



Article

Exploiting Misconfiguration Vulnerabilities in Microsoft's Azure Active Directory for Privilege Escalation Attacks

Ibrahim Bu Haimed ¹, Marwan Albahar ^{2,*}  and Ali Alzubaidi ² 

¹ School of Computing Science, University of Newcastle, Newcastle upon Tyne NE1 7RU, UK; buhaimedi2@gmail.com

² Department of Computer Science, Umm Al Qura University, P.O. Box 715, Mecca 24382, Saudi Arabia; mabahar/aakzubaidi@uqu.edu.sa

* Correspondence: mabahar@uqu.edu.sa

Abstract: Cloud services provided by Microsoft are growing rapidly in number and importance. Azure Active Directory (AAD) is becoming more important due to its role in facilitating identity management for cloud-based services. However, several risks and security issues have been associated with cloud systems due to vulnerabilities associated with identity management systems. In particular, misconfigurations could severely impact the security of cloud-based systems. Accordingly, this study identifies and experimentally evaluates exploitable misconfiguration vulnerabilities in Azure AD which can eventually lead to the risk of privilege escalation attacks. The study focuses on two scenarios: dynamic group settings and the activation of the Managed Identity feature on virtual devices. Through experimental evaluation, the research demonstrates the successful execution of these attacks, resulting in unauthorized access to sensitive information. Finally, we suggest several approaches to prevent such attacks by isolating sensitive systems to minimize the possibility of damage resulting from a misconfiguration accident and highlight the need for further studies.

Keywords: Azure Active Directory; cloud services; misconfiguration vulnerabilities; privilege escalation attacks; identity management; cloud-based systems



Citation: Haimed, I.B.; Albahar, M.; Alzubaidi, A. Exploiting Misconfiguration Vulnerabilities in Microsoft's Azure Active Directory for Privilege Escalation Attacks. *Future Internet* **2023**, *15*, 226. <https://doi.org/10.3390/fi15070226>

Academic Editor: Ishaani Priyadarshini

Received: 21 May 2023
Revised: 13 June 2023
Accepted: 20 June 2023
Published: 23 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the past few years, cloud computing has revolutionized the way institutions develop and provide their services. In today's mission-critical business requirements, cloud technology plays an important role. Cloud computing provides effortless, on-demand access to a shared pool of computing resources, while requiring little management complexity [1]. With the advent of cloud computing, it has become possible to outsource the entire IT infrastructure to a service provider. As a result, cloud solutions have become the industry standard for running IT systems in a range of organizations, from small businesses to large corporations. Consequently, several organizations find it appealing to move from on-premises computing to cloud computing, either fully or partially, which has led to increased competition between service providers [2].

Cloud computing has emerged as a rapidly growing sector in the IT industry, offering numerous benefits such as cost efficiency, flexibility, accessibility, scalability, and increased productivity. Projections indicate that the demand for cloud computing services is expected to reach approximately \$791.48 billion by 2028 [3]. However, alongside its potential, there are significant risks and threats associated with cloud services. The storage of confidential data in multi-cloud environments exposes organizations to a higher risk of attacks. This concern is particularly relevant for businesses that are still considering a migration to cloud computing, as it poses a potential threat to their transformation efforts.

Cloud computing enables efficient data management at a reasonable cost, with flexible pricing options tailored to customer requirements and the expertise of service providers. Price plays a crucial role in organizations providing cloud-based services, as it directly

affects customer requirements and company profitability. Pricing decisions have a direct impact on the economy, shares, profits, and losses [4]. Service vendors prioritize delivering guaranteed quality of service (QoS) to their customers, ensuring customer satisfaction and meeting service level agreements (SLAs). While pricing in the technology industry has traditionally followed a set framework, the advent of cloud computing has brought about new models of value chains, leading to the evolution of pricing models in this domain [5]. Overall, statistical data and figures indicate a growing trend among companies in adopting Microsoft cloud services, encompassing Microsoft Office 365, SharePoint for collaborative purposes, Microsoft Office applications, and various other services [6]. Microsoft Azure Active Directory (AAD) serves as a cloud-based identity and access management solution offered by Microsoft, delivering Identity and Access Management (IAM) services through the cloud [7]. Moreover, AAD can be leveraged for authentication purposes across different cloud environments. AAD shares similarities with Windows Active Directory (AD), which operates as a server-based service, running on Windows Server [6].

Despite the advantages of cloud computing, the storage of sensitive data with a third party, such as a cloud service provider, raises significant security concerns. Security is a crucial requirement for both cloud providers and their customers [8]. Cloud systems, due to their accessibility and complexity, are associated with various risks and security issues. The National Security Agency (NSA) has identified cloud misconfigurations as a prominent cause of security breaches [9,10]. These misconfigurations create opportunities for malicious actors to exploit vulnerabilities or obtain user credentials through phishing campaigns or social engineering techniques, thereby enabling privilege escalation attempts [11]. Misconfigurations can have severe consequences, including the compromise of sensitive data and privilege escalation attacks [9].

The focus of this paper is on Microsoft Azure cloud services, specifically highlighting its security architecture. A distinctive cloud security component provided by Microsoft Azure is Active Directory (AD), a cloud-based identity and access management service [6,7]. Consequently, this research contributes the following:

1. The study identifies exploitable misconfiguration vulnerabilities in Microsoft Azure Active Directory (Azure AD), specifically focusing on privilege escalation attacks. Two scenarios are examined: misconfigurations in dynamically assigned identity groups and risks associated with managed identities on virtual machines.
2. Rigorous experimental evaluations are conducted to demonstrate the successful execution of privilege escalation attacks resulting from the identified misconfigurations. The evaluations utilize a common configuration scoring system to analyze and assess the outcomes, providing empirical evidence of the risks and consequences associated with these vulnerabilities.
3. The research provides detailed insights into the identified misconfiguration vulnerabilities, highlighting their potential impact on the security of cloud-based systems. These insights contribute to a better understanding of the risks involved and provide a foundation for enhancing the security measures in Azure AD and similar cloud platforms.

The rest of the paper is organized as follows: Section 2 covers the related works of cloud computing, security concepts and Microsoft Azure. Section 3 describes the research methodology. Section 4 describes the design and configuration of the cloud environment used in the study. The detailed results are presented in Section 5. Discussion and conclusion are presented in Sections 6 and 7, respectively.

