universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„Incident response configuration based on security policies“

verfasst von / submitted by

Felix Fally, BSc (WU)

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2021 / Vienna 2021

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066 926
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Wirtschaftsinformatik UG2002
Betreut von / Supervisor:	Univ.-Prof. Dipl.-Ing. Dr. Dr.Gerald Quirchmayr

Thesis Affirmation

I hereby affirm that this master's thesis was written by me without the unauthorised assistance of a third party. The resources used are clearly indicated and quoted in the references. Furthermore, I confirm that this thesis has not been submitted in any form to an examination board.

Vienna, May 28, 2021

Acknowledgments

I would like to take this opportunity to thank the people without whom this thesis would not have been possible. First and foremost, I would like to thank my supervisor, Prof. Dipl.-Ing. Dr. Dr. Gerald Quirchmayr, who has always given me advice and support in numerous meetings and guided me through the entire project. I would also like to thank my fellow students who motivated me and shared their experiences with me. Finally, I would like to thank my family and my girlfriend who have been with me through this whole journey and have always given me support and love.

Abstract

English The number of cyber attacks is growing steadily and hence, also the demand for countermeasures. [21] [7] This master's thesis discusses to what extent an incident response configuration can help the user to determine why an incident has resulted in a successful attack while focusing on the question whether an insufficient coverage of the policy or a violation of policy is the relevant cause. At first, a literature best practice review is performed through gathering information from related scientific areas. In the second step, a process model is developed, prototypically implemented and evaluated using a scenario-based approach. The result is an easily accessible, scalable, device-independent prototype that can lead to a targeted diagnosis of the cause of an incident at security policy level.

Keywords: Security Policy, Incident Response Model, Prototype

Deutsch Die Zahl der Cyber-Angriffe wächst stetig und damit auch der Bedarf an Gegenmaßnahmen. [21] [7] In dieser Masterarbeit wird diskutiert, inwieweit eine Incident-Response Konfiguration dabei helfen kann, die Ursachen für einen erfolgreichen Angriff zu ermitteln, wobei der Fokus darauf liegt, ob eine unzureichende Abdeckung der Security Policy oder eine Verletzung der Policy die relevante Ursache ist. Zunächst wird ein Literaturrecherche durchgeführt, indem Informationen aus verwandten wissenschaftlichen Bereichen und der Praxis gesammelt werden. Im zweiten Schritt wird ein Prozessmodell entwickelt, prototypisch implementiert und anhand einer Fallstudie evaluiert. Das Ergebnis ist ein leicht zugänglicher, skalierbarer und geräteunabhängiger Prototyp, der zu einer gezielten Diagnose der Ursache eines Vorfalls auf Ebene der Security Policy führt.

Table of Contents

1 Introduction

The first chapter provides motivation for and background to the topic. In the next step, the goals of this master's thesis are defined. Finally, an overview of the structure is given and briefly explained.

1.1 Motivation & Background

Incident response configuration based on security policies is a core aspect of developing effective responses to cyber attacks. As amply described in the literature, it is essential to align cyber incident handling with security policies. Such an approach will allow the embedding of incident handling in an enterprise in security management in general, thereby assuring that technical and organizational responses are aligned with the enterprises' overall security strategy. The motivation for this work originates from the ever-increasing threat of cyber attacks. A cyber attack is defined as a malicious attempt by one person or organisation to breach the information system of another person or organisation. [11] In contrast, cyber security is a means of protecting assets such as computer servers, mobile devices, electronic systems, networks and, above all, information in form of data. Since information as a digital commodity has become an increasingly important business asset over the past decades [23], it is important to understand the risks of cyber attacks. The background of such attacks is usually financially motivated, with the intention of stealing data and selling it in the dark market. The top five motives include financial, espionage, disruption, political and retaliation. The ENISA (European Union Agency for Cybersecurity) report shows that between January 2019 and April 2020, more than 620 million account details were stolen from 16 hacked websites and offered for sale on the dark web marketplace Dream Market. In January 2020, more than 770 million email addresses and 21 million passwords were posted on a popular hacking forum hosted by the cloud service Mega. This is referred to as the most significant collection of breached personal credentials, called "Collection#1".[5] As new threats emerge [21] and known threats are still effective due to their continuous improvement, the demand for countermeasures is high. [7] The supply for countermeasures is similarly high, ranging

from classic preventive controls, such as restricting access to network, programmes and data, to detection controls aimed at identifying threats, and to corrective controls, such as compensation insurances for damage caused by cyber attacks. [6] Nevertheless, these attacks cannot be completely prevented or the resulting damage completely eliminated despite the existing multitude of countermeasures. Therefore, researchers and practitioners in the field of information system security are concerned with developing approaches to combat these threats by analysing risks, modelling information system security, developing information security strategies and policies, and establishing international security standards. [9] The aim of this work is to develop an approach to support the further development of countermeasures. Since the first principle for securing information technology systems is the establishment of a solid security policy [44], this master's thesis follows a literature best practices review and outlines a process model that reviews a security incident on policy level focused on a specific attack technique. Based on this, it is investigated if the incident occurred due to non-implemented measures or if the implemented measures were violated.

1.2 Goals

The goal of this thesis is to find an answer to a practical need represented by the following research questions:
Q1: How can a process model be constructed and prototypically implemented that helps the user in determining why an incident has resulted in a successful attack? The focus is on analyzing whether a violation of policy or an insufficient coverage of policy is the relevant cause.
Q2: What can be achieved with the developed model and what are the limitations? To answer this research question, best practices are presented in the first step, on which the model development can then be based on and prototypically implemented. To demonstrate the viability of the model, a prototype is tested in a scenario-based approach.

1.3 Structure

This master's thesis is structured as follows: First, **State of the Art in Literature and Practice** , all aspects of the underlying subject area are put into context and scientifically based. In the **Research Approach** chapter, the research approach is outlined and the methodology justified. In **Model Development**, the solution approach is explained in theory and visualized by means of a use case diagram, a process diagram and a component diagram. The technical and logical requirements for the prototype implementation chapter are derived from this section. **Prototype Implementation** discusses the technological aspects according to which the prototype was implemented. The section **Scenario-based Testing of the Prototype** evaluates the developed prototype by means of a scenario-based approach in which three different scenarios are simulated in order to show what can be achieved with the model and what conclusions can be drawn from it. Last but not least in **Conclusion and Future Work**, the most important outcomes are once again summarized and an outlook on possible future work is given.

