Attribute Threat Analysis and Risk Assessment for ABAC and TBAC Systems

Leonard Bradatsch[a], Artur Hermann[b] and Frank Kargl[c]
Institute of Distributed Systems, Ulm University, Albert-Einstein-Allee 11, 89081 Ulm, Germany

Keywords: Access Control, Threat Analysis, Risk Assessment, Zero Trust Security.

Abstract: As enterprises increasingly adopt Zero Trust security, access control based on attributes is regaining attention as a core aspect of Zero Trust. Evaluating the accuracy of access decisions is a vital aspect of securing access control systems, typically involving threat analysis and risk assessment. A notable threat is attackers gaining illegitimate access by compromising the attributes checked by the access control policies. However, a systematic methodology for assessing attribute compromise risk is lacking. Knowing this risk aids in designing more accurate access control policies. This paper introduces a novel framework to address this gap, using modeled attackers and enterprises for risk assessment of attribute compromise. We also present a detailed case study featuring six attackers and two enterprises, demonstrating the framework's practicality and providing insights into the security strength of fifteen common access control attributes. In the context of the case study, attributes such as *Certificate Authentication*, along with *User Usage* and *Device Usage*, which both reflect the coupling of users and devices, demonstrated high resilience against compromise attempts.

1 INTRODUCTION

Organizations are increasingly adopting Zero Trust (ZT) security due to the limitations of traditional perimeter security against threats such as those from malicious insiders (Rose et al., 2020). ZT emphasizes stringent access control, encompassing both authentication and authorization, for every access request. ZT distinguishes two main access control types: criteria-based and score-based (Rose et al., 2020).

Criteria-based access control, similar to Attribute-Based Access Control (ABAC) (Hu et al., 2013), grants access only if an access request meets all predefined policy criteria. These criteria, defined by attributes such as *usual access time* and valid *device certificates*, must be satisfied by the client's request, representing both user and device. An example is shown in Listing 1.

Conversely, score-based access control, akin to Trust-Based Access Control (TBAC) (Xiaoning, 2012), assigns weights to attributes. Access is granted if the cumulative score from these weighted attributes meets or exceeds a defined trust threshold. An example TBAC policy using an additive score calculation method is in Listing 2.

```
if accessTime != usualAccessTime  {
    request.deny
}
if providedDeviceCert != correctDeviceCert {
    request.deny
}
request.permit
```

Listing 1: Example criteria-based access control policy.

```
score = 0
if accessTime == usualAccessTime {
    score += 1 # Example attribute weight
}
if providedDeviceCert == correctDeviceCert {
    score += 4 # Example attribute weight
}
if score >= threshold {
    request.permit
} else {
    request.deny
}
```

Listing 2: Example score-based access control policy.

It is crucial to assess the security strength of ABAC and TBAC systems to prevent illegitimate access, particularly by evaluating the accuracy of their access decisions. Conducting a Threat Analysis and

[a] https://orcid.org/0000-0001-7120-6557
[b] https://orcid.org/0009-0004-3406-267X
[c] https://orcid.org/0000-0003-3800-8369

Bradatsch, L., Hermann, A. and Kargl, F.
Attribute Threat Analysis and Risk Assessment for ABAC and TBAC Systems.
DOI: 10.5220/0012715300003767

In *Proceedings of the 21st International Conference on Security and Cryptography (SECRYPT 2024)*, pages 26-39
ISBN: 978-989-758-709-2; ISSN: 2184-7711

Risk Assessment (TARA) is a common approach to evaluate and enhance the security strength of such access control systems (Shostack, 2014). TARA is a systematic process that involves identifying, assessing, and prioritizing potential threats, evaluating the risks they pose to the target under evaluation, and developing strategies to mitigate them.

In a TARA for ABAC and TBAC systems, a particular focus is placed on the attributes used. Both ABAC and TBAC rely on these attributes to dictate the conditions under which clients are granted access. A significant threat is the potential compromise of these attributes, enabling attackers to mimic legitimate clients and gain illegitimate access. For example, in 2022, stolen credentials accounted for 49 percent of all data breaches (Verizon, 2023). The ease of compromising attributes varies. For instance, the *usual access time* is relatively easy to infer by just probing and thus more vulnerable. In contrast, a *device certificate* is significantly harder to compromise due to strong default security measures like storing the certificate's private key in a protected key storage.

Systematically assessing threats to attributes through a TARA can be utilized for determining the attributes' risk level. We define the risk level as the risk (low, medium, or high) with which an attribute can be compromised. A TARA involves analyzing potential attackers and the security strength of attribute implementations. However, there is a lack of research specifically focused on evaluating attribute compromise risks in ABAC and TBAC systems. Focused research in this topic could improve understanding of attributes' security strength, aiding in developing more accurate access control policies and preventing illegitimate access. In ABAC, such research could identify the most secure attributes for securing sensitive data, while in TBAC, it could guide the assignment of attribute weights based on their risk of compromise.

This paper addresses the research gap in conducting systematic threat analysis to assess attribute compromise risks. Our main contributions are as follows:

- We present two use cases motivating the risk level assessment of attributes within the context of ABAC and TBAC.
- We introduce a framework that allows the assessment of attribute risk levels depending on threats modeled by a TARA.
- We provide a detailed case study with six modeled attackers and two modeled enterprises demonstrating the framework's application. This case study provides concrete insights into the security strength of the 15 evaluated attributes.

The paper is organized as follows: Section 2 provides an overview of the necessary technical background and reviews related work. Section 3 outlines two use cases, motivating the application of our framework. In Section 4, we introduce our attribute assessment framework. Section 5 presents a detailed case study utilizing this framework. The use cases, case study, and broader implications are discussed in Section 6. Finally, Section 7 concludes with a summary and an outline of potential future work.

2 BACKGROUND & RELATED WORK

This section outlines TARA aspects pertinent to our work.

TARA outlines cybersecurity methods used to identify and evaluate threats and vulnerabilities within a system. It involves thoroughly examining the system's components and operations to pinpoint potential compromise points and assess the impact of such compromises.

Significant contributions to TARA methodologies include works by (Rocchetto and Tippenhauer, 2016), (Cardenas et al., 2009; Cárdenas and Baras, 2006), (Corman and Etue, 2012), and (Papakonstantinou et al., 2021). The STRIDE model by (Shostack, 2014) is one of the most established methods for identifying threats in a TARA. We will follow STRIDE in our paper.

The STRIDE model addresses the security threats of Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege. It starts with scrutinizing each access control system component under these threat categories to pinpoint vulnerabilities. This is followed by a risk assessment evaluating the likelihood and impact of threats, considering aspects like data sensitivity and ease of compromise.