2. Related Work

2.1. Adaptability and Shared Accountability

Takabi et al. [12] examined the sharing of cloud services between customers and vendors in terms of privacy and security. The researchers pointed out that customers using Software as a Service (SaaS) have limited extensibility because vendors typically offer solutions with a wide range of integrated functionalities. Consequently, providers bear

a greater responsibility in ensuring the confidentiality of software solutions, particularly in public cloud environments, where client organizations may have stringent security requirements and can enforce necessary protection measures. In certain cases, private clouds may necessitate additional customization to accommodate specific needs.

Furthermore, Platform as a Service (PaaS) primarily aims to empower developers to build their own applications on top of the platform. Therefore, customers hold the main responsibility for safeguarding the software they create and deploy on these platforms. As a result, suppliers have the complete responsibility of segregating the applications and workspaces of their clients.

2.2. Threats and Vulnerabilities

In [13], the author investigates potential threats targeting cloud computing systems. The researcher highlights the significance of securing high-authority accounts, such as those belonging to programmers or technical operators, as their compromise could have catastrophic consequences. An illustrative example is the recent breach of sensitive information concerning 80 million customers of Anthem, Inc. Moreover, the researcher observes that customers often overlook security measures and best practices when configuring cloud services, including the assignment of privileges to each service and identity. Additionally, users generally lack comprehensive knowledge about cloud resources and heavily rely on service providers. However, it is important to note that service providers are typically bound by the terms of their agreement with the customer and do not offer services beyond the agreed-upon scope.

Furthermore, the researcher discusses the risk of human error, which is relevant to all aspects of the information technology sector, including cloud computing. Both users and system administrators, despite their expertise, can inadvertently commit errors that may result in a compromise of the entire system [14]. However, the study does not specifically address the possibility of these risks occurring within the context of SaaS models such as Azure Active Directory. Additionally, the scientific paper does not explore whether these factors contribute to privilege escalation attacks.

2.3. Security Pattern

A recent scientific paper by Rath et al. [15] discusses security patterns in SaaS-based cloud systems. These patterns involve recognizing recurring problems and identifying corresponding solutions [16]. The researchers have divided the patterns into several sections and described the security issues associated with each section. Particularly, several problems have been raised within the categories of authentication, authorization, and identification. One of these issues pertains to the secure authentication of users in cloud computing. The authors proposed a solution which involves the use of three-factor authentication. Three-factor authentication is the most commonly method used to enhance user authentication. It consists of the user knowing something (e.g., their username and password), possessing something (e.g., a mobile phone), and providing user biometric information (e.g., a fingerprint). In most cases, only two factors are applied, and a third factor is required when accessing sensitive information. Although this solution is effective, its success heavily relies on the user's decision-making.

Another problem discussed is access token management and security. Access tokens play a critical role in securing various operations that require specific permissions, such as managed identity and service principals. To ensure the security and proper management of access tokens, a proposed solution involves specifying an activation time and granting specific permissions that do not exceed the required level. While this solution is effective, its implementation can be challenging, particularly when access tokens need to always remain active. Furthermore, defining the permissions is crucial, but it does not prevent these privileges from being exploited, even if they are limited, leading to a potential escalation of privileges in the hands of intruders.

3. Research Methodology

For this study, a quantitative research method was employed, which involved collecting and analyzing numerical data to draw conclusions and make statistical inferences. Additionally, an experimental systems research approach was adopted, and the results were evaluated using the Common Configuration Scoring System (CCSS). Figure 1 presents an overview of the research methodology employed in this study.

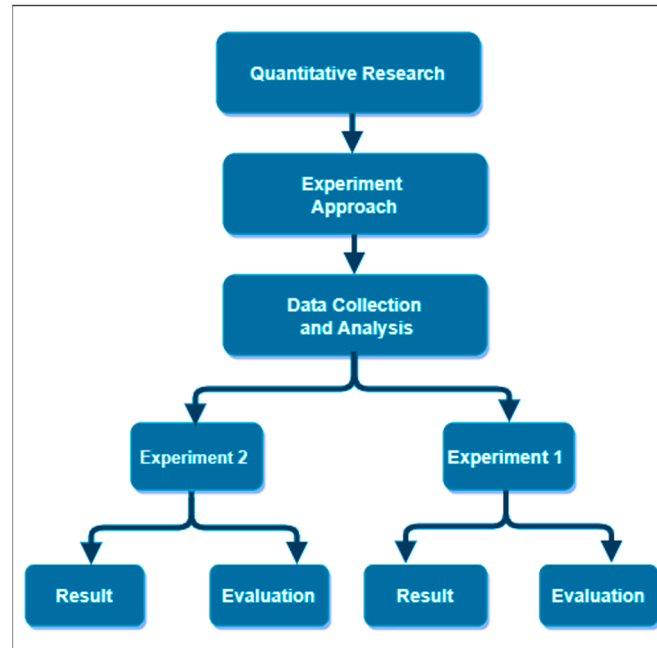


Figure 1. An overview of the research methodology.

3.1. Experimental Research

In this experimental research study, our primary objective was to investigate the misconfiguration vulnerabilities present in Microsoft Azure Active Directory (Azure AD). To achieve this, we utilized Microsoft Azure as the underlying infrastructure to conduct our experiments. Two separate experiments were carefully designed and executed on this Azure-based infrastructure. Prior to conducting the experiments, a comprehensive exploration of the features, applications, and benefits of Microsoft Azure was undertaken to ensure a solid foundation for our research. This knowledge guided the development of the experiment designs and the creation of distinct attack paths and configurations for each experiment. The aim was to capture the diverse aspects and potential risks associated with misconfigurations. We present a detailed analysis and comparison of the two experiments, highlighting their specific variations and characteristics, in Table 1.

Table 1. The differences between the two experiments.

Experiment Elements	Experiment 1	Experiment 2
The type of attack	Horizontal privilege escalation	Vertical privilege escalation
The objects that were abused in the attack	Dynamic group users	VM-02 Managed Identity users
Pre-attack privileges of the hacker	Reader role on the subscription	Reader role on the subscription Virtual Machine Contributor role on the VM-02 machine Virtual Machine Administrator Login role on the VM-02 machine

Table 1. Cont.

