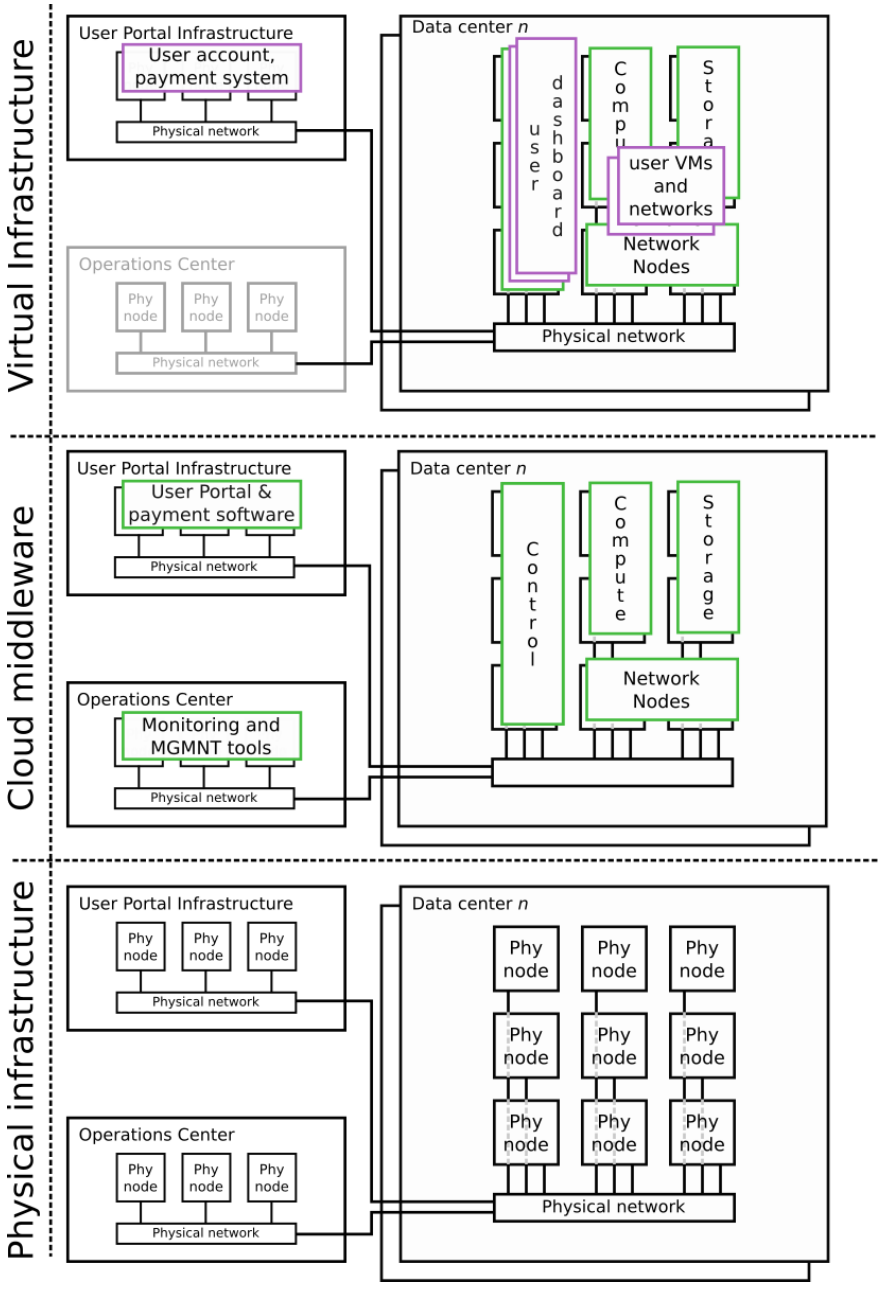


Figure 2: A typical IaaS Cloud Architecture

2. Elements of an IaaS Cloud

The essence of an IaaS cloud is to provide separate virtual infrastructures for the clients while sharing a physical infrastructure. This will result in the better utilization of the costly physical infrastructure and is therefore saving money and energy. However, the shared physical infrastructure also poses security challenges.

- **Physical Infrastructure**

The physical infrastructure of an IaaS cloud usually consists of Layer-1-2-3 physical network infrastructure, compute nodes with powerful CPUs, plenty of RAM and optionally GPUs installed, physical nodes optimized to networking, physical nodes optimized for storage, meaning usually that they contain many disks, and nodes optimized for controlling the other nodes.

Additional to these elements, there is need of a network/security operations center, which is responsible for the monitoring and controlling of the IaaS cloud. This is the gate through which the operations team accesses the IaaS, therefore if this infrastructure is compromised that usually means that the whole cloud is compromised.

- **IaaS Middleware**

The IaaS middleware, e.g. OpenStack is a modularized software, where the modules are responsible for providing the elements of the virtual client infrastructure.

This means that there is a component running on the compute nodes, there is a storage subsystem, a network subsystem. Additionally, there are control modules that should run on dedicated hardware. Another subsystem is responsible for the user dashboard through which the user can manage the its IaaS cloud.

- **Virtual Resources**

The virtual resources provided by an IaaS cloud are intended to replace owned physical infrastructure. Therefore, the features of a Virtual Machines and Virtual Networks are supposed to be the same as their physical counterparts.

Moreover, different user's resources are supposed to be isolated from each other. One area of attacks on IaaS clouds is on the isolation, both in terms of confidentiality and integrity.

- **Operations Center Software and User Portal**

The operations center usually runs monitoring software, configuration and change management, hardware checks and measurements. Most often these tools are web- based, and therefore in theory yield a significant attack surface, which is countered by limitations on the networks that can access it.

The network filtering strategy is rarely possible for the User Portal though. This portal manages the user account, is responsible for presenting the user the bills (even though those might be produced on the control nodes) and for the payment. Also it might double as the selling point and promotional site for the cloud. For certain clouds it is possible to limit access to this part of the system on the network level, but in general the users will require this to be public.

- **Considerations on the Layered Approach**

It is essential to consider the holistic perspective of security measures in an IaaS environment. While the segmentation into three principal levels and an auxiliary element is valuable, understanding the interplay and impact of security measures across these layers is crucial for ensuring comprehensive security.

The integration of security measures across layers is vital to establish a robust security posture in IaaS environments. For instance, the integrity of logs, which is essential for security monitoring and incident response, can be influenced by security measures implemented at the virtual infrastructure layer. Additionally, the usage of monitoring coupled with machine learning algorithms for anomaly detection can provide insights into potential security threats that may transcend multiple layers of the infrastructure.

Moreover, the implementation of encryption techniques, such as Elliptic Curve Cryptography (ECC), should be considered across all layers to ensure data confidentiality and integrity. By addressing security concerns holistically and considering the interconnectedness of layers, IaaS users can enhance the overall security posture of their environments and mitigate potential risks effectively.

In conclusion, while the segmentation into distinct layers is valuable, it is imperative to acknowledge the interconnectedness of these layers and the potential influence of security measures across them to establish a comprehensive and robust security framework in IaaS environments.

3. Physical Infrastructure

The physical infrastructure is the foundation the entire system is built on, thus it is essential to protect it against attacks. A cloud system's physical infrastructure can be attacked with any method as a conventional computer system, but there are also cloud-specific attack vectors. We focused more on the latter, and provided more IaaS-specific measures. We grouped them into three categories, and provided checklists for physical resources, cryptography and trusted computing.

- **Physical Resources**

Physical resources, such as CPUs, GPUs, memory and storage devices can be targeted in IaaS cloud systems in similar ways to an on-premise system. However, for a cloud service provider there is less room for error, since they need to maintain the trust of their customers, who do not have direct access to their data and virtual devices. Because of this there need to be additional security measures, in order to protect the system and also prove trustworthiness to the customers.

As in every computing system, the maintenance of hardware is essential. Regular checkups should be conducted on every device (Aburukba et al., 2022), both physically (ensuring physical integrity) and using diagnostic tools. Any malfunctional device should be replaced as it not only compromises the performance of the system, but can increase the attack surface.

In case of a natural disaster multiple devices can be damaged or malfunction. To provide high availability these should be resolved quickly. Also, it is essential that user data and virtual machines are recovered. Thus a multi-failure disaster recovery system should be implemented (Halabi et al., 2018).