In our work, we focus on the likelihood of threats and address the security threat of spoofing. Under the STRIDE model, spoofing refers to masquerading as a legitimate client to gain illegitimate access to a resource. This threat is particularly relevant in ABAC and TBAC systems, where access decisions are based on client attributes. Attackers may exploit these systems by compromising attributes to spoof legitimate clients, thereby gaining illegitimate access.

In ABAC and TBAC, various studies address spoofing threats. OWASP covers types of spoofing under the terms *Broken Access Control* (OWASP Foundation, 2021a) and *Identification and Authentication Failures* (OWASP Foundation, 2021b), espe-

cially in web security. They describe vulnerabilities that enable spoofing, such as weak credentials or access control bypassing. (Crampton et al., 2015) and (Morisset et al., 2018; Morisset et al., 2019) assess the impact of missing attributes in ABAC systems. Research by (Mandal et al., 2021), (Chattaraj et al., 2020), and (Manoj and Chandrasekar, 2014) focuses on detecting or preventing attribute compromise, such as location, in ABAC models. (Zhang et al., 2013) and (Esmaeeli and Shahriari, 2010) discuss secure attribute storage to prevent attribute compromise. Additionally, several works, such as from (Sasse et al., 2014), examine user mistakes or fatigue in access control leading to spoofing attacks. The identified related work extensively addresses specific spoofing threats and countermeasures through various approaches like evaluating vulnerabilities, integrating detection mechanisms, and elucidating secure attribute storage.

However, there is a lack of understanding about attributes' security strength in the sense of a risk of compromise. To the best of our knowledge, no comprehensive framework exists for evaluating attributes' compromise risk to assess their security strength. Such an assessment can aid in designing more accurate ABAC/TBAC systems, particularly in mitigating the risk of spoofing for illegitimate access.

3 USE CASES

This section outlines two use cases in the ZT domain, demonstrating how applying our framework for assessing attribute compromise risk can contribute to designing more accurate ABAC and TBAC policies. The first use case relates to ABAC, and the second to TBAC.

3.1 ABAC Use Case

In ZT environments, ABAC policies use specific attributes to define conditions for resource access, as depicted in Listing 1 (Garbis and Chapman, 2021). Studies by (Garbis and Chapman, 2021) or (Bradatsch et al., 2023) identify attributes like access time or authentication factors for these policies. The number of such attribute conditions varies with the sensitivity of the service; for instance, a calendar system may have fewer conditions (Listing 1), while an enterprise billing system might require more (Listing 3).

While (Garbis and Chapman, 2021), (Vanickis et al., 2018), and (Ghate et al., 2021) provide ABAC policy languages suitable for such scenarios, research on the security strength of individual attributes is still ongoing. For example, adding the *access time* attribute may not significantly enhance security, as it is relatively easy to compromise. Our framework aims to assess the risk level of attributes, providing insight into how much an attribute strengthens a policy against illegitimate access through attribute compromise.

```
if accessTime != usualAccessTime  {
    request.deny
}
if softwarePatchLevel != upToDate  {
    request.deny
}
if providedUserPassword != userPassword {
    request.deny
}
if providedDeviceCert != correctDeviceCert {
    request.deny
}
request.permit
```

Listing 3: ABAC policy for an example billing system.

3.2 TBAC Use Case

TBAC policies, like ABAC, utilize attributes to control access. However, TBAC systems calculate a trust score from weights assigned to these attributes. Access is only permitted if the calculated trust score meets or exceeds a predefined threshold. An example of policy using an additive approach is shown in Listing 2. Models such as by (Yao et al., 2020), (Bradatsch et al., 2023), and (Dimitrakos et al., 2020) elaborate further approaches for trust score calculation for TBAC. The accuracy of TBAC hinges on precise attribute weighting. For instance, in a TBAC-enabled corporate billing system with attributes like *access time*, *user password*, *software patch level*, and *device certificate*, imbalanced weighting can create access inaccuracies. If the TBAC system overvalues *access time* and *software patch level* for trust calculations, an attacker could exploit this by compromising these two attributes, achieving the necessary trust score without a valid device certificate or the correct user password. This misconfiguration in attribute weighting could result in illegitimate access. Current research, often using example weights, needs a systematic approach for the determination of the weights. Our framework addresses this by guiding the assignment of reasonable weights to attributes based on their compromise risk, arguing that attributes that are easier to compromise should be weighted lower.

4 FRAMEWORK

Motivated by the use cases described in the previous section, we introduce a novel framework designed to assess the risk of attributes being compromised by analyzing threats to these attributes manifested by attackers. This framework models attacker scenarios, each involving a modeled attacker aiming to compromise the attribute under evaluation within a modeled enterprise. The outcome of this framework is a risk level that indicates the likelihood of the attacker successfully compromising the attribute. The impact assessment of an attribute compromise is out of scope of this paper. We assume that the access control system of the modeled enterprises is secure and cannot be compromised or bypassed.

We briefly overview the modeling process for attacker scenarios in the next paragraph, followed by a detailed description including a comprehensive modeling example in the upcoming subsections. This modeling process must be carried out for each attribute to be evaluated.

The modeling process consists of four phases. Phase One focuses on assessing the general feasibility of compromising the evaluated attribute. This involves analyzing the attribute's inherent vulnerabilities and the potential skills an attacker might utilize for compromising the attribute. This assessment should be grounded in literature and expert insights. The primary goal of this phase is to ascertain the baseline security strength of the evaluated attribute, independent of the modeled enterprise or attacker in the specific attacker scenario.

Phase Two involves developing the considered enterprise model, defining the implementation strength of the evaluated attribute. This implementation strength must be assessed individually for each enterprise and should be aligned with the responsible security experts. The implementation strength is then integrated with the attribute's general feasibility level to refine the general feasibility from Phase One according to the modeled enterprise.

Phase Three focuses on modeling the attacker, who aims to compromise the evaluated attribute. This involves defining the attacker's skill set, including skills such as effort or offensive knowledge. These skill sets can either be customized for particular scenarios or derived from literature. The framework then integrates these attacker skills with the attribute's adjusted feasibility level from Phase Two. This allows the refinement of the feasibility level to the specific attacker in the scenario.

In Phase Four, a risk level is derived from the feasibility level, adjusted for implementation strength and the attacker skill set. This risk level indicates the extent to which the evaluated attribute implemented in the modeled enterprise is at risk of being compromised by the modeled attacker.