Experiment Elements	Experiment 1	Experiment 2
Hacker privileges after attack	Virtual Machine Administrator Login role on all VMs	Access to Managed identity virtual machine with an owner role on the subscription level

3.2. Evaluation Frameworks

In the realm of cloud computing security evaluation, the utilization of effective frameworks is essential to accurately assess security levels and identify potential vulnerabilities. To this end, numerous evaluation frameworks have been employed for analyzing and measuring different aspects of cloud security. Each framework offers unique perspectives and approaches to assess the security of cloud systems.

1. Information Security Risk Management Framework for Cloud Computing Environments: This framework, described in [17], focuses on identifying threats to cloud computing security, integrity, availability, and management. It provides a comprehensive overview of security levels in cloud systems. However, it does not evaluate the most crucial factor for this study, which is the severity of configuration errors.
2. Common Vulnerability Scoring System (CVSS): CVSS is a framework for calculating a qualitative estimate of system security flaws [18]. In this framework, vulnerabilities are assessed based on a fixed formula, which allows organizations to evaluate their system's vulnerability severity. A wide range of businesses, governments, and other organizations use CVSS.
3. Common Configuration Scoring System (CCSS): In terms of the factors it measures, this framework is based on the CVSS framework, with a few modifications, to accurately measure the severity of misconfiguration threats. A downside of this standard is that it is less widely adopted than CVSS. However, it is based on CVSS factors, which makes it reliable [18]. As a result, this framework was adopted in the evaluation process of this study.

4. Design and Implementation

4.1. Design

In accordance with Microsoft's policy [19], individuals are permitted to conduct Azure vulnerability assessments and penetration testing without involving Microsoft, provided they own the system or possess legal authorization. Consequently, in this study, an infrastructure will be created to simulate a medium-sized company's environment, enabling authorized penetration testing and necessary experiments. The infrastructure will replicate that of a game programming company, specifically designed to facilitate the authorized testing. Only essential components required for the experiment will be created, built, and prepared. For instance, the experiment will involve the creation of two virtual machines, as the scope of the experiment does not necessitate a larger number. It is important to note that, in real-world infrastructures, the number of virtual machines would typically be greater.

The company chosen for this simulation is Tala Game, a video game development company. Tala Game specializes in developing complex video games that involve the collaboration of multiple programmers. Additionally, the company invests in independent game developers worldwide, providing them with the necessary equipment, such as access to virtual machines, in exchange for a percentage of the game's revenue. To ensure flexibility and easy access to resources for its employees, Tala Game relies on Microsoft's cloud computing services.

The following is an outline of the design process. The first step is to create a free account on the Microsoft Azure website, which will serve as the admin account for creating the lab. The second step involves activating Active Directory Premium 2, as Active Directory plays a crucial role in the experiment. Active Directory Premium 2 provides enhanced functionality, including identity management and dynamic groups. Identity management

allows granting permissions to objects, such as virtual machines, without requiring a username and password. Dynamic groups automate tasks, such as adding users to a group automatically when specific conditions are met. The third step is to create two accounts with different permissions, serving as the starting point for the attack. Both accounts have read-only permissions, while the VM-Reader account has access to one of the virtual machines. Finally, two virtual machines with distinct features are created, and managed identities with owner roles in the subscriptions are activated for VM-02. The VM-Reader user is granted access to this virtual machine. The subsequent section will provide detailed coverage of all the permissions assigned to the users and virtual machines, followed by a diagram illustrating the design, as depicted in Figure 2.

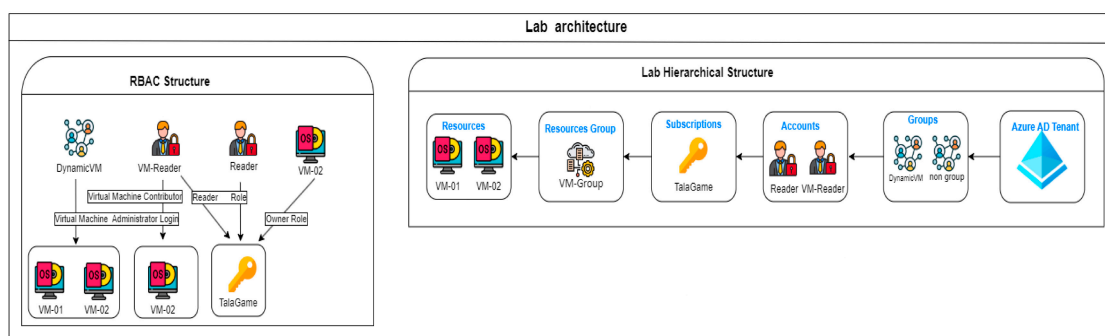


Figure 2. Lab architecture.

4.2. Implementation

This section explains how to create the cloud infrastructure that will be used in this study using Microsoft Azure.

Setting up an Azure account:

Create an account on Microsoft Azure using the following link: (<https://portal.azure.com>). Microsoft provides \$200 for each new account, which is valid for a month; this amount may be sufficient to cover the cost of building the infrastructure, initially [20]. After creating an account in Azure, a tenant is automatically created, and the account used is the global administrator.

5. Experiment and Results

This section presents the experimental methodology developed to address the research objectives. The purpose of the experiment is explained, including the research questions it aims to answer and the criteria used to assess its success. Furthermore, this section outlines the exploitation of incorrect settings in each scenario, detailing the tools employed, the resulting outcomes, and potential solutions. The focus of the experiment is on the fourth step of the MITRE ATT&CK Framework, which involves privilege escalation. The aim is to simulate the attacker's mindset and replicate real-world scenarios. It is assumed that the initial three steps of the framework, involving gaining initial access, executing the attack, and establishing persistence, have already been accomplished successfully by the attacker.

5.1. Assumption

In this experiment, we made a few assumptions. Firstly, it was assumed that the system was built using Microsoft products available on Azure to simulate the infrastructure of a small or medium-sized organization. Additionally, it was assumed that the attacker was able to implement the initial access tactic from the MITRE ATT&CK Framework. It is worth noting that, prior to selecting the most effective attack method, actual attackers conduct a substantial amount of reconnaissance within the system.