A significant part of the recovery capability is backup. Backup can be expensive, but data loss should be avoided, thus a good tradeoff should be made. A sufficient backup and recovery plan is essential, the number of distributed backup locations and the amount of data stored on each location

should be carefully considered (Sahu & Nene, 2021). The backup itself can also be vulnerable and can be the target of attacks such as unauthorized access and tampering, thus the security of the backup is also an important part of providing availability. The backups should be encrypted and also should be distributed to multiple locations (though the service provider should be transparent about these locations) (Kumar & Goyal, 2019). A cautionary case is Capital One, a financial institution which, because of cloud misconfiguration, suffered a data breach 2019. Hackers gained unauthorized access to the company's systems and encrypted critical data, including operational data and backup information. While the specific details of Capital One's backup strategy are not public, the incident underscores the importance of distributed backups with strong security controls (Capital One Data Breach Compromises).

Logs are also a significant element of any computing system. Logs contain sensitive information that can aid investigations and can also be used as evidence in forensic operations. The integrity of the logs is essential, however attackers might want to erase evidence and target logs, either with the goal of deleting or modifying them. Logs should be encrypted and published to multiple locations on the cloud (Rao et al., 2019). The latter measure ensures that the logs are preserved and any modifications can be detected. Rao et al., (2019) proposed a secure logging system where logs are encrypted using AES and also published at three places on the internet, in order that the logs stored in the database can be verified by investigators.

Sometimes the protection of stored data is not per se enough, the CSP needs to prove the integrity of said data to their users. One possibility to prove that the data is available and has not been tampered with is Proof-of-Retrievability (PoR). This cryptographic formulation enables a remote audition of data integrity, without disclosing a copy of the original file. El Balmany et al., (2022) proposed a trusted zone-based storage method for virtual machine images that provides PoR. Kumar & Goyal, (2019) also mentions models that include PoR. It's important to note that PoR doesn't guarantee complete security against all threats. However, it provides a valuable tool for users to hold CSPs accountable for data integrity. Additionally, implementing PoR can be complex and requires careful integration with cloud storage systems.

Multi-tenancy is common in cloud computing, in several cases tenants share the same physical resources. If a user deletes their data or destroys a virtual machine, the previously used resources might be used by another tenant. Because of this it is essential that none of the data of the previous user can be recovered, since it might contain sensitive information. It is important that storage is sanitized completely before release, and disk blocks are never reassigned without nullifying or overwriting previous data (e. g. with a template image). Volatile data stores such as RAM should be nullified on the start of a new VM. Encryption of disk and anonymization can also be a solution for this problem, since the new VM cannot read back the data without the encryption key. In 2020, Microsoft acknowledged a vulnerability in their Azure Blob Storage service that potentially exposed customer data under certain conditions. The issue stemmed from insufficient sanitization of storage accounts when a customer deleted their data. This incident underscores the critical importance of data sanitization in multi-tenant cloud environments (Microsoft Data-Exposure).

One way to prove to the users that their data has been deleted is proof of deletion or proof of secure erasure, that is a remote attestation method. Kacha & Zitouni, (2018) mentions that it is hard to prove the deletion of data, since multiple copies of the data might exist, thus it is often a problem of trust. One potential challenge is the verification of data deletion across distributed backup locations. Ensuring that data is effectively erased from all backup repositories can be intricate, especially when considering the replication of data across multiple sites for redundancy and disaster recovery purposes.

Coordinating the deletion process across these distributed locations while maintaining data integrity and security presents a significant challenge that needs to be addressed. In terms of possibilities, advancements in remote attestation methods offer potential solutions for verifying data deletion securely. Leveraging innovative techniques like behavior-based remote attestation can enhance the integrity of the deletion process by providing verifiable evidence of data erasure. By adopting sophisticated remote attestation mechanisms, organizations can instill greater confidence in users regarding the thoroughness and effectiveness of data deletion procedures.

Namboodiri et al., (2023) explores using remote attestation techniques for secure data deletion in cloud storage environments. It discusses the limitations of traditional methods and proposes a framework that leverages remote attestation to ensure verifiable data erasure.

The vulnerabilities and predictability of computer architectures can be exploited by side-channel attacks and covert channel attacks. Side-channel attacks are based on accidental information leakage that is caused by implementation of known protocols and algorithms. One of the most common types is cache-based attacks. Parast et al., (2022) proposes the use of CPU-integrated solutions (SGX for Intel and ARM TrustZone) and turning off S-Box access. Against information leakage it advises the use of partitioned cache or partition-locked cache. Tao et al., (2021) proposes a novel system to mitigate cache-based side-channel attacks, called SCAMS. It is based on capturing vulnerable cryptographic operations, using a proactive notification system for them, and monitoring cache anomalies during these vulnerable operations. Guo et al., (2023) details a cache-based side-channel attack that could break real-world RSA encryption implementations. The attack exploited the predictability of cache access patterns to recover the secret key used for decryption.

Covert channel attacks differ from side-channel attacks, since these are based on infected systems. These attacks allow tunneling information between systems that are not allowed by policies to do so. Semal et al., (2020) conducted a study on microarchitectural covert channel attacks that exploit the vulnerabilities of the architecture of a processor. Apart from expensive new architectures that ensure better isolation, the paper proposed the disabling of simultaneous multi-threading, which mitigates some of the vulnerabilities. However, doing so reduces efficiency of IaaS systems. Another solution can be to provide isolated instances for security-critical systems, though it can be an expensive solution.

DDoS (Distributed Denial of Service) attacks are dangerous, since massively overusing the system could lead to hardware failure in certain cases, and they are quite frequent in cloud systems, so it is important to mitigate them. One solution to this is limiting the usage of certain services and sending warnings to the provider in case of out of pattern usage, so these kinds of attacks are recognised in time and held back from doing any major damage (Parast et al., 2022). We list this at the physical layer but to some extent this is a layer-independent phenomenon.

4. In the Middleware

- **Cryptography**

Cryptography is really important to take into consideration, for the obvious reason that a weak or non-existent cryptography method could lead to classified data reaching someone or some organization which should not have access to it. When an adversary with malicious intent gets hold of private data, the involved personnel in the cloud system could face serious problems, for example having their personal data (e.g address or bank account details) going public, or having to pay an

egregious amount of money to someone who got hold of said data in order to keep it from reaching the public.

In 2017, a major security vulnerability, dubbed “Cloudbleed,” was discovered in the popular cloud service provider CloudFlare. The vulnerability stemmed from a bug in their implementation of a cryptographic middleware security feature called “Universal SSL.” This feature aimed to automatically provision SSL certificates for websites using CloudFlare's services. The incident resulted in a portion of unencrypted website content, including potentially sensitive information like private messages, login credentials, and personal data, to be unintentionally leaked into the cache of CloudFlare's servers. The incident demonstrates how vulnerabilities in cryptography implementations can lead to data breaches (Cloudbleed Triggered 1.2M Times).

During our research, we came up with some important points to consider when deciding how to include encryption in the system. First of all, a system should use strong, state of the art cryptographic algorithms like ECC (Elliptic Curve Cryptography), which is a good enough algorithm to protect even against attackers using quantum computers (Jana et al., 2017), SVM (Support Vector Machine) (Samanta et al., 2021), or a combination of said algorithms. It is also important to encrypt and protect log files, hence through those one could track some important traffic in the systems (Rao et al., 2019). The employed cryptographic method should have a good key rotation policy and strong access control in key management to ensure the maximum possible security.

Data-in-transit might be exposed to more attacks than data-in-rest as it travels from one place to the next, and might be private, so it is important to consider security measures to make the travel a safe one. Encryption plays a key part in securing data-in-transit, however it isn't sufficient, so further security protocols and network security equipment are needed. Furthermore, encryption of this data is different to encryption of data-in-rest. For example, the encryption for transit might use short-term keys as opposed to the longer lasting ones used for data-in-rest (Kacha & Zitouni, 2018).

A key rotation policy dictates how frequently cryptographic keys are changed. Regular key rotation mitigates the risks associated with a key being compromised over time. The frequency of rotation depends on several factors. Highly sensitive data requires more frequent changes compared to less critical information. The evolving threat landscape and potential vulnerabilities in cryptographic algorithms should also be considered when defining your policy. Finally, it's important to find a balance between security and performance, as frequent key rotation can introduce some overhead. NIST SP 800-57 (NIST SP 800-57 Part 1 Rev) 5 provides comprehensive guidance on key management practices, including key rotation, while the Cloud Security Alliance defines a standardized protocol (KMIP) (Key Management in Cloud Services) for secure key management, promoting interoperability between different key management systems while enabling access control features.

Based on these points, a few known algorithms and techniques could be used in order to strengthen the cryptographic method: A multilevel encryption (MLE) technique (for example using AES for encryption, and generate its keys using ECC) (Singh et al., 2022), employ client side encryption, so the customer can secure data in transit as opposed to the provider (Kumar et al., 2022), and employ blowfish algorithm (a slightly faster symmetric-key algorithm than DES with a good encryption rate) (Arulkumaran et al., 2023). MLE adds an extra layer of security by encrypting data multiple times with different algorithms (NIST to Withdraw Special Publication 800-67). discusses MLE as a technique for improving security. It mentions that using multiple encryption algorithms can offer additional protection against attacks that target specific algorithms. The selection of algorithms recommended in our paper is further supported by NIST (SP 800-38A) (NIST Recommendation for Block Cipher Modes of Operation).