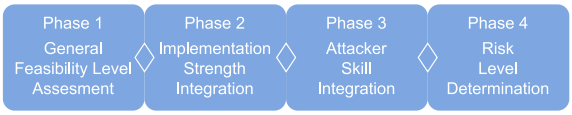

Figure 1: Conceptual depiction of the attacker scenario modeling process.

The conceptual framework of this process is depicted in Figure 1. In the following part of this section, the attacker scenario's modeling process is described in detail.

4.1 Phase One: General Feasibility Levels

The initial phase in attacker scenario modeling focuses on evaluating the general feasibility of an attacker compromising the evaluated attribute. This general feasibility is assessed for each attacker skill. Considering both the attacker's skills and the attribute's implementation strength as medium, this assessment aims to establish a baseline scenario. We denote each attacker skill's feasibility to compromise the evaluated attribute under these baseline conditions as $FL_{skill}^{attribute}$. These feasibility levels are categorized into high $>$ medium $>$ low.

This approach of establishing a baseline with medium levels for both attacker skills and implementation strength lays the groundwork for Phase Two and Three. In these subsequent phases, these general feasibility levels can be adjusted to reflect the unique characteristics and capabilities of each specific attacker and enterprise. Using the same baseline feasibility assessment allows multiple attacker and enterprise models to be effectively applied to the evaluated attribute.

For our framework, we use a set of attacker skills from (Rocchetto and Tippenhauer, 2016). They categorize these skills into three groups: (1) knowledge-based skills, (2) resource-related skills, and (3) psychological skills. The specific skills are defined as follows:

- (1) Knowledge-based skills:
  - *Offensive* describes the expertise of the attacker regarding performing attacks
  - *System* describes the attacker's knowledge about their target
- (2) Resource-related skills:

  - *Distance* refers to the attacker's proximity to their target, with higher skill levels indicating closer proximity, excluding blackmail or physical threats
  - *Manpower* specifies the amount of people involved in the attacks
  - *Tools* defines which tools are available to the attacker
  - *Financial Support* specifies the attacker's budget. We exclude the ability to bribe a target
  - *Effort* defines the willingness of the attacker to perform sophisticated attacks
- (3) Psychological skills:
  - *Determination* specifies how long an attacker is willing to performed an attack
  - *Periodicity* defines how often the attacker tries to attack
  - *Strategy* described how structured an attacker performs their attacks

This set of skills can be adjusted to fit specific use cases.

This general feasibility assessment should be grounded in literature and expert opinion. In our study (Section 5), we interviewed professional penetration testers to assess feasibility levels. For instance, focusing on *Password Authentication* (PwAuth) with *Offensive* skills, we found that passwords with medium implementation strength (minimum 8 characters, without mandatory use of password managers, salted and hashed passwords in a secure shadow file) are highly vulnerable to attackers with medium offensive capabilities. One major vulnerability arises from the absence of password managers, which typically provide domain verification, thereby making passwords significantly prone to phishing attacks. Consequently, we determined a high feasibility level ($FL^{PwAuth}_{Offensive} = high$) for breaching PwAuth with *Offensive* skills.

4.2 Phase Two: Enterprise Modeling

Phase Two of the modeling process focuses on modeling the specific enterprise in the attacker scenario. While Phase One assumes medium levels for attribute implementation strengths, actual implementation strengths vary among enterprises. Consequently, Phase Two requires assessing the specific attribute implementation strength within the enterprise to be modeled. These implementation strengths are classified into three categories: high $>$ medium $>$ low.

For instance, low *Password Authentication* strength indicates weak password policies and clear text storage on servers. Medium strength suggests improved policies and secure password hash storage. High strength involves stringent policies, mandatory password manager use, and highly secure server-side hash storage.

The implementation strength is denoted as $IS^{enterprise}_{attribute}$ and should be assessed by security experts responsible for the enterprise being modeled. An attribute's feasibility level is inversely related to its implementation strength: a high implementation strength makes compromising the attribute more challenging, lowering its feasibility level, and vice versa.

To reflect the diverse implementation strengths of attributes across different enterprises, we adjust the general feasibility levels $FL^{attribute}_{skill}$ from Phase One, using the enterprise's specific attribute implementation strength $IS^{enterprise}_{attribute}$. Therefore, for each skill, we apply the **First Integration Logic** to $FL^{attribute}_{skill}$. The logic is as follows:

- $FL^{attribute}_{skill}$ is decreased to the next lower level if $IS^{enterprise}_{attribute} = high$, with *low* as the minimum level
- $FL^{attribute}_{skill}$ is increased to the next higher level if $IS^{enterprise}_{attribute} = low$, with *high* as the maximum level
- $FL^{attribute}_{skill}$ stays unchanged if $IS^{enterprise}_{skill} = medium$.

For example, if the modeled enterprise has a high implementation strength for passwords ($IS^{enterprise}_{PwAuth} = high$), the feasibility level $FL^{PwAuth}_{Offensive} = high$ is reduced from high to medium.

4.3 Phase Three: Attacker Modeling

The third step in the modeling process for attacker scenarios involves the creation of attacker models, each possessing a specific set of skills and skill levels. We advocate for modeling concrete attackers rather than testing all possible combinations of attacker skills and levels. The latter approach lacks the capability to clearly recognize which attacker is a realistic threat and which is not. Our method allows for modeling concrete attackers tailored to each specific TARA, representing realistic threats.

The general feasibility levels $FL^{attribute}_{skill}$ determined in Phase One were based on the assumption of medium attacker skill levels. However, different types of attackers may possess varying skill levels, which we categorize as high $>$ medium $>$ low. For example, an attacker with low offensive skills may conduct basic attacks, medium skills enable advanced attacks, and high skills facilitate highly sophisticated attacks.

When modeling attacker scenarios, it is crucial to consider the actual skill levels of the attackers. There-

fore, each considered skill must be assigned an attacker skill level denoted as $ASL_{skill}^{attacker}$ for the attacker to be modeled. This assignment should be grounded in a thorough review of relevant literature and informed by the expertise of security experts. A high skill level indicates greater feasibility for the attacker to compromise a specific attribute, whereas a low skill level suggests the opposite.

To account for varying attacker skill levels, the adjusted feasibility levels $FL_{skill}^{attribute}$ from Phase Two are modified according to the corresponding attacker skill levels ($ASL_{skill}^{attacker}$) of the attacker under consideration. To achieve this, we apply the **Second Integration Logic** to each skill considered for the attribute under evaluation. The specific logic is defined as follows:

- $FL_{skill}^{attribute}$ is increased to the next higher level if $ASL_{skill}^{attacker} = high$, with *high* as the maximum level
- $FL_{skill}^{attribute}$ is decreased to the next lower level if $ASL_{skill}^{attacker} = low$, with *low* as the minimum level
- $FL_{skill}^{attribute}$ stays unchanged if $ASL_{skill}^{attacker} = medium$.