2 State of the Art in Literature and Practice

The literature review is intended to provide an overview of the state of the art in literature and practice in the field. Within this chapter, an overview of best practices will be given, which was explored based on the problem defined in the introduction and expanded during the iterative development of the process model. Due to, this chapter is divided into *Incident Handling*, *Process Models of Incident Handling*, *The MITRE ATT&CK Framework* and *The IT baseline protection*. In the course of *Incident Handling*, the term is defined and its relevance underlined, and then an already established, continuous procedure for dealing with incidents is presented in *Process Models of Incident Handling*. In the *MITRE ATT&CK Framework*, a knowledge base is presented which contains information about attack techniques and can be accessed via an interface. In the last section, *The IT Baseline Protection*, a methodology to identify and implement security measures for an information security concept are presented. The knowledge obtained in this chapter is then applied to the development of a model in the next chapter.

2.1 Incident Handling

This chapter gives an overview of how Incident Handling is defined. At the beginning, it is important to provide a definition of the word "Incident" and as well as related terms in order to be capable of putting them into connection with this thesis afterwards.

2.1.1 Incident Definition The term *incident* is correlated with the word *event*. An event can be any observable occurrence in a system or network. [17] Examples are:

- a user connecting to a Virtual Private Network
- a server receives a request for a web page
- a user receiving an email
- a spam filter alert
- a firewall blocking a connection attempt.

On the other hand *Adverse events* are *events* with a negative consequence, such as

- a break into the system
- the occurrence of malware
- network packet floods
- the detection of vulnerabilities with corresponding risk potential
- unauthorized use of system privileges
- defacement of a Web page
- execution of malicious code that destroys data [17].

When dealing with IT security incidents, overlaps with incidents in other areas must also be taken into account. [47] However, these will not be discussed in the course of this work. The use of the term *incident* within this paper, addresses solely *adverse events*, which are computer security-related and exclude adverse events raised by natural disasters and power failures. In this context, following NIST, an incident is considered a security related adverse event that results in a loss of data confidentiality, a disruption of data or system integrity, or disruption or denial of availability. [34] NASA defines incidents similarly in their incident response and management handbook: "An Information Security incident is an adverse event or situation associated with electronic and non-electronic information that poses a

threat to the integrity, availability, or confidentiality of that system." [27] Incidents are defined more generally, but with the same underlying statement, in the "Österreichisches Sicherheitshandbuch". According to the "Österreichisches Sicherheitshandbuch" an incident is defined as follows: "A security incident is an information security event that has a concrete impact on information security and subsequently causes or can cause major damage. Typical consequences of security incidents can be the spying, manipulation or destruction of data" [47]. NIST adds to the definition that in addition to the violation, the imminent threat of a violation of computer security policies, acceptable use policies or standard security practices are also considered an incident [17]. In the course of an incident, various techniques of an attacker can then be applied. Popular examples of this can be the following:

- Denial of Service - explicitly crafted packages are sent to a web server with the intention to crash it.
- Malicious Code - A so called "worm" takes usage of open file shares to quickly infect several hundred workstations within an organization.
- Unauthorized Access - An exploit tool is used to access a server's password file.
- Inappropriate Usage - Illegal copies of software are provided by a user through peer-to-peer file sharing services. [17]
- Phishing - Adversaries may send phishing messages to gain access to victim systems.
- Persistence - The adversary uses techniques to keep access to systems across restarts, changed credentials, and other interruptions that could cut off their access. [3] [27]

The above-mentioned threats, among others, give rise to the need for a deescalating handling of incidents.

2.1.2 Need for Incident Handling According to Forbes [45], 4.1 billion records were exposed within the first six months of 2019, which is a 52% rise from the same period in 2018. In July of 2019 Capital One, an American bank holding company suffered from what they refer to as "security incident", unauthorized access to 106 million credit cards and accounts. Due to that cyber threat, 140.000 so-

cial security numbers, 1 million social insurance numbers and 80.000 bank accounts have been breached [30]. Many security incidents only become a major problem due to incorrect or unplanned reactions, when decisions are made hastily or the wrong measures are taken in a stressful situation. Detailed guidelines as well as target group-oriented guidelines for the handling of security incidents which are made available to the respective employees enable all those involved to behave correctly in exceptional situations. [47] As incidents handling means responding fast and efficiently to address such incidents systematically on a daily basis, there is a need for building an incident response capability. Benefits in this manner are systematic response by taking appropriate steps, quick recovering from security incidents by keeping damages as low as possible, using knowledge gained from those security incidents to prepare for future handling of incidents and addressing legal issues, which may arrive during incidents. To counter these threats, the concept of responding to computer security incidents has been widely accepted and implemented in the federal government, the private sector and academia [17]. For dealing with incidents, a process model has emerged which is widely recommended and therefore applied in different organisations and based on the same foundation. [27] [14] [47]

2.2 Process Models of Incident Handling

So even if effective security measures and a high level of security are in place, the occurrence of such events cannot be completely prevented. Every institution must have a vital interest in reacting to security-relevant events as quickly and effectively as possible and in logging information about such incidents to prevent future damage. [47] In order to satisfy this demand, a process model for handling incidents has been developed, which represents the state of the art. This section relies on the Incident Handling Process of the National Aeronautics and Space Administration (short NASA), representative for an organization with a high need for a reliable process model due to critical infrastructure. The Incident Management Lifecycle, as shown in figure 1, is composed of serial phases (Preparation,Identification, Containment, Eradication, Recovery, and Follow-Up) and of ongoing parallel activities (Analysis, Communication,

and Documentation). The incident response process consists of several phases, from initial preparation to post-incident analysis. In the initial phase, an incident response team is assembled and trained and the necessary tools and resources are procured. During the preparation phase, the organization also tries to limit the number of incidents by selecting and implementing a number of controls based on the results of risk assessments. Detecting security breaches is therefore necessary to alert the organization when incidents occur. Depending on the severity of the incident, the organization can act to mitigate the impact of the incident by containing it and ultimately recovering from it. After the incident has been properly managed, the organization issues a report detailing the cause and cost of the incident and the steps the organization should take to prevent future incidents [17]. Each phase of the incident response process and the ongoing parallel activities are described in detail in the following subsections.