5. Trusted Computing

While Trusted Computing is not unanimously accepted, the largest hardware manufacturers provide the option to use it. There are strong benefits of Trusted Computing, such as the computer behaves in expected ways, and most of them can be adapted to cloud computing systems.

One of the hardware elements of Trusted Computing is Trusted Platform Modules (TPM). A TPM is a secure processor or chip that ensures the integrity of the computer, and enables disk encryption among other uses. Several articles suggest the use of TPMs in cloud computing. El Balmany et al., (2019) points out that Openstack achieves integrity of launch processes with TPMs, (Paladi et al., 2016) proposes a protocol that uses TPMs and Secure Components for secure launch and storage protection, (El Balmany et al., 2022) also incorporates TPMs for its trusted zone-based cloud storage system.

A potential use of TPMs is remote attestation (RA), a method to detect unauthorized changes on a computer. According to (Parast et al., 2022) TPM-based remote attestation can combat hardware tampering. Ibrahim & Hemayed, (2019) provides an overview of different remote attestation methods, concluding that behavior-based RA is the most secure option. Huang et al., (2021) proposes an ABS (attribute-based signature)-based remote attestation technique, that allows users to test the integrity of the system.

TPMs can also be the target of attacks, such as the TPM reset attack. Against that and BIOS attacks DRTM (Dynamic Root of Trust for Measurement) or late-launch is a solution (Ibrahim & Hemayed, 2019). With this approach, software trust can be ignored until a secure event, it launches without system reboot. DRTM is supported by several manufacturer-specific technologies.

It is also a good idea to have a trusted third party available for attestation and key management (El Balmany et al., 2019; Paladi et al., 2016). It makes communication more secure during critical interactions, and guarantees trustworthiness. The assessment of quality of vendors is often done with TTPs (Svatá & Zbořil, 2020).

The components of trusted computing can be used for a vast amount of goals, and the reviewed articles provided some practical solutions to different problems. For example, (El Balmany et al., 2022) proposes vmiTPL, a security protocol to ensure the trusted launch of virtual machine images. Rahumath et al., (2021) describes a trust-based anomaly detection method to combat DDoS attacks.

Technologies like Intel SGX (Software Guard Extensions) and AMD SEV (Secure Encrypted Virtualization) create isolated regions within the processor memory. These technologies leverage processor architecture features to establish isolated memory regions called enclaves. These enclaves function as secure compartments for sensitive computations. Code and data loaded into the enclave are encrypted, ensuring confidentiality even from the underlying operating system or the cloud provider itself.

6. Virtual Resources

We grouped the security measures connected to virtual resources into four categories: the virtual infrastructure itself, virtual machine updates, the updates of IaaS software and the security of virtual machine images.

- **Virtual Infrastructure**

Virtual infrastructure is built upon software-defined components which make up the IT environment. It provides the same capabilities as the physical resources, but as software to make allocating virtual resources across multiple systems possible. There are multiple safety measures to be taken into consideration when talking about virtual infrastructure.

In the virtual network, security service functions (eg. Firewall) should be implemented so that they can collaborate with each other in order to make the system secure while not compromising the efficiency of it. Migault et al., (2017) proposes a Secure Firewall Framework (SFF) to achieve this. The SFF collaboration model aims to improve the security and efficiency of cloud systems by facilitating cooperation among security components like firewalls. Through the implementation of the SFF organizations can establish a unified security infrastructure that capitalizes on the combined capabilities of security components to defend against potential threats and vulnerabilities in a timely manner.

In 2018, several Google Cloud customers reported unauthorized cryptocurrency mining activity on their cloud instances. Attackers exploited vulnerabilities in container orchestration platforms like Kubernetes to deploy cryptojacking malware within containers. This incident highlights the potential consequences of security service functions (like firewalls) operating in isolation within a virtual network. While firewalls might be configured to block malicious traffic, they might not be able to detect and prevent sophisticated attacks like cryptojacking within containers (The Year Cryptojacking Ate the Web).

Providing the possibility of using virtual private cloud could (VPC) also be beneficial, because each user could have a dedicated VM and a dedicated backup, hence they do not need to share servers with other users, making the VMs more isolated and therefore more secure from attacks coming from other VMs in the system. Each user's data is stored separately, reducing the likelihood of data corruption or loss due to interactions with other users' resources. Sharma et al., (2017) proposes using the SpotCheck VPC approach that facilitates the implementation of dedicated VMs and backups for each user within the cloud infrastructure, by this approach, organizations can ensure that users have exclusive access to their VM instances and backup resources, enhancing the security and privacy of their data.

While it is important for a system to resist attacks, it is also important to consider recovery methods in case an attack does go through. Therefore, the use of self-recovery approach is important to implement in the virtual infrastructure. Joseph & Mukesh, (2019) provides a detailed method to achieve this by taking snapshots regularly, using a self-recovery algorithm (e.g Naive Bayes or SVM) to investigate whether or not the VM has been attacked, then powering off the VM and restoring it to the earliest snapshot in case the algorithm detected an anomaly. Joseph & Mukesh, (2019) provides some different snapshot technologies for a method similar to the aforementioned.

One of, if not the most important part of the virtual infrastructure is the hypervisor, which is responsible for the hosting of the VMs. Therefore, extra attention must be paid when considering securing the hypervisor, as it could compromise the whole system in case of an attack reaching it (Singh et al., 2022). Barrowclough & Asif, (2018) provides a good overview on the hypervisor security measures which should be taken and some possible exploits which should be taken into consideration. One important method is to limit the hypervisor’s access to protected areas or self-recovery mentioned above (Zimba et al., 2017).

While the usage of VM migrations is important to allocate resources in order to keep the system efficient, the system also becomes vulnerable during the process so it is important to make preparations against possible attacks. During migration, security policies of the VM must be applied to the security devices in that area, such as IDS and firewalls (Huang et al., 2021). Duncan et al., (2013) describes a sophisticated cyberattack campaign targeting vulnerabilities in cloud workloads during migration between different cloud environments. Attackers could scan for cloud workloads undergoing migration processes. This could be achieved by analyzing network traffic patterns or exploiting potential information leaks within the cloud platform. The intercepted communication could reveal security policies that hadn't yet been applied to the destination environment's security devices (firewalls, IDS). This temporary gap in security would allow attackers to inject malware or gain unauthorized access to the VM.

Yin et al., (2018) proposes a 3 layered security framework dubbed HyperAV for securing the VMs in the system. The first layer of the framework emphasizes tenant domain perimeter security, ensuring that each tenant's domain is protected at the network boundary to prevent unauthorized access and attacks, establishing a secure boundary for each tenant's domain. The second layer addresses VM perimeter security, aiming to secure individual VM instances within the system. The third layer of the security framework focuses on VM OS internal security, hardening the operating system within each VM. Techniques recommended by HyperAV include multilevel encryption, together SSL/TLS security protocols, additionally, limiting direct access to files and data, employing access control mechanisms, and implementing monitoring systems for anomaly detection.

- **VM Updates**

In cloud systems template images are used to create VM instances. These images are normally accompanied with configuration templates and first-time running instructions. This has practical advantages, but the monoculture of VM-s also present a risk.

It is a problem that these images tend to not be updated with security features the way live operating system instances are. Moreover other factors such as a known vulnerability of software in certain configurations, can allow serious security incidents. Also, as the VM-s are essentially clones, a malicious insider, by having an own instance, gains information of a high number of other instances. Moreover, in case of careless configuration (see Networking section), the attacker might monitor every VM start in the IaaS by renting just one VM in the system. This way even when the newly started VM is configured to download security patches right away after first start, an exploitable time window presents itself for the attacker.

One solution is to keep running “etalon” virtual machines always with the latest security patches. This running VM can be copied when new virtual machines are started. This, however, rules out all the simple technical options for duplication as a running VM can write to the disk at any given time, and the copy of its disk can result in an inconsistent new VM. This inconsistency can be at file-system level or on the level of OS software.

In 2017, the WannaCry ransomware attack wreaked havoc across the globe, infecting hundreds of thousands of computers. A critical factor in the attack's success was the exploitation of a vulnerability in older versions of Microsoft Windows Server Message Block (SMB) protocol, codenamed "EternalBlue." Many organizations unknowingly used cloud infrastructure templates containing outdated and unpatched versions of Windows Server. These templates served as blueprints for rapidly provisioning new virtual machines (VMs). Even if the newly created VM downloaded security updates upon its first boot, a vulnerable window existed between launch and patch application. Attackers raced

against this window to exploit the EternalBlue vulnerability before the system became secure (EternalBlue Exploit).