The feasibility level is adjusted by only one level, regardless of whether the feasibility level is, for example, initially low and the attacker's skill level is high. This approach is due to the baseline used in the framework, where the *medium* attacker skill level was assumed for setting the general feasibility levels. As a result, both *high* and *low* attacker skill levels are considered to deviate by only one level from this *medium* baseline during the feasibility assessment. As an example, consider an attacker with a *low* offensive skill level ($ASL_{offensive}^{attacker} = low$). For the attribute *Password Authentication*, the feasibility level for this skill associated to the attribute ($FL_{Offensive}^{PwAuth}$) would be downgraded from high to medium.

4.4 Phase Four: Risk Level Determination

The final phase in the modeling process is the determination of the risk level for the evaluated attribute with which it can be compromised for an attacker-enterprise combination. For our framework, we use three different risk levels: (1) an attribute can be at high risk, (2) at medium risk, or (3) at low risk to get compromised. The following **Risk Level Determination Rules** are applied to each attacker scenario to determine the risk level, where an attribute is considered at **high risk** for compromise by an attacker against an enterprise if one of the following conditions is true:

- there is at least one $FL_{skill}^{attribute} = high$ in both the categories *Knowledge* **and** *Resources*
- there is at least one $FL_{skill}^{attribute} = high$ in the categories *Knowledge* or *Resources* **and** at least one $FL_{skill}^{attribute} = medium$ in the respective other category **and** at least one additional $FL_{skill}^{attribute} = medium$ in any category
- there is at least one $FL_{skill}^{attribute} = medium$ in both the categories *Knowledge* **and** *Resources*, **and** at least additionally $FL_{skill}^{attribute} = medium$ in two arbitrary categories.

An attribute is considered at **medium risk** for compromise if one of the following conditions is true:

- there is exactly one $FL_{skill}^{attribute} = high$ in the category *Knowledge* **and** exactly one $FL_{skill}^{attribute} = medium$ in *Resources* **and** all other feasibility levels are low
- there is exactly one $FL_{skill}^{attribute} = high$ in the category *Resource* **and** exactly one $FL_{skill}^{attribute} = medium$ in *Knowledge* **and** all other feasibility levels are low
- there is both one $FL_{skill}^{attribute} = medium$ in the categories *Knowledge* **and** *Resources*, **and** one additional $FL_{skill}^{attribute} = medium$ in any category.

An attribute is considered at **low risk** for compromise in all other cases.

We require this specific distribution of an attribute's feasibility levels across various skill categories. We consider it unrealistic for an attacker to compromise an attribute by possessing sufficient skills in only one of the three categories. For instance, an attacker should not be able to compromise an attribute by having high skills solely in determination and periodicity. Knowledge about the system or general offensive skills and having the resources to execute attacks is always required. We further argue that an attacker can compensate a high-level skill with two medium-level skills. However, having at least a medium level in the *Knowledge* and *Resources* categories remains a prerequisite.

4.5 Modeling Example

Before we present the case study, a comprehensive example is given to provide more insights into the modeling process and the thoughts behind it. In this example, we use the attacker archetype *Hacktivist* trying to compromise the attribute *Certificate Authentication* (CertAuth) implemented by a *Big Enterprise*. The assessments made below are the result of thorough interviews with penetration testers. The complete example modeling process is presented in Figure 2.

In Phase One, we evaluate the general feasibility levels for the attribute *Certificate Authentication*. An attacker must acquire the certificate and its private key to compromise the attribute. Unlike passwords, certificates are resilient to common attacks like phishing, weak password policies, and brute-force attempts. Standard tools like OpenSSL ensure robust default settings for certificates' private keys, including proper file permissions, making extraction difficult. Therefore, most of the attacker skills (with medium level) show a low feasibility for compromising certificates implemented with medium strength. Only offensive knowledge and proper tools have medium feasibility for compromising.

In Phase Two, we assessed the implementation strength of *Certificate Authentication* in Big Enterprises. Focusing on Windows, the predominant enterprise OS, we examined its certificate security measures. Windows provides robust protection for certificates' private keys by default. This includes hardware-based key storage with strong authentication and encryption, comprehensive cryptographic APIs, and extended detection and response tools that safeguard key storage processes and block illegitimate access. These mechanisms result in a high implementation strength $IS_{CertAuth} = high$. Consequently, when integrating this with the general feasibility levels from Phase One, the feasibility for compromising *Certificate Authentication* is low for all attacker skills ($FL_{skill}^{CertAuth} = low$).

In Phase Three, we focused on the *Hacktivist* attacker archetype, following (Rocchetto and Tippenhauer, 2016). This archetype describes hackers with medium offensive knowledge and tools, working in small groups. Their standout trait is high motivation, leading to significant effort and determination. When we applied these characteristics to the implementation strength-adjusted feasibility levels from Phase Two, both effort and determination feasibility levels were adjusted from low to medium ($FL_{Effort}^{CertAuth} = medium$, $FL_{Determination}^{CertAuth} = medium$). All other feasibility levels remain low.

Consequently, following the rules for determining the risk level for an attribute from 4.4, the attribute *Certificate Authentication* is at low risk getting compromised by a *Hacktivist* if implemented by a *Big Enterprise*.

5 CASE STUDY

In the following section, we present a detailed case study that evaluates the feasibility and practicality of our framework. This study also attempts to offer insights into the security strength of common attributes used in ABAC or TBAC systems.

The study evaluates general feasibility levels for 15 attributes and ten attacker skills. It includes the construction of models for two separate enterprises. Additionally, the study adapts six unique attacker archetypes, each characterized by distinct skill sets. Drawing from these data, we derive the attribute risk levels from each attacker-enterprise combination.

The methodology of this modeling process is elaborated in the subsequent sections.

5.1 General Feasibility Levels

In the initial phase of our case study, we assessed the general feasibility level for each attribute-attacker skill combination.

We first aimed to identify a set of attributes for the risk level assessment. (Bradatsch et al., 2023) provide a comprehensive list of attributes relevant to access control. Focusing on attributes common in both educational and enterprise networks, we conducted structured interviews with two security experts and three IT managers from these sectors. This approach yielded a subset of attributes detailed in Table1 and categorized by user or device association as in (Bradatsch et al., 2023).

For the attacker skills, we included all the skills described in Subsection 4.1.