5.2. The First Experiment: Exploit the Dynamic Groups Settings to Escalate Privileges

5.2.1. Explanation

In this scenario, the administrator wishes to automate the process of managing the accounts of independent game developers. For this reason, several steps have been taken by the system administrator. As a first step, the system administrator creates a dynamic group and adds a condition that automatically adds users whose accounts contain the word “indep”. In the second step, he gave this group a role that allows all the users in the group to log on to the virtual machines, which is “Virtual Machine Administrator login”. As a final step, the system administrator has configured virtual machines for users.

A dynamic group configuration fault will be exploited in this experiment to simulate the horizontal privilege escalation attack. The following is the path of attack: First, assume the hacker was able to obtain a minimally privileged user (reader) role. As a second step, the hacker will attempt to gather as much information as possible about the organization, such as a list of users and groups, along with their roles. Through the analysis of the collected information, the hacker will be able to discover the misconfiguration, which will enable him to add an external user to the dynamic group and take advantage of the privileges assigned to that group. Due to the successful elevation of privilege attack, the hacker will have gained access to the virtual machines.

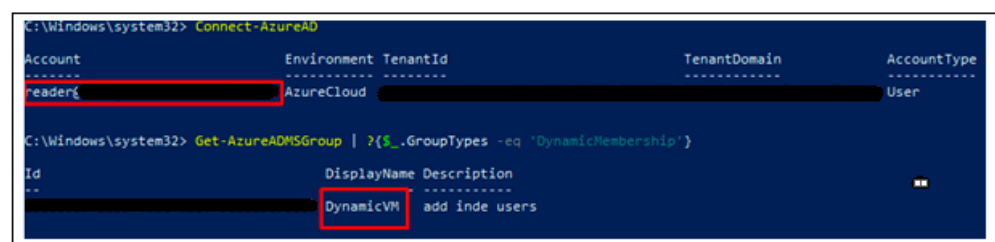
5.2.2. Aim of the Experiment

This experiment aims to demonstrate the dangers associated with the misconfiguration of automation tasks in general, especially in cases where users can directly or indirectly input values into the system. There are two conditions that must be met to consider this experiment successful. The first aspect is the successful escalation of power, either vertically or horizontally. Second, the focus should be on the flaws caused by the incorrect settings, and not on any software vulnerabilities.

5.2.3. Exploit the Dynamic Groups

The purpose of this section is to describe the steps necessary to exploit the dynamic group setup error to perform a privilege escalation attack. The process begins by exploring the system with the PowerShell Azure AD Module, using the reader account. The “Connect-AzureAD” command is used for authentication. After that, the following command will provide a list of all dynamic groups. As shown in Figure 3, the display names of all dynamic groups are retrieved.

```
Get-AzureADMSGroup | ?{$_GroupTypes -eq 'DynamicMembership'}
```



```
C:\Windows\system32> Connect-AzureAD
Account Environment TenantId TenantDomain AccountType
-----
reader AzureCloud
C:\Windows\system32> Get-AzureADMSGroup | ?{$_GroupTypes -eq 'DynamicMembership'}
Id DisplayName Description
--
DynamicVM add inde users
```

Figure 3. The dynamic group enumeration.

The next step is to determine the permissions granted to this group. By using Az PowerShell, it is possible to determine this information using the following command:

```
Get-AzRoleAssignment | Where-Object {$_.DisplayName -like 'DynamicVM'}
```

Figure 4 illustrates that the group has the role “Virtual Machine Administrator Login” at the subscriber level. As a result of this role, all users in this group can log on to virtual machines with local administrative privileges.

```
C:\Users\buhaimeidi> Get-AzRoleAssignment | Where-Object {$_.DisplayName -like 'DynamicVM'}

RoleAssignmentName : 5931fc28-3404-44bd-9bac-8b5f9597bb1e
RoleAssignmentId   : /subscriptions/.../providers/Microsoft.Authorization/...
Scope              : /subscriptions/...
DisplayName        : DynamicVM
SignInName         :
RoleDefinitionName : Virtual Machine Administrator Login
RoleDefinitionId   : 1c0163c0-47e6-4577-8991-ea5c82e286e4
ObjectId           : 1b2660d7-a259-498e-9baf-90825967e867
ObjectType         : Group
```

Figure 4. The dynamic group role.

The attacker can assume or predict the functionality of the group based on the name and description, but the conditions that must be met to join this group are unknown. One of the methods used to discover the conditions is to analyze the users in the group and try to find out what they all have in common. This command in AzureAD displays all users in a specified group:

```
Get-AzureADGroupMember -ObjectId [ Group ObjectID]
```

where the ID is replaced by the group number that was extracted using the previous command. As shown in Figure 5, the previous command’s output provides critical information. The first information is that all users share an “indep” word in their e-mail. The second is that there is a guest user, which means that an external user can be invited to the system.

```
C:\Windows\system32> Get-AzureADGroupMember -ObjectId 1b2660d7-...

ObjectID      DisplayName UserPrincipalName      UserType
-----
...          user-indep  user-indep@...          Member
...          user2-indep user2-indep@...         Member
...          alex_indep  alex_indep_mail.com#EX Guest
...          indep      pen-indep@...          Member
```

Figure 5. Enumeration of the users.

After analyzing the previous information, the hacker will most likely attempt to add a user via the invite feature. Any e-mail from any service provider can be used, but the username must contain an “indep” word; the following e-mail will be used in the next step: “ahmad-indep@hotmail.com”. It is possible to send invitations through Azure Portal or Azure AD Module by using the following command:

```
New-AzureADMSInvitation -InvitedUserEmailAddress ahmad-indep@hotmail.com -SendInvitationMessage $True -InviteRedirectUrl "http://myapps.microsoft.com"
```

Upon successful completion of the process, an e-mail will be sent, containing an account activation link, which will be used to prepare the account and password. From the outputs of the following command, the user “ahmad-indep@hotmail.com” has been added to “DynamicVM”, indicating that the escalation process was successful. The outputs of the command are shown in Figure 6.

```
Get-AzureADGroupMember -ObjectId [Object Id] | Where-Object {$_.UserPrincipalName -like '*ahmad-indep_hotmail*'}
```

```
C:\Windows\system32> Get-AzureADGroupMember -ObjectId ... | Where-Object {$_.UserPrincipalName -like '*ahmad-indep_hotmail*' }

ObjectID      DisplayName UserPrincipalName      UserType
-----
ba3e54f3-91d7 ahmad indep ahmad-indep_hotmail.com      Guest

C:\Windows\system32> Get-AzureADUserMembership -ObjectId ba3e54f3-91d7-...

ObjectID      DisplayName Description
-----
DynamicVM    add indep users
```

Figure 6. Enumeration of the external user.