There are more secure alternatives, but these increase the complexity. The IaaS may monitor the processes in the master VM and make a copy — for use as a template — preemptively right after the automatic update activity ended within the OS. Or the IaaS may monitor a security notice feed, start an instance of the master VM when there are security updates available, wait for the update to happen in the master VM and shut down the VM.

Infrastructure as a Service (IaaS) providers often offer methods for monitoring the overall update state and running processes in the master Virtual Machine (VM). For instance, Azure provides guidance on how to implement remote administration and management, enabling users to ensure that the master VM is continuously kept updated. Unfortunately this solution is not vendor agnostic (Monitor and maintain Windows Server).

Another problem presents itself if the users are allowed to take snapshots to which they can roll back later, to a state that is not patched for security.

The security updates themselves can be exploited in case of VMs that were suspended for a long time (Kourai & Shiota, 2019). If the VM comes online for the update it provides a time window for the attackers to exploit vulnerabilities that were present before the update. To solve this problem, offline updating of VMs is proposed, however this solution is not one without flaws. Some approaches require the resuming of VMs, others are only applicable to stopped VMs. The latter methods could corrupt virtual disks when applied to suspended machines. Emulating the security updates offline is also not a trivial task. Kourai & Shiota, (2019) proposes a method, OUAssister, that can be applied to VMs partly while suspended (via emulation), and then the results are applied online. This way both corruption and attacks can be prevented.

- **Software Updates**

Security update management is equally crucial in the IaaS infrastructure itself. A typical IaaS the amount of allowed downtime is minimal, even in a scheduled fashion for achieving high availability. This is the main issue in this area.

Security issues and updates concerning the IaaS infrastructure happen just as frequently as the software used in the VM (for instance because they use a similar Linux distribution). Yet, regular automatic updating of the IaaS is very hard to achieve because an update often requires the restart of a service. The restart of a component that provides the virtualized disk or network for VMs might disrupt the running VMs or might even corrupt their file systems. A restart of the virtualization service or the whole host means downtime for the VMs. Therefore, as many elements of the IaaS should be updateable independently from other elements as possible. This shows that solutions that enable live failover of components are not only essential for mitigating hardware failures but also because they enable updating.

Wang et al., (2022) proposes a live networking device update mechanism that minimizes the performance loss during security updates. The updates do not require a shutdown of the hardware nor hardware-redundancy, offering an effective method to achieve high availability in the cloud system.

In a similar way, the possibility of live migration of VMs between hosts is not only a convenience, but a security feature.

While our paper focuses on IaaS environments, a similar challenge exists for containerized workloads managed by Platform-as-a-Service (PaaS) offerings. Container orchestration systems often address this issue more effectively, frequently employing techniques like continuous service delivery with rolling updates. This approach, often supported natively by cloud container platforms (e.g., Kubernetes (Performing a Rolling Update)), could serve as a valuable pattern for developing effective solutions within IaaS contexts.

Finally, just like with operating systems, it is preferred that the IaaS updates are coming from an authentic source (e.g., signed updates) and that there are proper tools to upgrade and roll back between software versions.

The issues around infrastructure updating most often lead to a situation in which systems containing known vulnerabilities must run for an extended period of time before they can be updated. The problem is partially alleviated by having an isolated network with strong security for the management side.

- **Virtual Image Security**

In 2022, security researchers at Comp airtech issued a comprehensive report on access security misconfigurations in Google Cloud Platform (GCP) that exposed sensitive data in customer storage buckets. Besides predictable naming conventions for buckets and credentials, the incident involved VMs provisioned from shared templates that potentially contained misconfigured security settings, allowing access to storage buckets. This demonstrates how information present within a template can be inadvertently exposed if not properly secured (6% of all Google Cloud Buckets).

While this problem is partially present in OS distributions in general, the setup process of an OS includes generating a unique salt value, choosing users and passwords, maybe even disk encryption, etc. This is not necessarily emulated in VM instantiation in an IaaS server.

In some IaaS systems, users can upload their VM templates into a shared IaaS repository or they can use a common marketplace. In such systems it is essential that the creator and creation time be made explicit and visible. Moreover, it is preferred that there are VM templates marked as approved and that the management interface can filter by creator or such tags.

While in a previous section it was suggested that the update of VMs should be performed regularly, images at rest can contain vulnerabilities that can be exploited even in a limited time window. VMIs can even be contaminated, causing the user to use a malicious virtual machine. To prevent this, both the encryption of virtual images and the maintenance of the repository (e. g. virus and vulnerability scans) is important (Kumar & Goyal, 2019). Secure encryption of the images can be achieved by the end users' own keys, that way only the authorized user can access their own VMIs. The aforementioned trusted zone-based storage method (El Balmany et al., 2022) provides a practical solution for secure VMI storage. Liu et al., (2019) offers a data-centric VMI management approach, where VMIs are viewed as structured data. The approach, named Hemera, supports vulnerability scanning and auditing operations (VMIs are stored in a centrally-managed storage, exposed to scanning tools).

7. Operations Center

In this section we cover the security measures connected to the Operations Center. This includes monitoring, as the main goal of the component, but also networking and web interfaces, since monitoring tools are often web-based.

- **Monitoring**

The monitoring of different components in the cloud system is important because it can produce valuable data on the tendencies of attacks or regular workflow, making the detection of anomalies faster and easier. It is important to note that monitoring by itself is not a sufficient security step, its importance comes from the fact that security focused analysis of the collected monitoring data can provide other measures and solutions with the data required to mitigate attacks, and that is why it is mostly used in security incident detection solutions.

Monitoring plays an important part in using the Complex Event Processing (CEP) technique as a detection systemmeasure, since it monitors incoming traffic, processes data in real time and detects possible incoming (DDoS) attacks (Devi & Subbulakshmi, 2021). However, security monitoring extends far beyond DDoS attack detection. CEP can be configured to identify deviations from normal system behavior. This can include unusual login attempts (brute-force attacks, credential stuffing), unauthorized access to sensitive data (data breaches), a surge in failed login attempts (potential account takeover attempts), departures from typical file access patterns (potential ransomware encryption). Security monitoring helps ensure compliance by tracking user activity, access controls, and system configurations, identifying potential breaches or deviations from compliance standards. By analyzing activity data, access patterns, and file modifications, unusual activity can be flagged for investigation, potentially uncovering attempts at sabotage or data theft by insiders.

The usage of monitoring can be paired with machine learning algorithms to detect anomalies as well, for example (Lin et al., 2021) proposes the BTDetect method, which monitors incoming traffic based on normal user usage instead of known attack patterns, so it has a better chance of detecting currently unknown types of attack. Another example would be the proposition of (Al-Bayati et al., 2018), which suggests the usage of behavior profiling with machine learning algorithms and comparing them to user profiles for continuous and transparent identity verification. However, this method might compromise the users' privacy and also raises ethical questions.

Another method of employing monitoring is trust based anomaly detection, which evaluates the trustworthiness of the monitored data based on metrics such as metrics such as Frequency Value, Trust Hypothesis Statistics, Trust factor value, and trust policy (Rahumath et al., 2021).

Lin et al., (2022) proposes VNGuarder, which aims to protect virtual network devices, virtualization management processes and API by monitoring these parts and using trace-enable mechanisms to establish a normal behavior to compare to the monitored behavior.

Gill & Shaghaghi, (2020) proposes STARK, a complex resource allocation model, which focuses on mitigating DDoS attacks, and is employing monitoring methods as well.

Gill & Buyya, (2018) proposes SECURE, which also requires monitoring in intrusion detection systems (be it signature based or anomaly based).

Naturally, a logging system is important to have in a cloud system, and monitoring can also be used to make logs based on usage of the system (Auxsorn et al., 2020). A good logging system should capture incidents in the system at least for later evaluation, but ideally for online alerting. The usage of

a central logging server might be important to make searching and analyzing the logs more efficient (Parast et al., 2022). A good logging system can also be combined with the IaaS controller (eg. Openstack) to make its existing security measures more secure (Wichep et al., 2020). It goes without saying that ensuring the existing security logs' protection is also important considering possible later forensic investigations. Rao et al., (2019) provides a good framework for log security.

- **Networking**

In an IaaS, the users typically share the same physical network resources. This requires active steps to be taken even only to reach the inherent security level of local deployments.