We evaluated the general feasibility levels ($FL_{skill}^{attribute}$) for each attribute-attacker skill pairing. This assessment was based on in-depth interviews with three professional penetration testers. Details of these interviews, including assumptions, methodology and proof of concept implementation, will be available on our GitHub[1]. The derived feasibility levels are presented in Table 1.

Our results highlight that the skills *Offensive*, *Tools*, *Effort*, *Determination*, and *Strategy* are each effective in compromising a wide range of attributes. This emphasizes the importance of an attacker's knowledge (*Offensive*), methodical approach (*Strategy*), tool availability (*Tools*), and time investment (*Effort* and *Determination*) in successfully breaching security measures. However, different attacker skills are of varying efficacy for specific attributes. For instance, *Offensive* knowledge is highly effective in compromising passwords due to well-known attack strategies like phishing, while its feasibility is only medium for device certificates due to better default protection mechanisms, which prevent several attack vectors existing for passwords. Additionally, *System* knowledge is particularly useful for attributes where

[1]https://bitbucket.org/attrtara/data/

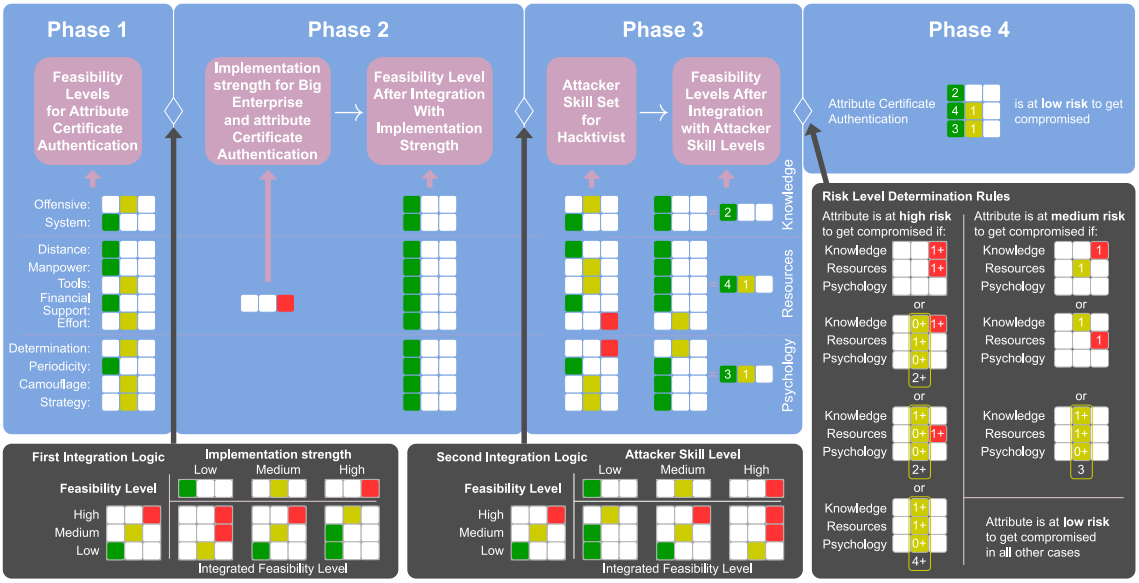

Figure 2: Example attacker scenario modeling process for the attribute *Certificate Authentication* implemented by a *Big Enterprise* and targeted by a *Hacktivist*.

internal understanding is crucial, such as the *Enterprise Presence* of colleagues. *Financial Support* was found to be less effective overall. Most attacks do not depend on expensive hardware and a high number of tools are open source. Attributes that reflect the time an attacker is willing to invest, such as *Effort* and *Determination*, are highly effective against attributes vulnerable to probing methods, such as *Access Time* and *Fingerprint*. Here, *Offensive* knowledge is of less importance as for probing no advanced knowledge is necessary, for example. The skills *Manpower* and *Periodicity* are less effective emphasizing that performing attacks appropriately is more important than performing attacks often. For example, performing just random attacks has almost no chance for compromising a device certificate. Moreover, attributes requiring up-to-date values, like *Software Patch Level* and *Connection Security*, were found vulnerable across various attacker skills. An attacker is able to compromise these attributes by using the latest versions of software. The reasoning behind the described results is accessible on GitHub[1]. These assessments are influenced by the personal experience of the interviewed penetration testers. However, we want to make an attempt to give a baseline assessment regarding the security strength of attributes.

The necessity for combining several attacker skills for being able to compromise an attribute is considered later in the modeling process.

5.2 Enterprise Models

For determining enterprise models, we conducted interviews with two security experts and three IT managers from small enterprises and universities (less than 500 employees) as well as from big enterprises (more than 500 employees). We surveyed them regarding which attributes are implemented in their enterprise and, if yes, with which implementation strength they rate the attributes. A high implementation strength indicates that the enterprise uses well-proven techniques and well-proven third-party software for the attribute protection. A medium level indicates that the enterprise implements its own solution for this attribute with the drawback that these solutions are not attacker-proof and are implemented mostly by non-experts. This is often the case for attributes such as *User Usage* or *Enterprise Presence*, which are not part of an existing software solution. The low level also indicates an enterprise's own solution but acknowledges that this attribute is easy to compromise, such as just checking for the latest patch for used software. If an enterprise type does not implement an attribute, we marked it as *not implemented*. The outcome of the interviews revealed that most of the surveyed bigger enterprises use existing software solutions for the authentication factors *Password Authentication* and *Certificate Authentication*, such as Microsoft Azure, whereas for other attributes, they prefer to implement their own solutions. Conversely, the surveyed smaller enterprises and also the universities only implement authentication factors

Table 1: Feasibility levels (high, medium, low) representing the likelihood of a successfully compromised attribute using the stated skill.

User Attributes / Attacker skills	Password Authentication	Enterprise Presence	Service Usage	Device Usage	Access Time	Access Rate
Offensive	high	medium	medium	low	medium	medium
System	medium	high	medium	low	medium	medium
Distance	low	medium	medium	medium	high	low
Manpower	low	medium	low	low	low	low
Tools	medium	medium	medium	low	high	medium
Financial Support	low	medium	low	low	low	low
Effort	medium	medium	medium	medium	high	medium
Determination	medium	medium	medium	medium	medium	medium
Periodicity	low	medium	medium	low	medium	low
Strategy	medium	medium	medium	medium	high	medium

Device Attributes	Cert Auth-entication	Conn-ection Security	System Patch Level	Software Patch Level	Finger-print	Enterprise Presence	Service Usage	User Usage	Type
Offensive	medium	high	high	high	medium	medium	medium	low	high
System	low	high	high	high	high	high	medium	medium	high
Distance	low	medium	high	high	low	medium	medium	medium	high
Manpower	low	high	high	high	low	medium	low	low	high
Tools	medium	high	high	high	medium	medium	medium	medium	high
Financial Support	low	low	high	high	low	medium	low	low	high
Effort	medium	high	high	high	high	medium	medium	medium	high
Determination	medium	high	high	high	high	medium	medium	medium	high
Periodicity	low	high	high	high	medium	medium	low	low	high
Strategy	medium	high	high	high	high	medium	medium	medium	high

Table 2: Implementation strength of the respective attribute in the respective environment such as a big enterprise.