The final step is to demonstrate Ahmad’s ability to execute commands on virtual systems. It is possible to execute commands remotely on one of the virtual systems using the PowerZure framework. First, log in with the “ahmad-indep@hotmail.com” account, and then execute a command to list all existing virtual machines. Finally, execute commands on one of the VMs. This process is done using the following commands. The outputs of the previous steps are shown in Figure 7.

```
Connect-AzAccount
Get-AzVM
Invoke-AzureRunCommand -VMName [virtual machine name] -Command [ the command]
```

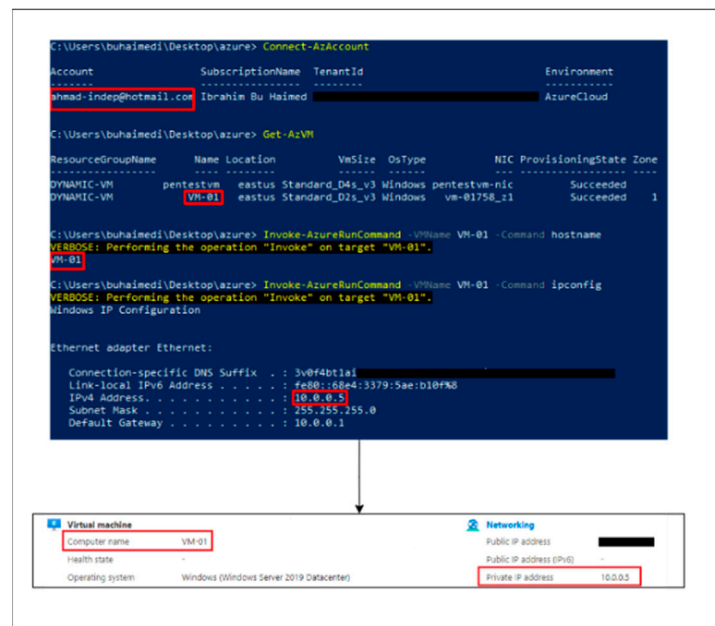


Figure 7. The process to execute command on VM.

5.2.4. Results

Before successfully executing the attack, the hacker took several steps. Initially, the hacker gained access to the user “Reader” with a reader role. A dynamic group called “DynamicVM” was discovered by the hacker, which has access to virtual machines. Due to the fact that dynamic groups are used to add users automatically according to predefined conditions, the hacker was able to discover the condition by analyzing the information of the members of this group. Using the “invite guest” feature, the hacker exploited the dynamic group settings through the Internet, without the need to access the internal network to carry out the attack. This makes this attack average in complexity. Moreover, all users have the guest invite feature enabled by default; it can be disabled, so this attack requires access to a user who has this feature enabled. The hacker created an email with the word “indep” in the address. Consequently, this address was added to the dynamic group as soon as it was invited as a guest. Due to this, the hacker could access all virtual machines using the guest account. By reading, writing, and modifying files and information on these devices, the hacker is partially violating the confidentiality and integrity of information in the organization. Considering that the hacker did not obtain the owner role in the system, this attack can be classified as a horizontal privilege escalation. As this attack relies on user attributes that can be controlled by users, and because the user invitation feature is enabled by default, it may be reproducible. The steps taken during this attack are summarized in Figure 8.

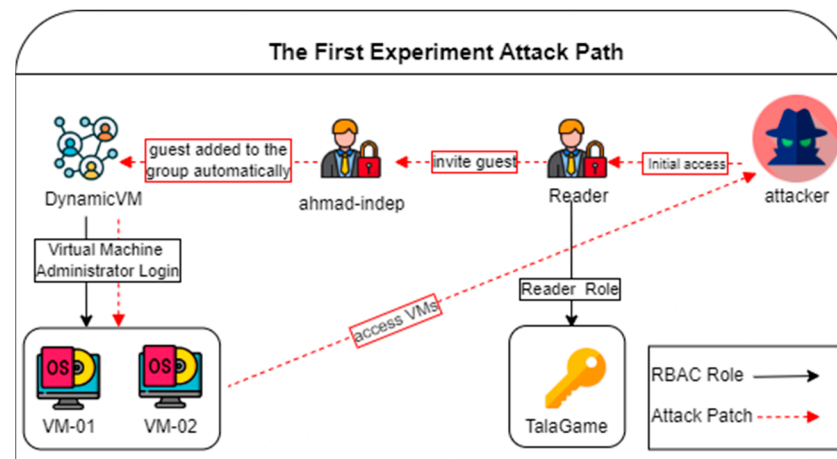


Figure 8. Description of the process of attack execution.

5.2.5. Recommendation

Several suggestions can be made to prevent this type of attack. As a first step, avoid relying on users’ controllable attributes in dynamic group conditions. Second step: prevent users from inviting external guests. If this feature is required, specific users are given this option, and only users from specific domains are allowed to be invited. In addition, all external users must be closely monitored.

5.3. The Second Experiment: Exploit Managed Identity Settings to Escalate Privileges

5.3.1. Explanation

In this scenario, “VM-reader” requested activation of the Managed Identity feature for his virtual machine so that he could accomplish his daily tasks. This feature allows the virtual machine to authenticate to cloud services without providing credentials. The following is an explanation of how the system administrator enabled the managed identity and configured the permissions needed to achieve the user’s desire. First, managed identity has been enabled on the VM-02 machine. Secondly, the VM-02 device has been given the validity of “Owner” at the level of subscriptions to ensure that it is able to connect to the cloud services as it should. Finally, the system administrator assigned the roles “Virtual Machine Contributor” and “Virtual Machine Administrator Login” to “VM-reader” and set the scope for VM-02 VM only, to prevent other virtual machines from being accessed using these permissions. Additionally, the account has a reader role at the level of subscriptions.