If the IaaS users share the same VLAN or LAN, an there are no further isolation measures, well-known LAN-based attacks (Aburukba et al., 2022) are possible: ARP poisoning, DHCP spoofing, CAM overflow and others. Therefore, ARP traffic needs to be filtered; all DHCP packages not originating from the valid DHCP server have to be filtered.

Moreover, an attacker with a VM can claim the IP address of another VM. On a simple LAN, this can cause IP conflicts leading to denial of service as well as data theft, that is, capturing IP packets not intended for the attacker. If the attacker can claim the gateway IP address, then the possibility of data theft is even bigger. These might be prevented by software-defined networking (SDN). Filtering needs to work in both directions, that is, an attacker VM should not be able to send IP packages with another VMs address as a source because this might also enable different kinds of service disruptions.

It is preferable to run the elements of the IaaS itself in a separate network. For instance, it is not advisable to run the DHCP server, accounting, etc. components of the IaaS in a VM (just like user VMs) on the IaaS itself. While the drawbacks of a shared network might be mitigated through security best practices like segmentation and access controls, it is still preferable to run the core elements of the IaaS itself on a separate, dedicated network (Barrowclough & Asif, 2018).

For obvious reasons IaaS management has to happen on a dedicated logical network, isolated from the network of the VMs. Otherwise it would be theoretically possible to eavesdrop or forge control messages or even capture net-based virtual disk content or even RAM content at live migration of VMs.

Also, it is critical that every security measure should work on IPv6 with the same efficiency as it works for IPv4, if there is any IPv6 networking in the IaaS. One such measure is network address translation (DNAT/ SNAT), which is widely employed on IPv4 but has no direct counterpart on IPv6. While SNAT is helping to preserve IPv4 addresses, it also provides some security by hiding the VMs virtual addresses from the network. As this is not possible in IPv6, a different measure needs to be found.

A quite simple, but nonetheless important security measure to take is to close unused ports as they could be vulnerable for Man-in-the-Cloud or Port scanning attacks (Parast et al., 2022). The article also states that pinging should be disabled - it could enable attacks such as ping of death (that could lead to buffer overflow - although not efficient nowadays) or ping flooding (denial of service). Man-in-the-Cloud attacks are a type of cyber threat where an attacker intercepts and alters communications between two endpoints in a cloud environment (Barrowclough & Asif, 2018). These attacks can lead to data theft, unauthorized access, and compromise the integrity of the system (Rakotondravony et al., 2017). Man-in-the-Cloud attacks are particularly concerning as they can exploit vulnerabilities in the shared network infrastructure of cloud environments, making it crucial to implement robust security measures to prevent such intrusions (Mthunzi et al., 2020). By leveraging

techniques such as encryption of virtual images, maintaining secure repositories, and employing third-party auditors, organizations can enhance their defenses against Man-in-the-Cloud attacks. Additionally, regular vulnerability assessments, monitoring with machine learning algorithms, and utilizing TPM-based remote attestation can help detect and mitigate these attacks effectively.

One way to establish a secure, encrypted communication on the network is to use SSL/TLS certificates (SSL stands for Secure Socket Layer, TLS, which is an upgraded version of SSL, stands for Transport Layer Security). SSL/TLS can ensure the confidentiality of transactions with databases (Halabi et al., 2018) or between clients and servers (Kumar & Goyal, 2019). It is also suggested by (Halabi et al., 2018) that service providers should enable client certificates of SSL/TLS beside server certificates as it can improve the trustworthiness of authentication.

8. Web Management Interface

IaaS systems in general offer a web management interface for the user. As most of the resources can be managed from there, it is a high-value target. It is even worse if the administrators are sharing the very same web interface with all users. This kind of risk was realized in the case of the 2013 SolusVM exploit.

An Akamai report conducted on hundreds of billions of API calls found that 12% of calls came from known bad actors, and 25% came from end clients that were neither web browsers nor mobile devices or applications, meaning that they could have originated with malicious actors instead of legitimate users (Akamai State of the Internet Security Report).

Therefore, hardening th web interfaces of IaaS is very important. Web hardening has its own methodology, that is not covered in this article. A good starting point is OWASP testing for the software version to be deployed. A Fuzz testing for the web interface, and, if provided the API by the vendor is also recommended.

Deployment-level measures, like hiding server and OS information from the error messages, unloading unnecessary modules, and turning off directory listing are also essential.

The endpoints of the public interface should be protected with secure protocols, fortunately there are standardized solutions that proved to be efficient. The most important is using HTTPS instead of HTTP which is unreliable (Chaudhary et al., 2020). The connection can be further secured using VPNs, however in this case the endpoints should be thoroughly authenticated.

9. User Portal

The User Portal is responsible for managing user accounts, therefore it is subjected to attack vectors that every user management system faces. The user portal could be the same web application that serves as the management interface, but it is often architecturally distinct. In general, all measures mentioned in the previous section applies, on top of which we can consider the measures below.

In the IaaS cloud there be threats caused by the renouncement of control, such as malicious insiders, who have to be filtered by strong authentication and access control. This is more likely in the user population which is larger and often self-registered. It is essential that the AAA principles, authentication, authorization and access control are sufficiently enforced.

- **Authentication**

In a cloud system, users are properly authenticated is very important, because a lack of a secure authentication system could leak sensitive data to unauthorized personnel, or an attacker could attack the system from the inside if they managed to get past it. During our research, we came up with some important points to consider when designing the cloud's authentication system.

Batista et al., (2018) comes forward with the idea of combining Single Sign-On mechanisms like OpenID connect or Facebook connect with OpenStack's Keystone to make authentication simple and secure as it is safer than to implement a new Single Sign-On. Alsaadi et al., (2020) suggests that authenticating the biometric could be the safest way to authenticate single sign-on.

Employing multi factor authentication could marginally increase the effectiveness of authenticating a user, for example when the user's password gets leaked, it is unlikely that the user's email account has also been compromised as well. Halabi et al., (2018), among many other things, mentions the importance of TFA.

Kumar & Goyal, (2019) among many other things, proposes the idea of using automatic validation for authentication called ProVerify, which can be used with OpenID connect.

As mentioned in the monitoring section before, (Al-Bayati et al., 2018) proposes behavior profiling for continuous and transparent identity verification.

Park et al., (2022) suggests the usage of robust passwords, and lists some best practices regarding the making of a strong password, and setting these as policies for password-making.

Kumar & Goyal, (2019) also states that using a digital signature with SSO and Lightweight Directory Access Protocol (LDAP) provides a stronger authentication process while providing user mobility and flexibility at the same time.

In case of breaches detected by the failure of authentication, it might be a good idea to send notifications about the breach to see what went wrong (Parast et al., 2022).

Chattaraj et al., (2018) proposes an authentication system based on two servers rather than the usual single server solution to increase the availability and security of the authenticating process itself (for example this solution makes the system not susceptible to single point of failure).

10. Authorization and Access Control

Unauthorized users gaining access to sensitive resources due to inadequate access control mechanisms in an IaaS management environment poses a significant security risk. Weak access control policies can lead to unauthorized access to critical data and systems, potentially resulting in data breaches and unauthorized activities. Implementing a trusted zone-based storage method for virtual machine images can enhance the protection of sensitive resources and prevent unauthorized access.

After sufficient authentication the proper authentication and access control policies have to be enforced, in order that only users who have the right to make changes to the cloud resources can do so. It means that other, unauthorized users, and even employees of the cloud service provider should not be granted access, the latter measure preventing malicious insider attacks.

The main goal of access control is to prevent unauthorized users from modification, prevent authorized users from unauthorized modification (Kumar & Goyal, 2019), while enabling authorized users to perform authorized actions (Singh & Sharma, 2019).

There are two main types of access control: attribute-based and role-based access control. Role-based access control (RBAC) is the most common model, where users are provided roles and privileges, and they can perform actions according to them. Attribute-based access control (ABAC) is a more customizable approach; more complex rule sets can be defined with the help of attributes. Another classification method is that access control models can be either MAC (mandatory access control) or DAC (discretionary access control). The former is an access control approach where the operating system defines constraints to the access of some objects. Meanwhile, in DAC usually the owner of the object can define who is to perform operations on the object. Kumar & Goyal, (2019) suggests that for every approach there is a strong solution (trust-based access control is also mentioned).

In the case of RBAC, (Couto et al., 2018) suggests the use of hierarchic roles. There are local and global administrators; the former deals with management of their own site, while the latter can perform every action a local administrator can, but they have their own specific capabilities such as user creation and global migration.

Singh et al., (2022) provides an attribute-based access control model that preserves the RBAC-approach, but also augments ABAC to achieve its flexibility. The model uses a security label engine, security tokens, and it assigns users into security zones.