2.2.1 Preparation In this initial phase, the security incident management process itself has not been activated as organizations are in a state of preparation. The Incident Response Capability plans how to handle incidents and tries to limit the number of potential incidents by selecting and implementing a set of controls based on the results of risk assessments. In this step, the responsibilities of all parties involved, hardware, tools, documentation, etc. are outlined and steps are taken to reduce the possibility of an incident. The purpose of this phase is to prepare as well as prevent incidents by means of securing IT resources. [17] This phase is characterized by several activities [27]:

- Set up, organise and maintain incident response teams
- Procurement and maintenance of the necessary tools and resources for the handling of incidents
- Familiarise the incident response team with the environment through exercises such as frequent log reviews to better identify unexplained entries
- Synchronisation of all clients, servers and other device clocks to strengthen event correlation
- Training users to recognise and report IT security incidents.

Fig. 1. NASA Incident Management Lifecycle [27]

2.2.2 Identification The report of an Information Security Incident means that the preparation phase is stopped and the Incident Management process begins. [17] When a security relevant incident is suspected or confirmed, the report about this incident is determined either by calling NASA Security Operations Center (SOC) or by opening a ticket with NASA SOC Incident Management System (IMS). An Information Security Incident Report contains the following information and is told if available at the time of the report [27]:

– Name of the Submitter

- Phone Number of the Submitter
- Email Address of the Submitter
- IP Address of the Victim
- Domain Name of the Victim
- OS / Service Pack of the Victim
- Room Number and Building of the Victim
- System Security Plan number of the Victim
- Sensitivity and Description of Information Residing on Victim System
- IP Address of the Attacker
- Country and City of the Attacker
- Hostname / Domain Name of the Attacker
- Incident Summary
- Date / Time Incident Occurred
- Date / Time Incident Discovered
- Exploit Use or IDS/Anti-Virus Alter
- Incident Category
- Labor Hours / Cost of Downtime
- User's Role
- Systems or Networks the User Connected to Lately
- Symptoms
- User Activity When Problem First Noticed
- User Location When Incident Occurred
- Detailed Description of Incident

As soon as an incident is reported, the identification of it starts immediately. First, it must be determined whether or not an incident has occurred and if so, its type is determined based on the categorization of the incident and the prioritization of the response to it. The Incident Response Manager should appoint someone to be responsible for dealing with the incident and then conduct an analysis to determine whether it has actually occurred and understand its scope. When the Information Security incident has been confirmed it has to be categorized according to the Categories in Figure 2, which contains types of incidents that may threaten IT infrastructure. [27]

When multiple incidents occur in parallel, the potentially most serious one should be handled first, with the Incident Response Manager (IRM) determining the priority of the incident based on knowledge about it. By default, all incidents are handled with medium

Name	Description
Unauthorized Access	When an individual or entity gains logical or physical access without permission to a federal agency network, system, application, data, or other resource.
Denial of Service (DoS)	An attack that successfully prevents or impairs the normal authorized functionality of networks, systems or applications by exhausting resources. This activity includes being the victim or participating in the DoS.
Malicious Code	Successful installation of malicious software (e.g., virus, worm, Trojan horse, or other code-based malicious entity) that infects an operating system or application. Agencies are NOT required to report malicious logic that has been successfully quarantined by antivirus (AV) software.
Improper (or Inappropriate) Usage	When a person violates acceptable computing policies.
Suspected PII Breach	If an incident involves personally identifiable information (PII) a breach is reportable by being merely **Suspected**. (Suspected PII incidents can be resolved by confirmation of a non-PII determination.)
Suspected Loss of Sensitive Information	When an incident involves a suspected loss of sensitive information (not PII) that occurred as a result of Unauthorized Access, Malicious Code, or Improper (or Inappropriate) Use, but the cause or extent of which is not known.

Fig. 2. Incident Categorizations Examples [27]

priority, whereby an incident can be classified as low priority or high priority based on the assessment by the IRM. [27]

2.2.3 Containment Containment is important before an incident overwhelms resources or increases damage. Its purpose is trying to contain the damage an incident could cause while allowing it to have as little impact on mission critical processes as possible. The containment phase of the Incident Management Lifecycle requires critical decision making. [17] Decisions could be:

- determining whether a system should be shut down
- disconnecting it from the network
- monitoring its activity
- disable functions as remote file transfer

Incident Containment consists of a short-term, planned action to eliminate access to compromised systems to limit the extent of current system damage and prevent additional damage from occurring. The specific step that should be followed depends on the type of incident (intrusion, virus, theft, etc.) and whether the incident is still in progress or discovered after the action. The scope and extent of the incident should be limited as soon as possible because this is a higher priority than to obtain evidence to identify or prosecute

the perpetrator. If an information security incident impacts mission-critical information or computing services, it is crucial to decide how to address the incident while minimizing the impact on mission-critical processes. In the case of a low risk event, the IRM can decide to quickly resolve the incident without shutting down the affected system. In the event of a high-risk incident involving a system containing sensitive information or applications, the IRM can cause the system to be shut down or at least temporarily isolated from the network. If there is a reasonable way to keep the system running without the risk of serious damage, disruption, or data compromise, or to identify an offender, the IRM may determine that operations can continue under closed monitoring. [27] Additionally, a potential problem with containment is that some attacks can cause additional damage despite being contained. Incident handlers should not assume that further damage to the host is excluded just because a host, for example, has been disconnected from the network. [17]

2.2.4 Eradication Once an incident has been contained, eradication may be needed to remove components of the incident. An example of this would be deleting malware and disabling breached user accounts, as well as identifying and minimising any vulnerabilities that were exploited. Meanwhile, it is important to identify all affected devices so that they can be remediated. [17] To be more precise, Eradication refers to the application of technical measures to the affected system to remove the causes and effects of an attack in such a way that the risk of reoccurrence of the cause of the intrusion can be eradicated, or at least reduced to a minimal or acceptable level. Once this mitigation of risks is complete and all data that could be useful for an analysis of the compromise has been collected, the eradication can be proceeded with. [27] The Guideline according to NASA [27] is as follows:

- **Review Incident Analysis:** The data collected and analyzed should be used to gain insights into the exploited vulnerability and meet the minimum requirements for eradication.
- **Perform a Vulnerability Analysis:** A vulnerability analysis tool is used to study exposed systems, services and applications connected to the affected systems, with special attention

for web server/services, databases or other complex architectures such as service-oriented architectures (SOA), mainframes and e-commerce systems.
- **Improve Security Controls on the affected System and other Systems:** Protection techniques shall be implemented in the environment where appropriate. These may include applying security patches,changing the system name or IP-address, securing and protecting boundary defense hardware and software etc.
- **Focus on Removing Malignant Artifacts:** IRT focuses on the eradication of malignant artifacts and can focus on the eradication of harmless artifacts if they pose a serious risk.
- **Thoroughly Remove Artifacts From all Media:** The Representative ensures that malicious artifacts are removed from all systems and media by using one or more audited commercial applications for eradication or manual surgical removal following an in-depth malware analysis that identifies the entire malware package or the affected host. [27]

2.2.5 Recovery Incident Recovery is defined as the restoration of a system to a state of normal functionality. [17] Recovery involves having administrators confirm that the systems are functioning properly. Recovery may include measures such as restoring systems from clean backups, rebuilding systems from scratch and replacing compromised files with clean versions or installing patches. Part of the recovery process is often a higher level of system logging or network monitoring. This results from the fact that after a resource has been successfully attacked, it is often attacked again, or other resources within the organisation are attacked in a similar way. [17] The procedures for resuming normal functionality in this section include a framework for recovery after an incident (see figure 3). Recovery begins when the cause of the incident has been eliminated or reduced to a level of risk deemed acceptable by the Information System Owner (ISO) and IRM. The required guideline for conducting Recovery according to NASA [27] is:

- **Document the Recovery Phase:** The documentation can help maintain the focus on the recovery process. All documentation concerning the incident will be included in the report for future review and reporting.

- **Decide the System Restoration Procedure:** Depending on the severity of the incident, the sensitivity of the affected system and the available backup systems, several recovery options may be available.
- **Validate Data Restored from Untrustworthy Sources:** When restoring files other than the operating system and application files, only the most trusted backup files are used. Recovered system data and user files are checked for altered data or other signs of corruption.
- **Validate the Restored System before Returning to Service:** Validation of the tested systems is performed by performing a series of development tests when the results of priority tests are available for comparison.
- **Get Authorization and Communicate with Users before Restoring Service:** Before connecting the recovered system to the network again to resume, the Information System Security Official (ISSO) should obtain the permission from the IRM and notify any organization that would be effected.
- **Conduct a review of the security controls:** The IRM verifies that the system is configured according to the current configuration management guidelines, that the logging, auditing and accounting programs are functional and also that all security tools are working properly.
- **Monitor the Restored System:** The system should be monitored to prevent additional intrusion or a recurrence of the incident. All knowledge gained from the analysis should be used to give insights into the attacker's techniques and to develop better surveillance techniques. Attempts to monitor may include: failed login attempts, attempts to access back doors, etc

In figure 3, both the required and the additional guidelines from the framework for recovery are brought together and the possible scenarios are shown on the basis of a process model. Additional measures are *Only perform "Rapid Restoration" when Mission-Critical*, *Replace the Affected System with a backup system when possible*, *restore the system offline whenever operations allow* and *Restore the operating system from Trusted Media whenever operations allow.* [27]

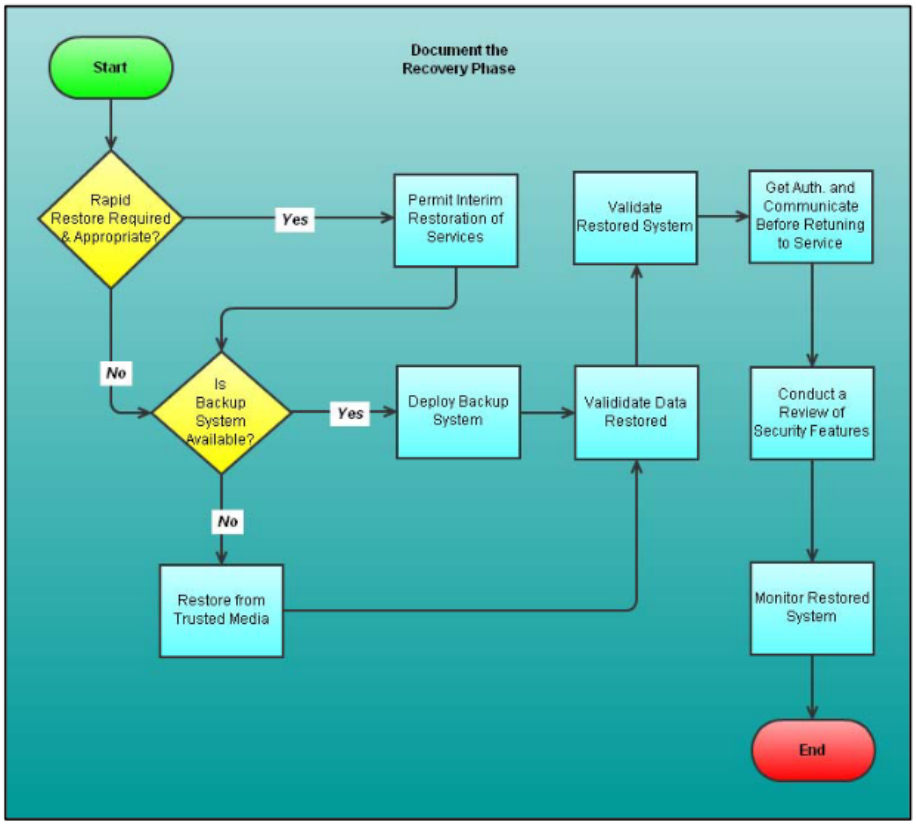


Fig. 3. Incident Recovery Framework [27]