This experiment will simulate a vertical privilege escalation attack by exploiting a managed identity misconfiguration, based on the assumption that the attacker has access to the “VM-Reader” account. In the same way as in the previous scenario, the hacker collected and analyzed the infrastructure data of the organization. Using this information, he was able to determine that the “VM-reader” account had access to the “VM-02” machine, which, in turn, had owner access at the subscription level. As a next step, the hacker attempted to log in or execute a command on the virtual machine to obtain the token that would make it possible for him to log in using the virtual machine’s identity. Finally, the attacker will attempt to log in with the “VM-02” identity to escalate its permissions to “owner” and access sensitive information, such as storage keys.

5.3.2. Aim of the Experiment

Managed identities present some of the risks that this experiment seeks to clarify. Furthermore, if the managed identity that a user can access is not adequately evaluated, it may be exploitable. Two conditions must be met for this experiment to be considered successful. The first aspect is the successful vertical or horizontal escalation of power. Second, rather than focusing on software vulnerabilities, the emphasis should be on flaws caused by incorrect settings.

in, while the fourth command is used to extract sensitive information from the storage. An illustration of the difference between using VM-reader and VM-02 tokens to extract sensitive information can be found in Figure 12.

```

$MIToken = "[ access token]"
$MIID = "[ Managed identity ID]"
Login-AzAccount -AccessToken $MITOKEN -AccountId $MIID
Get-AzPasswords -Verbose | Out-GridView
    
```



Figure 12. An illustration of the difference between trying to extract sensitive information from the storage before and after using the virtual machine’s identity.

5.3.4. Results

The hacker had to take several steps before he could successfully execute the attack. Firstly, the hacker obtained access to the user account “VM-Reader” with the “reader” role assigned at the subscription level. Additionally, this user possessed the roles of “Virtual Machine Contributor” and “Virtual Machine Administrator Login” on the virtual machine “VM-02”. The Identity Management feature was observed to be activated in the virtual machine, granting owner-level rights at the subscriber level. Exploiting this misconfiguration, the hacker managed to steal the virtual machine’s identity, thereby gaining unauthorized access to sensitive information.

To carry out this type of attack, the attacker must initially acquire local access to the targeted virtual machine. Subsequently, the attacker needs to grant administrative privileges to the virtual machine. If these two conditions are met, the attack can be replicated on a virtual machine with managed identity enabled. By executing this attack, the hacker gains access to the identity of the virtual machine possessing the owner role. With this elevated level of permission, the hacker can completely compromise the confidentiality and integrity of subscribers’ data. This experience resulted in the successful execution of the vertical privilege escalation attack. The stages involved in carrying out the attack are summarized in Figure 13.

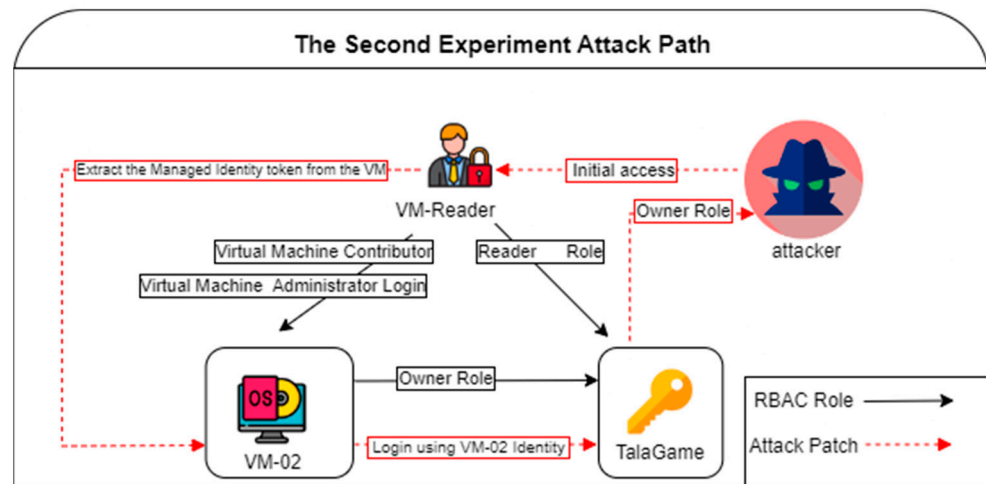


Figure 13. An overview of how the attack is executed.

5.3.5. Remediation

To enhance the security of identity management in Azure AD, the following remediations are proposed: conducting a necessity evaluation to determine when identity management is essential, implementing strict controls and monitoring mechanisms for access privileges, adopting a scope-based permission model, limiting local admin privileges on virtual machines, and establishing ongoing monitoring and compliance practices. Implementing these recommendations will mitigate misconfiguration risks and unauthorized access, strengthening the overall security of identity management in Azure AD.

6. Evaluation and Discussion

In this section, CCSS will be used to evaluate the study and determine the severity of each misconfiguration vulnerability. Further, the reasons for determining the values for each factor will be explained. These steps will be based on what was discussed in Section 3. The factor is also the result of what was extracted from each experiment in the Results section.

6.1. Evaluation of the First Experiment

Initial results of abuse in the dynamic groups experiment will be presented regarding configuration vulnerabilities as assessed using the CCSS framework. Table 2 contains all the factors and an explanation of why each value was chosen. Additionally, the results of the calculation of Exploitability, Impact, and Base are shown.

Table 2. The first experiment’s evaluation values.

Factor	Value	Score	Explanation
Access Vector (AV)	Network (N)	1.0	Hackers could invite external users without accessing the internal network or executing any commands locally.
Access Complexity (AC)	Medium (M)	0.61	For the hacker to invite an external user, he must have some permissions and an active user account.
Authentication (Au)	Single (S)	0.56	A single login attempt is sufficient to exploit this vulnerability.
Confidentiality Impact (C)	Partial (P)	0.275	As a result of the attack, the hacker was able to access all the virtual devices used in programming games, which may contain confidential data, but he could not access all the data stored in the company’s cloud.