Users who have access to certain resources, might have conflict of interest. It is important that conflicting parties cannot modify the same resource, thus entities have to be assigned to different classes, who have no conflict of interest (Oberoi et al., 2019). Users have no access to members of other classes. Oberoi et al., (2019) achieves it using a multi-level cloud security policy that uses both the Chinese Wall model (ensures the aforementioned classification) and the Clark-Wilson model.

The latter article also suggests that direct access to files and data should not be allowed to preserve integrity. Alsaadi et al., (2020) agrees with this point, and claims that using indirect references to objects prevents attackers from directly accessing unauthorized resources.

A large amount of models, approaches and protocols were created for authorization and access control, and a handful of them have been standardized. These protocols are well known to be secure, they are maintained and best practices have been formed around them. We suggest using these instead of homemade approaches. Kumar & Goyal, (2019) lists a number of options and implementations in this regard.

Manually evaluating access policies to check for known vulnerabilities might be hard, and that is why (Jin et al., 2022) proposes P-verifier. This article lists several possible vulnerabilities regarding said policies and explains how its authors took those into consideration when designing P-verifier, which can automatically detect said vulnerabilities with little performance overhead in a cloud system.

Most IaaS cloud providers commonly utilize the Role-Based Access Control (RBAC) authorization model for managing access rights and permissions within the cloud environment. RBAC allows administrators to assign roles to users based on their responsibilities and job functions, simplifying the management of access control policies. By implementing RBAC, cloud providers can streamline the authorization process, reduce the complexity of managing individual user permissions, and enhance overall security.

RBAC offers a structured approach to access control, where permissions are assigned based on predefined roles rather than individual users. This model aligns well with the dynamic nature of cloud environments, enabling efficient management of access rights as users' roles and responsibilities evolve. Additionally, RBAC facilitates the enforcement of the principle of least privilege, ensuring

that users only have access to the resources necessary for their tasks, thereby reducing the risk of unauthorized access (Shin et al., 2011).

Moreover, RBAC enhances security by centralizing access control policies and simplifying the enforcement of security measures across the cloud infrastructure. This centralized approach improves visibility and control over user permissions, making it easier to monitor and audit access activities within the IaaS environment. Overall, the RBAC authorization model provides a robust framework for managing access control in IaaS environments, contributing to enhanced security and efficient resource management (Sultan et al., 2022).

11. Other Security Measures

In this section we cover cloud security measures that cannot be assigned to the above mentioned parts of the IaaS architecture. The following checklists are about providing the users with information, auditing and other miscellaneous measures.

- **Providing Users with Information**

According to (Narang & Gupta, 2018). one of the security issues with cloud computing is that users do not have enough information on the physical location where their data is stored. This means it is difficult for them to gain sufficient information and control about their data. We suggest that providers should be transparent about physical parameters of VM and data storage.

Having access to this kind of information makes it possible for the users to increase their security. This includes preparedness as well as post-incident analysis and forensics abilities. While many users might opt to delegate every task related to infrastructure to the provider, this is not a reason to refrain from providing a possibility for self-monitoring or custom security enhancements. They could even be allowed to customize their security policies, as suggested in (Huang et al., 2021). The proposed architecture enables user-side validation of the custom policies.

Users might be interested in possible vulnerabilities in their systems or used services, so it is also important for providers to warn their users if some concerns arise. Providers should also consider recommending mitigation methods to the users along with the warnings, and they should also aim to mitigate said issues themselves as soon as possible.

As (Aburukba et al., 2022; El Balmany et al., 2019; El Balmany et al., 2022; Hirano et al., 2018) suggests, security in the cloud is a mutual responsibility. In fields such as integrity and authentication the only way to achieve the complete security of the system is to cooperate with the end users, so providing information for them is essential. Moreover, providing users with these kinds of information and being transparent establishes trust between the user and the provider, which is the foundation of efficient partnership.

- **Other Measures**

This section covers topics that are mostly unrelated to the previous sections, or that are just loosely connected to IaaS security, but still can be useful to consider.

The employment of a third-party auditor comes with the benefit of trustworthiness. Latha, (2018) claims that many service providers hide data corruption of users, thus a third party auditor is necessary. A TPA does not have access to the data but helps to secure it. It should have the following functionalities: no data leakage, integrity verification, high performance, scalability.

Every component of the system could benefit from regular vulnerability assessment to increase their security. Patil & Modi, (2019) The article then mentions quite a few methods for assessment, both for software and for VMs as well. The article then proposes a framework for vulnerability assessment and patching (VAP).

It should be mentioned that several established guideline exist on general best practices when building a cloud system, such as ISO (International Organization of Standardization) 2700x family, NIST (National Institute of Standards and Technology) Special publication 800-53 and Special publication 800-144, Cloud Security Alliance (CSA), which has 14 domains about critical areas in cloud computing and works on improvement and published best practice (Bauer et al., 2017).

Testing can be a difficult task for cloud systems, due to the distributed nature of it. Testers often work parallel to each other, and this method requires strong coordination, since the lack of it could cause problems with controllability and observability of testing (Moutai et al., 2019). The article also suggests that tests should not only follow their specification, but also some security policies and the CIA triad. The concept of a security policy language is also introduced there.

Another topic loosely connected to cloud security is forensics in the cloud. Bhatia & Malhotra, (2019) describes the two fields as two sides of the same coin - security focuses on prevention, while forensics is a post-implementation task, including investigation and analysis. They are connected, even though cloud service providers often neglect the proper implementation of forensics, according to the article. It is recommended that the provider prepares forensic tools in case an attack is launched successfully against the cloud system.

3 Conclusion

In this article we provided a number of security measures that can be used to protect IaaS cloud systems. We presented an overview of different security challenges in the cloud, and best practices/guidelines to counter them. In order that this article can be used for quicker reviews, we summarized our results in checklists in the table format in (An IaaS Security Checklist). An important limitation of our work however, that it is technically focused, with less emphasis on attempts at changing or nudging user behavior.

Obviously, the contents of such a work are subject to deprecation, however, we believe that while the particular attacks might change over time, the architecture of an IaaS cloud is quite stable and therefore our categorization of security issues should stand the test of time.

We argue that our article holds significant practical value which lies in the comprehensive mapping of the identified security threats within the IaaS environment. By outlining a problem map, the article offers a structured overview of potential vulnerabilities and risks that users may encounter, allowing for a clear understanding of the security landscape.

Understanding the landscape of security threats is crucial for IaaS users and operators to develop effective security strategies and responses. The problem map serves as a foundational tool for users to assess their current security posture, identify potential areas of concern, and prioritize security measures based on the severity and impact of the threats outlined in the document.

Moreover, the problem map can act as a reference point for users to engage in further research, consultation with security experts, and the implementation of tailored security solutions. By providing a structured overview of security threats without prescribing specific remediation options, the docu-

ment empowers users to delve deeper into each identified threat, explore relevant mitigation strategies, and customize security measures to suit their specific IaaS environment.

In conclusion, the added value of our research lies in its ability to offer a comprehensive problem map of security threats in the IaaS environment, enabling users to enhance their security posture through informed decision-making and targeted security initiatives.

References

[1] Aburukba, R., Kaddoura, Y., & Hiba, M. (2022). Cloud Computing Infrastructure Security: Challenges and Solutions. *In IEEE International Symposium on Networks, Computers and Communications (ISNCC)*, 1-7.

[2] Al-Bayati, B., Clarke, N., Dowland, P., & Li, F. (2018). Misuse detection in a simulated IaaS environment. In *Emerging Technologies for Authorization and Authentication: First International Workshop, ETAA 2018*, 103-115.

[3] Alsaadi, E. M. T. A., Fayadh, S. M., & Alabaichi, A. (2020). A review on security challenges and approaches in the cloud computing. *In AIP Conference Proceedings*, *2290*(1). https://doi.org/10.1063/5.0027460

[4] Arulkumaran, G., Jayagopalan, S., & Balamurugan, P. (2023). An Effective Analysis of Proficient Two Level Security Contraptions For Loading Data In Cloud. *In IEEE 1st International Conference on Innovations in High Speed Communication and Signal Processing (IHCSP)*, 536-540.

[5] Auxsorn, T., Wongthai, W., Porka, T., & Jaiboon, W. (2020). The accuracy measurement of logging systems on different hardware environments in infrastructure as a service cloud. *ICIC Express Letters, Part B: Applications, An International Journal of Research and Surveys*, *11*(5), 427–437.

[6] El Balmany, C., Asimi, A., Tbatou, Z., Asimi, Y., & Guezzaz, A. (2019). Openstack: launch a secure user virtual machine image into a trust public cloud IaaS environment. *In IEEE 4th World Conference on Complex Systems (WCCS)*, 1-6.