2.2.6 Follow-Up Post-accident activities are an important stage in responding to an incident and post-accident activities provide the Incident Response Team (IRT) and other staff with an opportunity to learn and improve in incident handling [47]. Recording, documenting and sharing what worked and what did not helps to reduce the likelihood of similar incidents from happening again. The follow-up effort should reflect the scale and consequences of the incident. [27]

2.2.6.1 Post-Incident Analysis To accurately recall and record critical details, a post-incident analysis meeting has to be appointed as soon as possible. All parties involved must have a meeting to share the details of the incident, discuss the response procedures and de-

velop an estimate of the resulting costs. [17] The aim of the meeting is to address the weaknesses identified in the response and analysis of the system and to decide whether or not it is necessary to conduct a new risk analysis of the system. [27] [17]

2.2.6.2 Post-Incident Report The information gathered during the post-incident analysis and any other relevant information need be included in a post-incident report. This incident response report should include any lessons learned and cost analysis that can be used for further staff awareness and/or training. Finally, this report should be distributed to relevant staff, as, for example, the computer security officer (CSO) of an organisation affected by an incident. [27] [17]

2.2.6.3 Revising Policies, Procedures, and Security Plans Lessons learned from incidents can be integrated into computer security policies, procedures and plans, training and testing/exercises, and implemented in the resulting changes. [17] In addition, the "lessons learned" from each incident can be used to review computer security measures at agency, centre and programme level. On the basis of incident analysis and reporting, corrective actions can be developed and implemented to ensure that the incident does not recur or to effectively address a recurrence. [27] [47]

2.2.7 Documentation Documentation is intended as a parallel process since all phases of Incident Management require detailed documentation in order to be able to draw conclusions from the actions taken and the general course of events [13]. Furthermore, detailed documentation helps to determine the correct incident response and to provide important data for follow-up investigations and the preparation of the post-incident report. Documents regarding incidents should be protected and stored in the NASA SOC Incident Management System. Any information associated with an incident is supposed to be included in the documentation:

- Computer or non-IT data including logs, forensics images and other data that provides information about the threatened system
- All actions initiated by the response team
- Records of conversations with all personnel

– Any other relevant auditing information that may be gathered [27].

2.2.8 Analysis Analysis is, like documentation, an ongoing activity which may occur during or after activities throughout the Incident Management Lifecycle (see figure 1) Potential findings of an incident analysis results in an accurate eradication phase. To achieve that thorough documentation and team-internal communication to all team members involved is a fundamental prerequisite fo significant findings. [17] The purpose of collecting artifacts before, during and after an incident is to support the successful resolution of the incident and prevent its occurrence in the future. Furthermore, it is crucial to document how these were collected in order to be capable to use them as evidence in a legal process. Artifacts should be labeled, dated and signed if possible and stored in a secure location. Protection is essential to prove validity. To demonstrate integrity of electronic artifacts, cryptographic hash can be generated, logical artifacts should have access controls and should be stored on tamper-resistant media. Physical artifacts are supposed to be kept under lock. These measures serve to prepare evidence for a court case and to refute their doubtfulness. [27]

2.2.9 Communication The purpose of incident communication is an ongoing activity to ensure that all involved parties are informed, maintain situational awareness. [27] Information concerning the incident should be protected accordingly. [17] Compromised hosts or systems should not be used for incident handling discussions. If any communication tools (e.g: email, instant messengers, chat, etc.) are believed to be compromised, alternate systems should be used to prevent relevant information from being intercepted by the attacker. [27]

2.2.10 Incident Management Process Flow Once an incident has been reported, it is navigated within the so-called Incident Management Process Flow and its status changes according to the phases previously described in the Incident Management lifecycle. Initially, the Incident is given the status "New" and assigned to a "Technical

Investigator" who starts the Incident Response. Based on the results of the first analysis, the incident is classified. If the incident turns out to be a non-incident, for example, the Technical Investigator will arrange a disposition for the suspected incident. If it is not, the incident is given the status "confirmed". In this status, an analysis is made of the severity of both the affected resources and the incident. All affected members are then notified and containment, eradication and recovery are carried out. The incident then changes to the "resolved" status. In this step, the technical investigator must ensure that the incident is fully and accurately documented. Furthermore, the effectiveness of the incident response process should be assessed, all documentation reviewed, the disposition of the incident confirmed and improvements recommended. The Incident Response Manager then may conduct a Lessons Learned meeting with parties involved. After the completion of all necessary actions, the Incident is considered closed. [27]

Assigning an incident to a category requires a source of knowledge about the different categories of incidents. Therefore, in the next section, a knowledge base is presented which collects information about attack techniques and tactics, keeps them up-to-date, constantly expands it and provides access to it via an interface.

2.3 The MITRE ATT&CK Framework

MITRE ATT&CK, as a worldwide accessible knowledge base of adversary tactics and techniques based on real-world observations has gained popularity over the past years and has been integrated into popular threat information sharing technologies. [22] In the course of the model development and later on in the implementation of the prototype, advantage of the MITRE ATT&CK framework are taken to retrieve information on specific attack techniques. In the further course of this chapter the ATT&CK Knowledge Base are introduced.

2.3.1 Overview MITRE's ATT&CK is a curated knowledge base, where Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK) are organized and categorized. It acts as a framework describing the actions that an adversary takes while operating within the network of an organization. [39] It reflects the various phases

of a cyber attack lifecycle and the platforms adversaries are known to attack. [38] MITRE is a non-profit company that aims to close gaps in the cyber defense of companies. [2] The overall structure of ATT&CK includes the following core components:

- **Tactics:** describe the tactical goals of an adversary
- **Techniques:** describe how adversaries aim to achieve their tactical goals

2.3.2 The MITRE ATT&CK Matrix The ATT&CK Matrix [3] visualizes the relationship between tactics and techniques. For example under the Column "Persistence" (tactic), meaning to persist in the target environment is the adversary's goal, we can find a series of techniques on how to achieve that. For example, Account Manipulation and Scheduled Task are separate techniques that may be used to reach the goal of persistence (see Figure 4) [38].