Table 2. *Cont.*

Factor	Value	Score	Explanation
Integrity Impact (I)	Partial (P)	0.275	As a result of the attack, the hacker gained access to all virtual machines used in game programming, enabling him to modify any file or attach malicious code to a game.
Availability Impact (A)	None (N)	0.0	No service was directly disrupted because of this attack.
Exploitation Method (EM)	Exploited Actively (A)	non	The hacker was able to access parts of the system after carrying out the attack.
Exploitability result	6.9		
Impact result	4.9		
Base result	4.9		
Vector	AV: [N]/AC:[M]/Au:[S]/C:[P]/I:[P]/A:[N]/EM:[A]		

In the first experiment, the exploitability factor was assessed at 6.9 points, indicating that exploiting this vulnerability is not difficult. This initial impression raises concerns about its potential danger. Although the impact factor was evaluated at 4.9, it is important to note that the actual impact may vary depending on the privileges assigned to dynamic groups. If higher privileges are granted, or if the scope of user access is narrower, the impact could be more significant. Conversely, lower privileges, or a more restricted user scope, could result in a lesser impact. It is crucial to recognize that the impact is directly correlated with the confidentiality, integrity, and availability of information and services. Taking all factors into account, the vulnerability was assessed with a Base Factor of 4.9 out of 10, placing it in the category of medium-risk vulnerabilities.

6.2. Evaluation of the Second Experiment

Using the CCSS framework, initial results will be presented regarding configuration vulnerabilities. Table 3 contains all the factors and an explanation of why each value was chosen. Additionally, the results of the calculation of Exploitability, Impact, and Base are shown.

Table 3. The second experiment’s evaluation values.

Factor	Value	Score	Explanation
Access Vector (AV)	Local Access (N)	0.395	Obtaining the token requires executing a command locally on the target device.
Access Complexity (AC)	Medium (M)	0.61	Special permissions are required to extract the token from the target device.
Authentication (Au)	Single (S)	0.56	A single login attempt is sufficient to exploit this vulnerability.
Confidentiality Impact (C)	Complete (C)	0.660	The hacker gained owner role access to the subscription scope because of the attack.
Integrity Impact (I)	Complete (C)	0.660	The hacker gained owner role access to the subscription scope because of the attack.
Availability Impact (A)	None (N)	0.0	No service was directly disrupted because of this attack.
Exploitation Method (EM)	Exploited Actively (A)	non	The hacker was able to access parts of the system after carrying out the attack.
Exploitability result	2.7		
Impact result	9.2		
Base result	6.0		
Vector	AV: [N]/AC:[M]/Au:[S]/C:[C]/I:[N]/A:[N]/EM:[A]		

Based on our experiment's results, the exploitability factor was rated at 2.9 points, suggesting that exploiting this vulnerability would be challenging, due to the requirement for specialized privileges and local execution of the exploit on the target device. In contrast, the impact factor scored 9.2 out of 10 points, indicating a critical level of impact. This high rating can be primarily attributed to the compromised integrity and confidentiality factors, as the attacker gained control over a wide range of subscribers, thereby accessing a significant amount of files and information. If the obtained virtual machine privileges are reduced or a specific scope is selected, the value may decrease to 2.7 out of 10. Finally, the base factor received a rating of 4.9 out of 10. The higher rating is primarily influenced by the elevated impact value compared to previous experiences. Therefore, it is highly recommended to consider all relevant factors, rather than solely focusing on the final value, when assessing vulnerabilities that arise from configuration errors. This holistic approach is crucial in obtaining a comprehensive understanding of the potential risks involved.

6.3. Discussion

This paper investigates the causes and risks associated with privilege escalation attacks on Microsoft Azure Active Directory due to configuration errors. A replicable laboratory environment was created using Microsoft Azure and Azure Active Directory, incorporating all security features to simulate the infrastructure of a software company. Realistic scenarios were implemented, including options to fulfill users' requirements, such as remote access to specific virtual machines. Through these simulations, it was observed that a hacker could elevate their privileges vertically and horizontally within the system, even without exploiting any software bugs. This experience highlights the significance of proper configuration, as a single misconfiguration can have substantial consequences for an organization, prompting a reevaluation of the reliance solely on the principle that SaaS is protected by the service provider.

Prior research and industry observations have revealed that the facilitation of user processes and the automation of tasks were primary motivations behind privilege escalation attacks. In the first scenario, the dynamic group feature was activated to expedite and automate the granting of access to new users. In the second experiment, the managed identity feature was employed to streamline authentication between the virtual machine and the required resources. Despite being classified as an additional layer of protection, inadequate preparation of these features rendered them vulnerable. System administrators were often unaware of the severity of these errors, as they occurred unintentionally. Their intention was not to introduce vulnerabilities to the system, but rather to enhance user efficiency by not specifying the scope of assigned roles. Consequently, system administrators must possess comprehensive knowledge of security aspects.

Azure Active Directory provides numerous configuration options, and their combinations can yield countless outcomes. As the system grows, the complexity of these options increases, leading to a higher likelihood of configuration errors and vulnerabilities. Considering that the present research adopts a hacker's perspective, the following question arises: Which factors contribute to the inclusion of misconfiguration vulnerabilities that enable privilege escalation attacks, and how does the assessment differ among organizations?

To address this question, the research employs the CCSS standard to assess configuration weaknesses and their associated risk implications. The assessment of weaknesses considers three fundamental values: exploitability, impact, and base. Exploitability determines the ease and conditions for exploiting a vulnerability, impact assesses the consequences for the organization, and base provides an overall evaluation of the vulnerability. The adoption of this standard aims to measure the threats associated with configuration errors, which often differ significantly from typical vulnerabilities. This discrepancy arises from the varied settings, privileges, services, and errors that may exist within an organization, as well as the extent of their impact. During the second experiment, the effect levels of integrity and confidentiality were set to "Complete" when the virtual machine's privileges were exploited at the subscriber level. If the device's permissions are confined to a specific

range, the effect becomes “Partial”, reducing the impact value from nine to two. While the privilege level of the virtual machine itself did not directly influence the successful execution of the privilege escalation attack, the privilege level of the user played a role. Therefore, while the vulnerability resulting from misconfigurations in the managed identity feature may exist across multiple organizations, its evaluation score may differ based on specific circumstances.