User Attributes	Big Enterprise	Small Enterprise
Password Authentication	high	medium
Enterprise Presence	medium	not implemented
Service Usage	medium	not implemented
Device usage	medium	not implemented
Access Time	medium	not implemented
Access Rate	medium	not implemented

Device Attributes	Big Enterprise	Small Enterprise
Certificate Authentication	high	high
Connection Security	medium	not implemented
Software Patch Level	low	not implemented
System Patch Level	low	not implemented
Fingerprint	medium	medium
Enterprise Presence	medium	not implemented
Service Usage	medium	not implemented
User Usage	medium	not implemented
Type	low	not implemented

and check for changing device fingerprints. The interview results are outlined in Table 2.

5.3 Attacker Models

For the attacker models, we adopted the attacker archetypes from (Rocchetto and Tippenhauer, 2016), which are defined as follows:

- **Basic User** describes a low skilled attacker such as a script kiddie
- **Cybercriminal** is an advanced but lone attacker with professional tools available
- **Hacktivist** is similar to a cybercriminal but with higher manpower
- **Insider** is a lone attacker with high knowledge about the system who wants to damage the enterprise
- **Nation State** is an archetype from a highly skilled attacker(s) in almost all categories. Due to the fact that **Cybercriminal Groups** such as Fancy Bear have similar resources and capabilities, we include them in this category.
- **Terrorist** is a low skilled attacker regarding the knowledge-related skills but with high effort and determination

Accordingly, the attackers' skill levels are defined as depicted in Table 3.

5.4 Modeling Process

As the final step in our case study, we derived the risk levels for attribute compromise by applying Phase

Table 3: Attacker archetypes and their skill levels adopted from (Rocchetto and Tippenhauer, 2016).

	Basic User	Cybercriminal	Hacktivist	Insider	Nation State	Terrorist
Offensive	low	medium	medium	low	high	low
System	low	low	low	high	low	low
Distance	low	low	low	high	low	low
Manpower	low	low	medium	low	high	medium
Tools	low	high	medium	medium	high	medium
FS	low	medium	low	low	high	medium
Effort	low	medium	high	medium	high	high
Determination	low	medium	high	medium	high	high
Periodicity	low	medium	low	low	low	low

Two, Three and Four to the general attribute feasibility levels subsequently. If the enterprise does not implement a specific attribute, we skipped it for this attacker-enterprise pair. A comprehensive example based on the presented data was already presented in Subsection 4.5. The final results of the modeling process are presented in Table 4.

5.5 Results of Case Study

For **User Attributes** we observed the following things regarding their security strength: *Password Authentication* proofs as high risk attribute for small enterprises, facing high risks from all attacker types except *Basic User*. In contrast, the modeled *Big Enterprise* experience a generally low risk in this area, except for heightened threats from *Nation States* and medium risk from *Hacktivists*. This is mostly rooted in the fact that *Small Enterprises* have only medium implementation strength for *Password Authentication* and thus are vulnerable to common attack vectors such as phishing or guessing. Contrarily, for our modeled *Big Enterprise*, we assumed high implementation strength including mandatory password managers and strict password policies which prevent these attack vectors.

For attributes like *Enterprise Presence*, *Service Usage*, and *Access Time*, which are not implemented by *Small Enterprises*, *Big Enterprises* show a uniform high risk across all types of more sophisticated attackers, indicating that attributes that can be easily probed are prone to compromise. Consequently, it should not be placed too much confidence in these attributes for access control.

Device Usage and *Access Rate*, exclusively relevant to *Big Enterprises*, demonstrate varied risks. *Device Usage*, which couples devices to users, is mostly safe, only facing high risks from *Insiders* and *Nation States*. Spoofing the correct device for the target client proofs to be non trivial and being robust against most of the attackers. *Access Rate* exhibits a range from low to high risks depending on the attacker type, with *Nation States* and *Insiders* again posing significant threats. While *Access Rate* is also an attribute vulnerable to probing, it is significantly harder to find the correct access rate for a user-service pair, especially if it is implemented with a lower and upper limit.

Overall, the analysis reveals that *Nation States* and *Insiders* consistently emerge as the biggest threat. While the *Nation State* attacker archetype can bring every attribute to high risk of compromise, the *Insider* only fails facing a high implementation strength for *Password Authentication*.

For the security strength of **Device Attributes**, we noted the following observations: *Certificate Authentication* appears to be the least vulnerable attribute across most attacker archetypes for both big and small enterprises, consistently showing low risk. However, it becomes highly vulnerable to *Nation State* attackers, indicating that while it is robust against common threats, it could be compromised against highly sophisticated attacks.

Connection Security, *Software Patch Level*, *System Patch Level*, and *Type* all exhibit uniformly high risk across all attacker types for both big and small enterprises. These attributes require only to have the correct, for example, cipher suite or patch level, and are thus easy to compromise for even non-sophisticated attacker archetypes like *Basic User*.

Fingerprints present a medium risk for basic users in both big and small enterprises, but escalates to a high risk against all other attacker types. This is due to the fact that they are vulnerable to probing but harder to figure out the correct value compared to the previous mentioned attributes as they combine several pieces of information like browser resolution, browser version and operating system version.

Enterprise Presence and *Service Usage*, both showing low risk against *Basic Users* and *Terrorists* but high risk against the other attacker types; similar to the user attribute pendants.

User Usage, which couples users to devices, demonstrates low risk across most attacker types but becomes highly vulnerable to *Insiders* and *Nation States*. This is rooted in the same reason as for *Device Usage*, showing the difficulty to compromise not

Table 4: Risk levels (high, medium, low) correspond to attacker and enterprise models. Results in black represent risk levels for attributes in a large enterprise, while orange text shows those for a small enterprise. Attributes not implemented by small enterprises have no label.