ATT&CK is organized in "Technology domains", which categorize the ecosystem the adversary is operating in. Until today, two domains are defined. **Enterprise** represents traditional enterprises and **Mobile**, which is for mobile communication devices. Technology Domains are further subdivided in platforms, which can be an operating system e.g. Microsoft Windows. Techniques can apply to multiple domains. In the course of this master's thesis, the focus is placed on the technological domain enterprise, which represents traditional enterprise networks.

2.3.3 Tactics Tactics are representative for the "why" of the adversary's techniques, the reason for performing an action. In the framework, tactics serve as useful categories for individual techniques. Tactics describe actions that adversaries do during an operation, such as persist, discover information, move laterally, execute files, and exfiltrate data. They act as labels to an associated technique. Within ATT&CK, a technique may be related to one or more tactic categories due to the various results that can be achieved by using a technique. Each tactic consists of a definition which serves as a guide for what techniques should be within. To give an example, "Execution" consists of techniques that result in execution of adversary-controlled code on a system. [38]

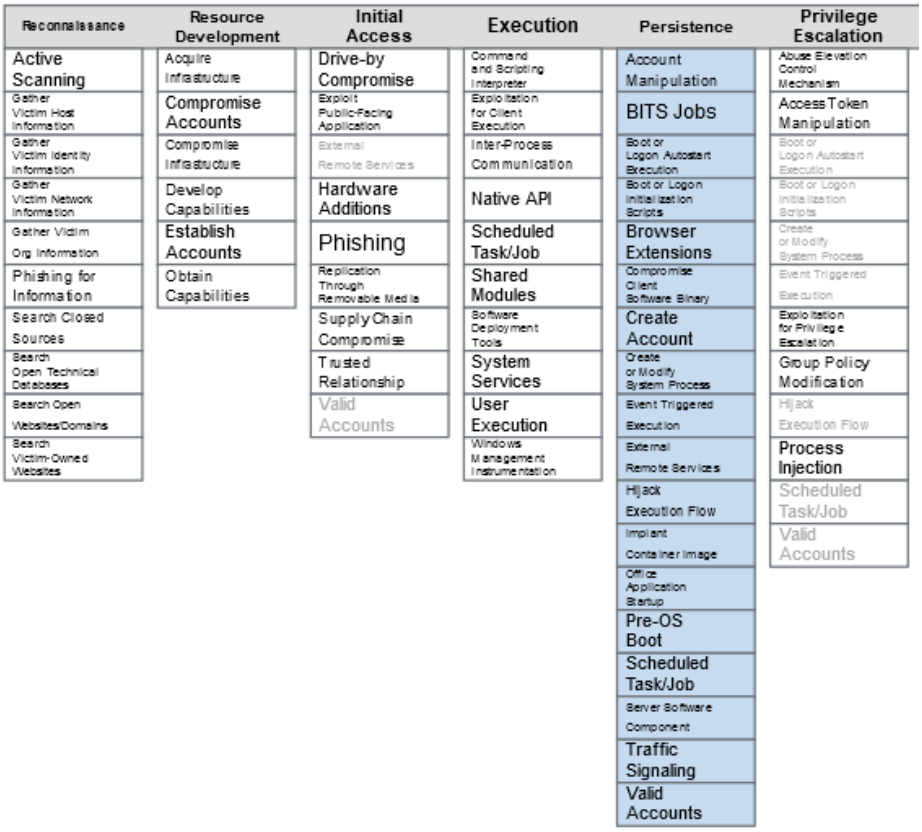


Fig. 4. Excerpt from the ATT&CK Enterprise Matrix [3]

Technology Domain	Platform(s) defined
Enterprise	Linux, macOS, Windows
Mobile	Android, iOS

Fig. 5. ATT&CK Technology Domains [38]

2.3.4 Techniques Techniques answer the "how" a tactical objective of an adversary can be achieved and are the foundation of ATT&CK. As there may be many ways to achieve tactical goals, there are multiple techniques in each category. Each technique in the ATT&CK model is based upon a technique object structure. In the ATT&CK, Technique Model describes the object structure

Data Item	Type	Description
Name*	Field	The name of the technique
ID*	Tag	Unique identifier for the technique within the knowledgebase. Format: T####.
Tactic*	Tag	The tactic objectives that the technique can be used to accomplish. Techniques can be used to perform one or multiple tactics.
Description*	Field	Information about the technique, what it is, what it's typically used for, how an adversary can take advantage of it, and variations on how it could be used. Include references to authoritative articles describing technical information related to the technique as well as in the wild use references as appropriate.
Platform*	Tag	The system an adversary is operating within; could be an operating system or application (e.g. Microsoft Windows). Techniques can apply to multiple platforms.

Fig. 6. Excerpt from the ATT&CK Technique Model [38]

of a technique, as seen in figure 6. There is a distinction between "Tag", an informational reference to filter or pivot on, and "Field", which is a text field used for the description of technique specific information and details, as seen in figure 6. [38] Techniques can be further divided into sub-techniques, which are more detailed variants of the specific technique. For example the technique "Phishing" consists of three sub-techniques. These are "Spearphishing Link", "Spearphishing Attachment" and "Spearphishing via Service". As there are various techniques when targeting a tactical objective, multiple techniques can be found in one tactic category. In this sense, there are several ways to perform a technique, which is why there are different sub-techniques under one.

In the next section, an approach is presented that provides a principle for the efficient establishment of a security concept to protect against these tactics and techniques.