[7] El Balmany, C., Asimi, A., & Tbatou, Z. (2022). VMITLP: A Security Protocol Towards a Trusted Launch Process of a User Generic Virtual Machine Image on a Public Cloud IaaS Platform. *IAENG International Journal of Computer Science*, *49*(1).

[8] Barrowclough, J. P., & Asif, R. (2018). Securing cloud hypervisors: a survey of the threats, vulnerabilities, and countermeasures. *Security and Communication Networks*, *2018*(1), 1681908. https://doi.org/10.1155/2018/1681908

[9] Batista, G. C., Miers, C. C., Koslovski, G. P., Pillon, M. A., Gonzalez, N. M., & Simplicio, M. A. (2018, May). Using externals IdPs on OpenStack: A security analysis of OpenID connect, Facebook connect, and OpenStack authentication. *In IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, 920-927.

[10] Bauer, E., Schluga, O., Maksuti, S., Bicaku, A., Hofbauer, D., Ivkic, I., & Wöhrer, A. (2017). Towards a security baseline for IaaS-cloud back-ends in Industry 4.0. *In IEEE 12th International Conference for Internet Technology and Secured Transactions (ICITST)*, 427-432.

[11] Bhatia, S., & Malhotra, J. (2019). Forensic based cloud computing architecture–exploration and implementation. *In IEEE 3rd International Conference on Computing and Communications Technologies (ICCCT)*, 37-45.

[12] Chattaraj, D., Sarma, M., & Das, A. K. (2018). A new two-server authentication and key agreement protocol for accessing secure cloud services. *Computer Networks*, *131*, 144-164.

[13] Chaudhary H., Batra S., Gautam M. (2020). Implementing a Secure Cloud Environment: an Explorative Study. *Journal of Critical Reviews, 7*(13), 350-352.

[14] El Balmany, C., Tbatou, Z., Asimi, A., & Bamarouf, M. (2022). Secure virtual machine image storage process into a trusted zone-based cloud storage. *Computers & Security, 120*, 102815. https://doi.org/10.1016/j.cose.2022.102815

[15] Couto, R. S., Sadok, H., Cruz, P., Da Silva, F. F., Sciammarella, T., Campista, M. E. M., & Rubinstein, M. G. (2018). Building an IaaS cloud with droplets: a collaborative experience with OpenStack. *Journal of Network and Computer Applications, 117*, 59-71.

[16] Devi, B. K., & Subbulakshmi, T. (2021, May). Cloud DDoS detection and defense system using complex event processing. *In IEEE 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 118-128.

[17] Gill, S. S., & Buyya, R. (2018). SECURE: Self-protection approach in cloud resource management. *IEEE Cloud Computing*, *5*(1), 60-72.

[18] Gill, S. S., & Shaghaghi, A. (2020). Security-aware autonomic allocation of cloud resources: a model, research trends, and future directions. *Journal of Organizational and End User Computing (JOEUC)*, *32*(3), 15-22.

[19] Halabi T., Bellaïche M., & Abusitta A. (2018). Online Allocation of Cloud Resources Based on Security Satisfaction. *17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/ 12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE)*, 379–384.

[20] Hirano, M., Tsuzuki, N., Ikeda, S., & Kobayashi, R. (2018). LogDrive: a proactive data collection and analysis framework for time-traveling forensic investigation in IaaS cloud environments. *Journal of Cloud Computing*, *7*(1), 18. https://doi.org/10.1186/s13677-018-0119-2

[21] Huang, C., Chen, W., Yuan, L., Ding, Y., Jian, S., Tan, Y., & Chen, D. (2021). Toward security as a service: A trusted cloud service architecture with policy customization. *Journal of Parallel and Distributed Computing*, *149*, 76-88.

[22] Ibrahim, F. A., & Hemayed, E. E. (2019). Trusted cloud computing architectures for infrastructure as a service: Survey and systematic literature review. *Computers & Security*, *82*, 196-226.

[23] Jana, B., Poray, J., Mandal, T., & Kule, M. (2017). A multilevel encryption technique in cloud security. *In IEEE 7th International Conference on Communication Systems and Network Technologies (CSNT)*, 220-224.

[24] Jin, Z., Xing, L., Fang, Y., Jia, Y., Yuan, B., & Liu, Q. (2022). P-verifier: Understanding and mitigating security risks in cloud-based iot access policies. *In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 1647-1661.

[25] Joseph, L., & Mukesh, R. (2019). Securing and Self recovery of Virtual Machines in cloud with an Autonomic Approach using Snapshots. *Mobile Networks and Applications*, *24*(4), 1240-1248.

[26] Joseph, L., & Mukesh, R. (2019). To Detect Malware attacks for an Autonomic Self-Heal Approach of Virtual Machines in Cloud Computing. *Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, *1*, 220–231.

[27] Kacha, L., & Zitouni, A. (2018). An overview on data security in cloud computing. *Cybernetics Approaches in Intelligent Systems: Computational Methods in Systems and Software*, *1*, 250-261.

[28] Kourai, K., & Shiota, Y. (2019). Consistent Offline Update of Suspended Virtual Machines in Clouds. *IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, 58–65.

[29] Kumar, M., Santhiya, G., Jeni, V., & Bhavna, J. (2022). Web Application Security on Top of Public Cloud. *In IEEE Second International Conference on Interdisciplinary Cyber Physical Systems (ICPS)*, 210–215.

[30] Kumar, R., & Goyal, R. (2019). On cloud security requirements, threats, vulnerabilities and countermeasures: A survey. *Computer Science Review*, *33*, 1-48.

[31] Latha, K. (2018). Enforcing security in cloud environment using elliptic curve cryptography and third party auditing. *International Journal of Engineering and Technology*, *7*(1.7), 84-86.

[32] Lin, L., Li, S., Lv, X., & Li, B. (2021). BTDetect: An Insider Threats Detection Approach Based on Behavior Traceability for IaaS Environments. *IEEE International Conference on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, 344–351.

[33] Lin, L., Yang, H., Zhan, J., & Lv, X. (2022). VNGuarder: An Internal Threat Detection Approach for Virtual Network in Cloud Computing Environment. *Security and Communication Networks*, *2022*(1), 1242576. https://doi.org/10.1155/2022/1242576

[34] Liu, H., He, B., Liao, X., & Jin, H. (2019). Towards Declarative and Data-Centric Virtual Machine Image Management in IaaS Clouds. *IEEE Transactions on Cloud Computing 7*(4), 1124–1138.

[35] Migault, D., Simplício, M.A., Barros, B.M., Pourzandi, M., Almeida, T.R.M., Andrade, E.R., & Carvalho, T.C.M.B. (2017). A Framework for Enabling Security Services Collaboration Across Multiple Domains. *IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, 999–1010.

[36] Moutai, F. Z., Hsaini, S., Azzouzi, S., & Charaf, M. E. H. (2019). Security testing approach for IaaS infrastructure. *In Proceedings of the 2nd International Conference on Networking, Information Systems & Security*, 1-5.

[37] Narang A., & Gupta D. (2018). A Review on Different Security Issues and Challenges in Cloud Computing. *International Conference on Computing, Power and Communication Technologies (GUCON)*, 121–125.

[38] Oberoi, P., Mittal, S., & Gujral, R. K. (2019). Multilevel Cloud Security Policy (MCSP) for Cloud-Based Environments. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, *8*(8S3), 145-152.

[39] Paladi, N., Gehrmann, C., & Michalas, A. (2016). Providing user security guarantees in public infrastructure clouds. *IEEE Transactions on Cloud Computing*, *5*(3), 405-419.

[40] Parast, F. K., Sindhav, C., Nikam, S., Yekta, H. I., Kent, K. B., & Hakak, S. (2022). Cloud computing security: A survey of service-based models. *Computers & Security*, *114*, 102580. https://doi.org/10.1016/j.cose.2021.102580

[41] Park, S. J., Lee, Y. J., & Park, W. H. (2022). Configuration method Of AWS security architecture that is applicable to the cloud lifecycle for sustainable social network. *Security and Communication Networks*, *2022*(1), 3686423. https://doi.org/10.1155/2022/3686423

[42] Patil, R., & Modi, C. (2019). Designing an efficient framework for vulnerability assessment and patching (VAP) in virtual environment of cloud computing. *The Journal of Supercomputing*, *75*(5), 2862-2889.

[43] Rahumath, A. S., Natarajan, M., & Malangai, A. R. (2021). Resource Scalability and Security Using Entropy Based Adaptive Krill Herd Optimization for Auto Scaling in Cloud. *Wireless Personal Communications*, *119*, 791-813.

[44] Rao J.N., Deshpande A.A., Patil P., & Nikam S. (2019). Design of Security Technique through Secure Logging for Cloud Forensics. *International Journal of Engineering and Advanced Technology*, *8*(6), 4035-4043.