User Attributes	Basic User	Terrorist	Hacktivist	Cybercriminal	Insider	Nation State
Password Authentication	low \ low	low \ high	medium \ high	low \ high	low \ high	high \ high
Enterprise Presence	low	high	high	high	high	high
Service Usage	low	low	high	high	high	high
Device usage	low	low	low	low	high	high
Access Time	low	low	high	high	high	high
Access Rate	low	low	medium	high	high	high
Device Attributes	**Basic User**	**Terrorist**	**Hacktivist**	**Cybercriminal**	**Insider**	**Nation State**
Certificate Authentication	low \ low	low \ low	low \ low	low \ low	low \ low	high \ high
Connection Security	high	high	high	high	high	high
Software Patch Level	high	high	high	high	high	high
System Patch Level	high	high	high	high	high	high
Fingerprint	medium \ medium	high \ high	high \ high	high \ high	high \ high	high \ high
Enterprise Presence	low	high	high	high	high	high
Service Usage	low	low	high	high	high	high
User Usage	low	low	low	low	high	high
Type	high	high	high	high	high	high

only the target user but also the correct device related to that user.

Overall, the analysis reveals that the attribute *Certificate Authentication* shows resilience against common attackers, they only falter against highly sophisticated attacks like those from *Nation States*. While *User Usage* shows also robustness against most of the attackers, the remaining attributes are at high risk to get compromised by more-sophisticated attackers such as *Hacktivists* or *Cybercriminals*.

6 DISCUSSION

In this section, we discuss key aspects of the framework, the case study, and the implications for the described use cases. Our discussion begins with an overview of the general aspects of our framework, followed by its limitations and the implications derived from our case study for both general attribute security and the specific use cases.

6.1 General Aspects

Our framework facilitates detailed risk level assessments for attributes in ABAC and TBAC systems, which is crucial for developing secure policies. Initially, it assesses attributes' general feasibility levels. Subsequent phases integrate specific attacker skills and enterprise implementation strengths, refining feasibility levels for different scenarios. In our case study, we employed six attacker types characterized by ten skills and two expert-driven enterprise models, leading to an extensive risk level assessment across various attributes. However, our framework's adaptability allows customization of attacker skills, attributes, and implementation strengths, ensuring flexibility and accuracy in diverse application scenarios. This flexibility is crucial in providing accurate and relevant risk assessments for each unique scenario.

In the modeling process of the framework, we utilize three distinct levels: *high*, *medium*, and *low*. This limitation to three levels was decided upon based on insights from expert interviews. We found that assessing the various aspects, such as feasibility levels, is already challenging with just these three distinct levels. Introducing more levels would likely lead to increasingly indistinct boundaries between levels.

6.2 Limitations

The application of our framework requires the determination of feasibility levels, attacker skill levels, and attribute implementation strengths. Due to the scarcity of reliable data on these factors, it is neces-

sary to consult security experts for these assessments. We initiated this with detailed interviews involving three penetration testers and three IT managers. However, this expertise is subject to the individual biases of each expert. Individual assessments might vary among experts. We recommend consulting multiple experts to obtain a more reliable assessment.

6.3 General Implications

In our case study, we aimed to realistically model attackers and enterprises to give insights on the security strength of commonly used attributes. The implications presented are based on the assumptions made for our case study.

Regarding user attributes in our *Big Enterprise* model, *Password Authentication* with high implementation strength is resilient against various attackers, except *Nation State* attacks. *Device Usage* is mainly susceptible to *Insider* and *Nation State* attacks, while *Access Rate* shows increased robustness but faces growing risks from advanced attackers like *Hacktivists*. Most user attributes are highly vulnerable to attackers at or above the *Hacktivist* level, though they are relatively secure against *Basic Users* and, to a lesser extent, *Terrorists*. In *Small Enterprises*, *Password Authentication* is generally vulnerable due to medium implementation strength.

To quantify these findings, we express the security strength of various user attributes in a big enterprise context in a relational order as follows:

$$\begin{aligned}&\text{Password Authentication} > \text{Device Usage} \gg \\ &\text{Access Rate} > \text{Service Usage} > \text{Others}\end{aligned} \tag{1}$$

Here, '$>$' signifies a higher security strength, and '$\gg$' indicates a substantially higher security strength based on the attribute's risk of being compromised derived from the conducted case study. For our small enterprise model, which primarily implement *Password Authentication*, we omit a similar relational order.

Regarding device attributes in both big and small enterprise models, *Certificate Authentication* is notably resistant to all attacker types except *Nation State*. Similarly, *User Usage* offers substantial security, except against *Insider* and *Nation State* attacks. *Service Usage* and *Enterprise Presence* primarily demonstrate spoof resistance to *Basic Users*. However, other device attributes generally show vulnerability to most attackers.

The relational order for device attributes in both large and small enterprise models is represented as:

$$\begin{aligned}&\text{Certificate Authentication} > \text{User Usage} \gg \\ &\text{Service Usage} > \text{Enterprise Presence} > \text{Others}\end{aligned} \tag{2}$$

6.4 Use Case Implications

The results from our case study has practical implications for the use cases discussed in Section 3. Specifically, when formulating ABAC access control policies, the most effective attributes are user passwords and device certificates, followed by *Device Usage* and *User Usage*. These attributes play a crucial role in securing sensitive information and preventing illegitimate access. Additionally, *Access Rate* and *Service Usage* can meaningfully enhance the accuracy of these policies.

In TBAC systems, our framework aids in determining appropriate attribute weights. The case study suggests assigning the highest weights to *Password Authentication* and *Certificate Authentication*, followed by *User Usage* and *Device Usage* with comparatively lower weights. There should be a significant difference in weights between these attributes and others, reflecting their security strength as shown in equations 1 and 2. Determining specific weight values requires aligning them with the system's threshold, the number of attributes considered, and the specifics of the TBAC system in question.

Determining specific ABAC policies and TBAC attribute weights is beyond this work's scope and is considered for future research.

7 CONCLUSION

This paper introduced a novel framework for assessing the risk levels of attributes used in ABAC and TBAC systems, evaluated as part of a TARA. This framework enables, for example, the evaluation of ABAC policies for their susceptibility to spoofing attacks and aids in determining attribute weights in TBAC systems. As part of a TARA, our framework can be integrated, for example, into STRIDE's spoofing threat evaluation.

Our framework models attackers by skill level, trying to compromise attributes in modeled enterprises characterized by implementation strength. This approach allows for an assessment of the risk level associated with each attribute being compromised. Attacker skills, attributes and their implementation strength can be adapted for the specific application scenario, making the framework highly customizable.

To demonstrate the framework's practicality, we conducted a comprehensive case study, evaluating 15 commonly used attributes in the domain of access control. These attributes were implemented in two enterprise models and assessed against six attacker models, with the entire study grounded in in-depth se-

curity expert interviews.