7. Conclusions

This paper investigates the causes and risks associated with privilege escalation attacks on Microsoft Azure Active Directory due to configuration errors. The experiment simulated two scenarios based on the employee requirements of a game programming company. In the first scenario, a system administrator’s error in dynamic group settings resulted in a loophole that allowed a hacker to gain access to all virtual machines and escalate their power. The attack can be prevented by avoiding dynamic group conditions that rely on user-controllable attributes and closely monitoring external users. In the second scenario, a loophole was created when the system administrator activated the Managed Identity feature on a virtual device, resulting in the hacker gaining access to sensitive information using VM privileges. To prevent such attacks, identity management should only be used when necessary, and permission to manage identity objects should be granted only to a specific scope.

Our findings contribute significant value by providing novel insights into the specific impacts and implications of misconfigurations in Microsoft Azure Active Directory within real-world scenarios. Furthermore, we identify areas for future exploration, including the assessment of privilege escalation in on-premises Microsoft Active Directory and the evaluation of the effectiveness of the Zero Trust concept. This paper enhances the existing knowledge base and offers valuable insights that contribute to ongoing efforts to enhance the security of Azure Active Directory, while also guiding future research endeavors in the field.

Author Contributions: Conceptualization, I.B.H. and A.A.; methodology, I.B.H.; software, I.B.H. and A.A.; validation, A.A. and M.A.; formal analysis, I.B.H.; investigation, A.A.; resources, I.B.H.; data curation, I.B.H.; writing—original draft preparation, I.B.H. and A.A.; writing—review and editing, M.A.; visualization, I.B.H. and A.A.; supervision, M.A.; project administration, M.A.; funding acquisition, M.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number: IFP22UQU4400257DSR033.

Data Availability Statement: Datasets used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ren, K.; Wang, C.; Wang, Q. Security Challenges for the Public Cloud. *IEEE Internet Comput.* **2012**, *16*, 69–73. [CrossRef]
2. Khalil, I.; Khreishah, A.; Azeem, M. Cloud computing security: A survey. *Computers* **2014**, *3*, 1–35. [CrossRef]
3. Shitta-Bey, A.M.; Adewole, M. Security Concerns of Cloud Migration and Its Implications on Cloud-Enabled Business Transformation. Ph.D. Thesis, Università della Svizzera Italiana, Lugano, Switzerland, 2023.
4. Logeswaran, L.; Bandara, H.M.N.D.; Bhathiya, H.S. Performance, Resource, and Cost Aware Resource Provisioning in the Cloud. In Proceedings of the 2016 IEEE 9th International Conference on Cloud Computing (CLOUD), San Francisco, CA, USA, 27 June–2 July 2016; pp. 913–916.
5. Rana, O. The Costs of Cloud Migration. *IEEE Cloud Comput.* **2014**, *1*, 62–65. [CrossRef]
6. Copeland, M.; Soh, J.; Puca, A.; Manning, M.; Gollob, D. Microsoft Azure: Planning, Deploying, and Managing Your Data Center in the Cloud: Build, Scale, and Strengthen Your Data Center with Microsoft Azure. Available online: <https://link.springer.com/book/10.1007/978-1-4842-1043-7> (accessed on 24 October 2022).

7. Mayank, M.; Garg, M. Introduction to Azure Active Directory. In *Developing Applications with Azure Active Directory: Principles of Authentication and Authorization for Architects and Developers*; Mayank, M., Garg, M., Eds.; Apress: Berkeley, CA, USA, 2019; pp. 1–16. [CrossRef]
8. Ramgovind, S.; Eloff, M.M.; Smith, E. The management of security in Cloud computing. In *2010 Information Security for South Africa*; IEEE: Piscataway, NJ, USA, 2010; pp. 1–7. [CrossRef]
9. Subashini, S.; Kavitha, V. A survey on security issues in service delivery models of cloud computing. *J. Netw. Comput. Appl.* **2011**, *34*, 1–11. [CrossRef]
10. Mitigating Cloud Vulnerabilities. NSA. Available online: https://media.defense.gov/2020/Jan/22/2002237484/-1/-1/0/CSI-MITIGATING-CLOUD-VULNERABILITIES_20200121.PDF (accessed on 3 September 2022).
11. Singh, V.; Pandey, S.K. Revisiting Cloud Security Threats: Elevation of Privilege. Social Science Research Network, Rochester, NY, SSRN Scholarly Paper 3331932. February 2019. Available online: <https://www.semanticscholar.org/paper/Revisiting-Cloud-Security-Threats%3A-Elevation-of-Singh-Pandey/94dc299dc0db9426707a0432662cf2cec85c30be> (accessed on 15 June 2022).
12. Takabi, H.; Joshi, J.B.; Ahn, G.-J. Security and Privacy Challenges in Cloud Computing Environments. *IEEE Secur. Priv.* **2010**, *8*, 24–31. [CrossRef]
13. Suryateja, P. Threats and Vulnerabilities of Cloud Computing: A Review. *Int. J. Comput. Sci. Eng.* **2018**, *6*, 297–302. [CrossRef]
14. Rath, A.; Spasic, B.; Boucart, N.; Thiran, P. Security Pattern for Cloud SaaS: From System and Data Security to Privacy Case Study in AWS and Azure. *Computers* **2019**, *8*, 34. [CrossRef]
15. Alexander, C. *A Pattern Language: Towns, Buildings, Construction*; Oxford University Press: Oxford, UK, 1977.
16. Zhang, X.; Wuwong, N.; Li, H.; Zhang, X. Information Security Risk Management Framework for the Cloud Computing Environments. In Proceedings of the 2010 10th IEEE International Conference on Computer and Information Technology, Bradford, UK, 29 June–1 July 2010; pp. 1328–1334. [CrossRef]
17. Mell, P.; Scarfone, K.; Romanosky, S. Common Vulnerability Scoring System. *IEEE Secur. Priv.* **2006**, *4*, 85–89. [CrossRef]
18. Mell, P.M.; Scarfone, K. The Common Configuration Scoring System (CCSS): Metrics for Software Security Configuration Vulnerabilities. NIST; 2010. Available online: <https://www.nist.gov/publications/common-configuration-scoring-system-ccss-metrics-software-security-configuration> (accessed on 12 November 2022).
19. TerryLanfeard. Penetration Testing. Available online: <https://docs.microsoft.com/en-us/azure/security/fundamentals/pen-testing> (accessed on 11 August 2022).
20. Azure Free Account FAQ | Microsoft Azure. Available online: <https://azure.microsoft.com/en-gb/free/free-account-faq/> (accessed on 17 August 2022).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.