2.4 The IT Baseline Protection

An essential prerequisite for successful information security management is the assessment of existing security risks. A risk analysis attempts to identify and assess these risks and thus determine the

overall risk. The goal is to subsequently reduce this risk to such an extent that the remaining residual risk becomes quantifiable and acceptable. This should be valid for the entire organisation and define how the objectives of risk analysis - identification and assessment of individual risks and overall risk - are to be achieved. Possible risk analysis strategies are: the detailed risk analysis, the IT baseline protection and the combined approach. [47] [15]
In the following chapter, the IT baseline protection approach "Grundschutzansatz" is discussed in more detail. Regardless of the actual need for protection, a generalised risk situation is assumed for all IT systems. So-called baseline security controls are used as security measures. By dispensing with a detailed risk analysis, this approach saves resources and quickly leads to a relatively high level of security. The disadvantage is that the baseline level of protection may not be appropriate for the IT system under consideration. [47] [15] The goal of the basic protection principle is to appropriately limit the effort required to create an information security concept. This is achieved by assuming a generalised risk situation and thus, dispensing with a detailed risk analysis. The security measures to be implemented are selected on the basis of predefined catalogues. Advantages of this approach are the reduction of the effort for risk analysis and that the use of basic protection measures quickly lead to a relatively high level of security against the most common threats. In addition, baseline protection measures are usually widespread and thus, relatively inexpensive and quick to implement. Disadvantages can therefore be that the basic protection level for the system under consideration are too high or too low. If it is too high, unnecessary financial and human resources are consumed; if it is too low, unacceptable risks may remain. Due to the lack of detailed risk analysis, it may be difficult to react appropriately to security-relevant hardware or software changes. The basic protection approach is thus recommended if it is clear that only IT systems with a low or medium ("normal") protection requirement are used in the area under consideration, or if there are still no or obviously too weak security measures in an area (e.g.: IT-System, department), the implementation of basic protection measures can help to quickly achieve a relatively good level of IT security. In this case, however, it should be checked in a subsequent step whether the level achieved is already sufficient or further

analyses and measures are required. [47] [15] Moreover, the baseline protection approach can be recommended as part of a comprehensive risk analysis concept in the sense of a combined approach. This can have the advantage of concentrating the workload for risk analysis and the selection of specific security measures on those in need of high protection. Basic protection measures can then be used for the remaining systems. [47]
The basic protection analysis essentially consists of two steps. In the first step, an IT system or an IT network is modelled using existing building blocks referred to as modelling in the following. In a second step, a target/actual comparison is made between existing and recommended measures. [47] [15]

2.4.1 Modelling The modelling of an IT system or an IT network is dependent on the underlying catalogue of building blocks and measures, as an attempt must be made to reproduce the system as accurately as possible through the existing building blocks. The central task of the "modelling" step is to reproduce the IT network under consideration with the help of the existing building blocks of IT-baseline protection. As a result, a model of the IT system is created, which consists of different, possibly also repeatedly used components of the security manual (e.g. "Österreichisches Sicherheitshandbuch") and contains a mapping between the components and the security-relevant aspects of the IT system.[47] [15] In order to facilitate the mapping of a generally complex IT system to the building blocks of the security manual, it is advisable to consider the security aspects grouped according to certain topics. [47] [15]

The security aspects of an IT network are assigned to the individual layers from figure 7 as follows: Layer 1 comprises all overarching security aspects that apply equally to all or large parts of the IT network. Typical components of layer 1 include information security management, organisation, data security concept and computer virus protection concept. Layer 2 deals with the structural-technical conditions in which aspects of infrastructural safety are brought together. This concerns particularly the building blocks of buildings, rooms, protective cabinets and the domestic workplace. Layer 3 concerns the individual IT systems of the IT network which may have been combined into groups. Here, the security aspects of not only clients

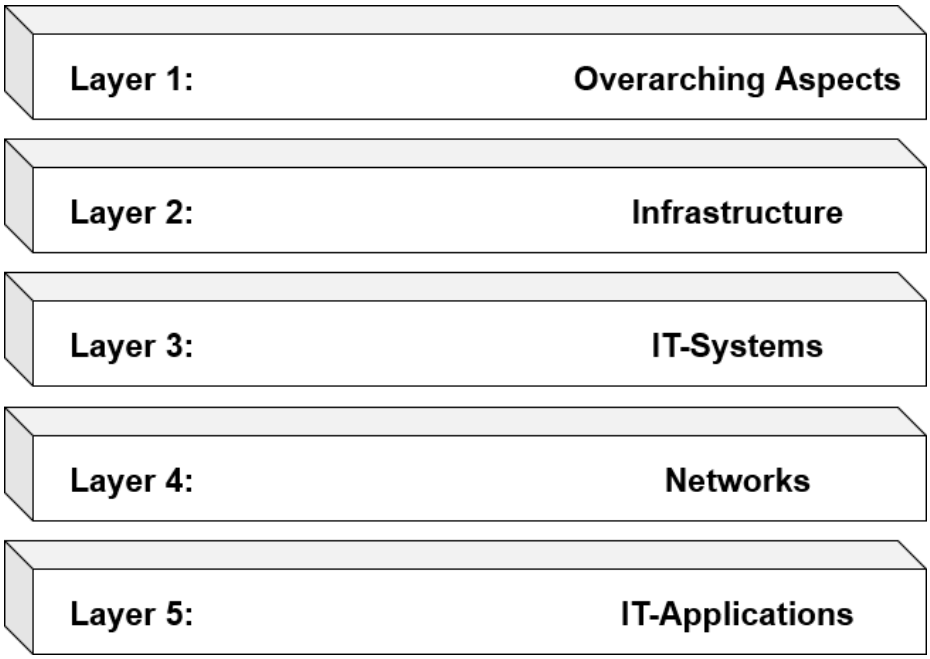


Fig. 7. IT baseline protection model - Layers [47]

and servers, but also stand-alone systems are dealt with. Layer 3 thus includes, for example, the components Unix system, portable PC and Windows NT network. Layer 4 looks at the communication and networking aspects of IT systems, such as the components network and system management and firewalls. Finally, layer 5 deals with the actual IT applications used in the IT network. In this layer, the components e-mail, web server, fax server and databases, among others, can be used for modelling. The task to be performed in this step is to replicate the real IT system as accurately as possible using the existing building blocks. Finally, it should be checked whether the modelling of the overall system is complete. It is recommended that the network plan or a comparable overview of the IT network is used for this and that the individual components are systematically gone through. Each component should either be assigned to a group or modelled individually. It is important that not only all hardware and software components are modelled in technical terms, but that

the associated organisational, personnel and infrastructural aspects are also fully covered.[47] [15]

2.4.2 Target-performance Comparison between existing and recommended measures In the second step of the basic protection analysis, modelling according to IT baseline protection is used as a test plan to determine which standard security measures have already been implemented, which have not been implemented or have been implemented insufficiently. The target consists of the measures recommended in the individual building blocks. The comparison with the existing measures results in the measures that still need to be implemented for IT baseline protection. For the establishment of basic IT protection, all basic IT protection measures proposed in the building block should be implemented in principle. However, it is possible that recommended basic protection measures cannot or should not be implemented in certain operational environments. This deviation from the recommendation must then be documented and justified. At this point, any existing IT security measures that go beyond basic IT protection should also be worked out and documented. As a result of the procedure described, a list of measures that still need to be implemented in order to achieve IT baseline protection is to be drawn up. [47] [15]

Based on this, it can be stated that the basic protection principle limits the effort to identify and implement computer security measures in an organization while leading to a high level of security.