[45] Sahu, I. K., & Nene, M. J. (2021). Model for IaaS security model: MISP framework. *In IEEE International Conference on Intelligent Technologies (CONIT)*, 1-6.

[46] Samanta, D., Alahmadi, A. H., Karthikeyan, M. P., Khan, M. Z., Banerjee, A., Dalapati, G. K., & Ramakrishna, S. (2021). Cipher block chaining support vector machine for secured decentralized cloud enabled intelligent IoT architecture. *IEEE Access*, *9*, 98013-98025.

[47] Semal, B., Markantonakis, K., Akram, R. N., & Kalbantner, J. (2020). A study on microarchitectural covert channel vulnerabilities in infrastructure-as-a-service. *In Applied Cryptography and Network Security Workshops: ACNS 2020 Satellite Workshops, AIBlock, AIHWS, AIoTS, Cloud S&P, SCI, SecMT, and SiMLA, Rome, Italy*, 360-377.

[48] Sharma, P., Lee, S., Guo, T., Irwin, D., & Shenoy, P. (2017). Managing risk in a derivative IaaS cloud. *IEEE Transactions on Parallel and Distributed Systems*, *29*(8), 1750-1765.

[49] Singh, A. K., & Sharma, S. D. (2019). High Performance Computing (HPC) Data Center for Information as a Service (IaaS) Security Checklist: Cloud Data Governance. *Webology*, *16*(2), 83-96.

[50] Singh, D., Sinha, S., & Thada, V. (2022). A novel attribute based access control model with application in IaaS cloud. *Journal of Integrated Science and Technology*, *10*(2), 79-86.

[51] Svatá, V., & Zbořil, M. (2020). Areas of focus for cloud security providers assessment. *In IEEE 10th International Conference on Advanced Computer Information Technologies (ACIT)*, 806-810.

[52] Tao, X., Wang, L., Xu, Z., & Xie, R. (2021). Scams: A novel side-channel attack mitigation system in iaas cloud. *In IEEE MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM)*, 329-334.

[53] Wang, L. M., Liang, C., Lu, X., Xia, C., Morgan, J., Willey, W., & Miskell, T. (2022). Design of a Live Networking Device Update Mechanism For Cloud Computing Systems. *In IEEE International Conference on Networking, Architecture and Storage (NAS)*, 1-6.

[54] Wichep, J., Winai, W., Thanathorn, P., & Thongrob, A. (2020). A Logging System in OpenStack Environment to Mitigate Risks Associated with Threats in Infrastructure as a Service Cloud. *ICIC Express Letters*, *14*(4), 387-397.

[55] Yin, X., Chen, X., Chen, L., Shao, G., Li, H., & Tao, S. (2018). Research of security as a service for VMs in IaaS platform. *IEEE Access*, *6*, 29158-29172.

[56] Zimba, A., Hongsong, C., & Zhaoshun, W. (2017). Edge aggregation based bayesian modeling of cyber attacks in hypervisor-enabled IAAS cloud networks. *In IEEE 17th International Conference on Communication Technology (ICCT)*, 1312-1317.

[57] Namboodiri, A. S., Sanodiya, R. K., & Arun, P. V. (2023). Remote Sensing Cloud Removal using a Combination of Spatial Attention and Edge Detection. *In IEEE 11th International Symposium on Electronic Systems Devices and Computing (ESDC)*, *1*, 1-6. https://doi.org/10.1109/ESDC56251.2023.10149875.

[58] Guo, P., Yan, Y., Zhang, F., Zhu, C., Zhang, L., & Dai, Z. (2023). Extending the classical side-channel analysis framework to access-driven cache attacks. *Computers & Security*, *129*, 103255. https://doi.org/10.1016/j.cose.2023.103255.

[59] Duncan, A., Creese, S., Goldsmith, M., & Quinton, J. S. (2013). Cloud computing: Insider attacks on virtual machines during migration. *In 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, 493-500. https://doi.org/10.1109/TrustCom.2013.62.

[60] Aburukba, R., Kaddoura, Y., & Hiba, M. (2022). Cloud Computing Infrastructure Security: Challenges and Solutions. *In IEEE International Symposium on Networks, Computers and Communications (ISNCC)*, 1-7.

[61] Barrowclough, J. P., & Asif, R. (2018). Securing cloud hypervisors: a survey of the threats, vulnerabilities, and countermeasures. *Security and Communication Networks*, *2018*(1), 1681908. https://doi.org/10.1155/2018/1681908

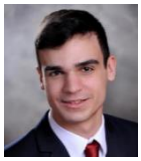
[62] Rakotondravony, N., Taubmann, B., Mandarawi, W., Weishäupl, E., Xu, P., Kolosnjaji, B., & Reiser, H. P. (2017). Classifying malware attacks in IaaS cloud environments. *Journal of Cloud Computing*, *6*, 1-12.

[63] Mthunzi, S. N., Benkhelifa, E., Bosakowski, T., Guegan, C. G., & Barhamgi, M. (2020). Cloud computing security taxonomy: From an atomistic to a holistic view. *Future Generation Computer Systems*, *107*, 620-644.

[64] Shin, D., Akkan, H., Claycomb, W., & Kim, K. (2011). Toward role-based provisioning and access control for infrastructure as a service (IaaS). *Journal of Internet Services and Applications*, *2*, 243-255.
[65] Sultan, N. H., Varadharajan, V., Zhou, L., & Barbhuiya, F. A. (2022). A role-based encryption (rbe) scheme for securing outsourced cloud data in a multi-organization context. *IEEE Transactions on Services Computing*, *16*(3), 1647-1661.
[66] Capital One Data Breach Compromises Data of Over 100 Million, July 2019, https://www.nytimes.com/2019/07/29/business/capital-one-data-breach-hacked.html
[67] Microsoft Data-Exposure Incident Highlights Risk of Cloud Storage Misconfiguration, October 2020, https://www.darkreading.com/cloud-security/microsoft-data-exposure-incident-highlights-risk-of-cloud-storage-misconfigurations
[68] Cloudbleed Triggered 1.2M Times, Damage Kept to Minimum, March 2017, https://threatpost.com/cloudbleed-triggered-1-2m-times-damage-kept-to-minimum/124023/
[69] NIST SP 800-57 Part 1 Rev. 5, May 2020, https://csrc.nist.gov/pubs/sp/800/57/pt1/r5/final
[70] Key Management in Cloud Services, Sep 2020, https://cloudsecurityalliance.org/artifacts/key-management-when-using-cloud-services
[71] NIST to Withdraw Special Publication 800-67 Revision 2, June, 2023, https://csrc.nist.gov/News/2023/nist-to-withdraw-sp-800-67-rev-2
[72] NIST Recommendation for Block Cipher Modes of Operation, December 2001, https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-38a.pdf
[73] The Year Cryptojacking Ate the Web, December 2018, https://www.wired.com/story/cryptojacking-took-over-internet/
[74] EternalBlue Exploit: What It Is and How It Works, May 2019, https://www.sentinelone.com/blog/eternalblue-nsa-developed-exploit-just-wont-die/
[75] Monitor and maintain Windows Server IaaS Virtual Machine, https://learn.microsoft.com/en-us/training/paths/monitor-maintain-windows-server-iaas-virtual-machine/
[76] Performing a Rolling Update, https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/
[77] 6% of all Google Cloud Buckets are vulnerable to unauthorized access, March 2022, https://www.comparitech.com/blog/information-security/google-cloud-buckets-unauthorized-access-report/
[78] Akamai State of the Internet Security Report, February, 2019 https://www.akamai.com/newsroom/press-release/state-of-the-internet-security-retail-attacks-and-api-traffic
[79] An IaaS Security Checklist, August 2024, https://github.com/sztaki-hu/iaas-sc

Authors Biographies

Mihály Héder, is a senior research fellow at the Department of Network Security and Internet Technologies at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN). He specializes in Trust & Identity: the Authentication and Authorization Infrastructures in the research and education sector.



Domonkos Baczó, is a computer scientist with an emphasis on web-based Software Development. Baczó has international experience in R&D on the field of Trust & Identity. He spent his engineering internship at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN).

Tibor Kovács, is a developer who focuses on object-oriented software development such as Java and APS.net He fulfilled his engineering internship at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN).

Ernő Rigó, is the Head of the Department of Network Security and Internet Technologies at the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN). As an ISACA Certified Information Systems Auditor (CISA) and ISC2 Certified Information Systems Security Professional (CISSP), his PhD research is focused on security evaluation of dynamic cloud infrastructures. He is the lead of HunCERT, a sectoral security competence hub of Hungarian ISPs.