The results offer insights into the security strength of the evaluated attributes, helping with creating secure access policies. In terms of user attributes, *Password Authentication* with high implementation strength emerged as the most robust against compromise, followed by *Device Usage*. Among device attributes, *Certificate Authentication* exhibited the highest security strength, closely followed *User Usage*.

These findings enable a direct evaluation of the resistance of ABAC policies to spoofing attacks. Based on our case study results, policies incorporating the aforementioned four attributes are less vulnerable to such attacks. In TBAC systems, this information can guide the weighting of attributes, suggesting assigning the highest weights to these four attributes.

For future work, we aim to conduct more in-depth analyses and interviews with security experts to overcome the framework's current limitation of relying on single experts' opinions when assessing attribute feasibility levels. Our goal is to establish a general and unbiased baseline for the feasibility levels of attribute compromise, taking into account specific attacker skills. Additionally, we plan to expand the risk assessment to further attributes like passkeys.

REFERENCES

Bradatsch, L., Miroshkin, O., Trkulja, N., and Kargl, F. (2023). Zero trust score-based network-level access control in enterprise networks. http://arxiv.org/abs/2402.08299.

Cardenas, A., Amin, S., Sinopoli, B., Giani, A., Perrig, A., Sastry, S., et al. (2009). Challenges for securing cyber physical systems. In *Workshop on future directions in cyber-physical systems security*, number 1. Citeseer.

Cárdenas, A. A. and Baras, J. S. (2006). Evaluation of classifiers: Practical considerations for security applications. In *AAAI Workshop on Evaluation Methods for Machine Learning*, pages 409–415.

Chattaraj, D., Saha, S., Bera, B., and Das, A. K. (2020). On the design of blockchain-based access control scheme for software defined networks. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 237–242. IEEE.

Corman, J. and Etue, D. (2012). Adversary roi: Evaluating security from the threat actor's perspective. In *Proceedings of the RSA Conference Europe*.

Crampton, J., Morisset, C., and Zannone, N. (2015). On missing attributes in access control: Non-deterministic and probabilistic attribute retrieval. In *Proceedings of the 20th ACM Symposium on Access Control Models and Technologies*, pages 99–109.

Dimitrakos, T., Dilshener, T., Kravtsov, A., Marra, A. L., Martinelli, F., Rizos, A., Rosetti, A., and Saracino, A. (2020). Trust Aware Continuous Authorization for Zero Trust in Consumer Internet of Things. In *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1801–1812.

Esmaeeli, A. and Shahriari, H. R. (2010). Privacy protection of grid service requesters through distributed attribute based access control model. In *Advances in Grid and Pervasive Computing: 5th International Conference, GPC 2010, Hualien, Taiwan, May 10-13, 2010. Proceedings 5*, pages 573–582. Springer.

Garbis, J. and Chapman, J. W. (2021). *Zero Trust Security*. Springer.

Ghate, N., Mitani, S., Singh, T., and Ueda, H. (2021). Advanced zero trust architecture for automating fine-grained access control with generalized attribute relation extraction. *IEICE Proceedings Series*, 68(C1-5).

Hu, V. C., Ferraiolo, D., Kuhn, R., Friedman, A. R., Lang, A. J., Cogdell, M. M., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K., et al. (2013). Guide to attribute based access control (abac) definition and considerations (draft). *NIST special publication*, 800(162):1–54.

Mandal, S., Khan, D. A., and Jain, S. (2021). Cloud-based zero trust access control policy: an approach to support work-from-home driven by covid-19 pandemic. *New Generation Computing*, 39(3):599–622.

Manoj, R. J. and Chandrasekar, D. A. (2014). An enhanced trust authorization based web services access control model. *Journal of Theoretical and Applied Information Technology*, 64(2):522–530.

Morisset, C., Willemse, T. A., and Zannone, N. (2018). Efficient extended abac evaluation. In *Proceedings of the 23nd ACM on Symposium on Access Control Models and Technologies*, pages 149–160.

Morisset, C., Willemse, T. A., and Zannone, N. (2019). A framework for the extended evaluation of abac policies. *Cybersecurity*, 2(1):1–21.

OWASP Foundation (2021a). A01:2021-Broken Access Control. https://owasp.org/Top10/A01_2021-Broken_Access_Control/. Accessed: 2023-11-15.

OWASP Foundation (2021b). A07:2021-Identification and Authentication Failures. https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/. Accessed: 2023-11-15.

Papakonstantinou, N., Van Bossuyt, D. L., Linnosmaa, J., Hale, B., and O'Halloran, B. (2021). A zero trust hybrid security and safety risk analysis method. *Journal of Computing and Information Science in Engineering*, 21(5):050907.

Rocchetto, M. and Tippenhauer, N. O. (2016). On attacker models and profiles for cyber-physical systems. In *Computer Security–ESORICS 2016: 21st European Symposium on Research in Computer Security, Heraklion, Greece, September 26-30, 2016, Proceedings, Part II 21*, pages 427–449. Springer.

Rose, S., Borchert, O., Mitchell, S., and Connelly, S. (2020). Zero Trust Architecture. *NIST Computer Security Resource center*.

Sasse, M. A., Steves, M., Krol, K., and Chisnell, D. (2014). The great authentication fatigue–and how to overcome it. In *Cross-Cultural Design: 6th International Conference, CCD 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014. Proceedings 6*, pages 228–239. Springer.

Shostack, A. (2014). *Threat modeling: Designing for security*. John Wiley & Sons.

Vanickis, R., Jacob, P., Dehghanzadeh, S., and Lee, B. (2018). Access Control Policy Enforcement for Zero-Trust-Networking. In *2018 29th Irish Signals and Systems Conference (ISSC)*, pages 1–6. IEEE.

Verizon (2023). Data Breach Investigations Report. https://www.verizon.com/business/resources/reports/dbir/. Accessed: 2023-11-15.

Xiaoning, M. (2012). Formal description of trust-based access control. *Physics Procedia*, 33:555–560.

Yao, Q., Wang, Q., Zhang, X., and Fei, J. (2020). Dynamic access control and authorization system based on zero-trust architecture. In *2020 International Conference on Control, Robotics and Intelligent System*, pages 123–127.

Zhang, G., Liu, J., and Liu, J. (2013). Protecting sensitive attributes in attribute based access control. In *Service-Oriented Computing-ICSOC 2012 Workshops: ICSOC 2012, International Workshops ASC, DISA, PAASC, SCEB, SeMaPS, WESOA, and Satellite Events, Shanghai, China, November 12-15, 2012, Revised Selected Papers 10*, pages 294–305. Springer.