3 Research Approach

This thesis draws on the methodology described in the work "Design Science in Information Systems Research", which has already been used successfully several times. [19] Design science is used as guiding idea to help achieve the goals of the thesis and assure that the approach taken follows good scientific practice. The aspects of design science described in this chapter are of core relevance while others are considered helpful but not central.

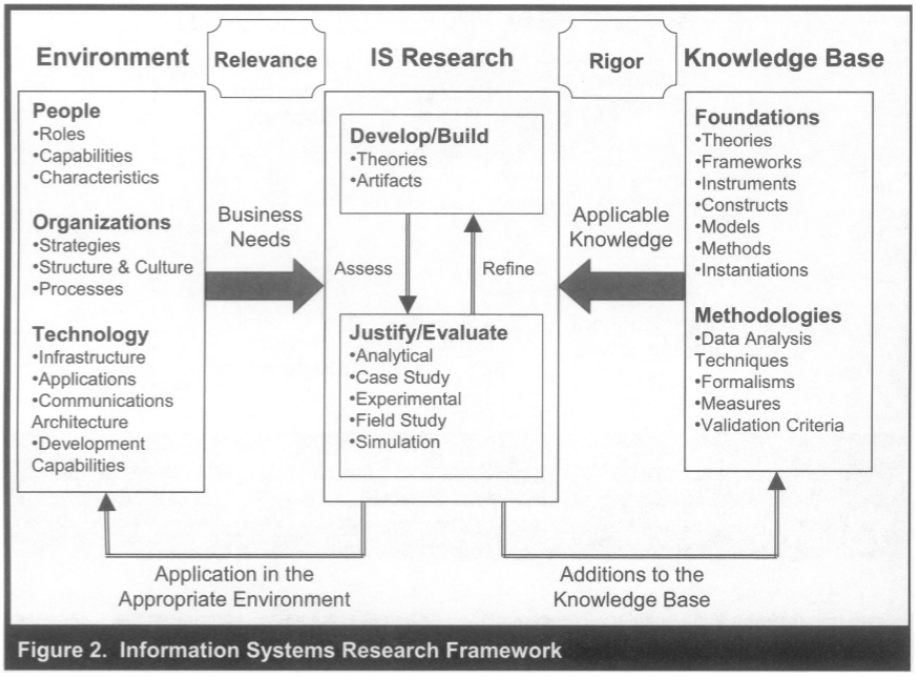


Fig. 8. Information System Research Framework [19]

3.1 Problem Relevance

In the course of the research, it became evident that the number of cyber attacks is constantly increasing and that there is a continuous challenge to protect organisations from attacks. Although there are already numerous countermeasures in the form of preventive, detective and protective countermeasures, no cyber security strategy has yet proven to be fully effective. This is the reason why a diverse combination of measures and the development of new approaches are needed to effectively combat threats. [6] However, as security policies represent a first line of defense, it is important to directly link them to the measures taken against cyber attacks. [44] Therefore, this paper examines the aspect of whether the incident could have been prevented by the prescribed behaviour derived from the security policy. The focus here is on the question of whether the incident is caused by non-implementation of recommended measures or a violation of the measures.

3.2 Research Rigor

The underlying methodology here can be described with the Information System Research Framework in figure 8. In this framework, behavioural research and design science paradigms are to be combined. The environment part of the framework defines the problem space which consists of the actors and their business needs. Once a business need is given, IS research is divided into two complementary phases. Behavioural science addresses research by developing and justifying theories that are scientifically close to the business need. Design science, in return, addresses research on building and evaluating artifacts that target the business needs. This correlation results in the interdependence between refinement and reassessment. The knowledge base part offers materials with which IS research can be conducted, divided into foundations and methodologies. [19] Rigor in this thesis is achieved through the application of existing foundations and methodologies.

3.3 Design as a Search Process

The search for a solution to the problem is based on an iterative development of the model. The cycle of identifying problems and posing solutions has been repeated until a result was gained with which the goals of this thesis were achieved. In a first instance, requirements are derived in form of measures from the IT baseline protection principle, to generate control variables. Subsequently, the model was adapted to gain a measurable result about the coverage of the security policy from these control variables. In a second step, control variables were again created to check compliance with the security policy. Afterwards, the model was adapted once more to make these results measurable. These measurable results are then evaluated and analysed with the help of scenario-based testing.

3.4 Design as an Artifact

In the course of the work, the following four artifacts are created. At first, the results derive from the literature research. Secondly, based on the literature research, a model is developed that is able to provide a concept for answering the research question. Thirdly, a

prototype is created as an installation of this model which is capable of generating measurable outcomes. Lastly, results are created, which describe the capability and limitations of the model.

3.5 Design Evaluation

The objects of this work are evaluated in two important steps that are independent of each other: In a first step, the requirements for the model are evaluated by means of a literature review. In a second step, the developed concept is tested by means of scenario-based testing with a developed prototype in order to obtain measurable results.

3.6 Research Contributions

The research contribution in this thesis is the model which is used to show that it is possible to construct an approach to answer the research questions "How can a process model be constructed and prototypically implemented that helps the user in determining why an incident has resulted in a successful attack?" and "What can be achieved with the developed model and what are the limitations?"

4 Model Development

In the course of this chapter, a model is developed which should enable the goals defined in the introduction to be achieved. The model developed in this chapter is then prototypically implemented in chapter 5. In the following sub-chapters, the model is visualised from different perspectives in order to guarantee a holistic understanding of the approach. The chapter is thus divided into the sub-chapters "Use Case Diagram representing the Scenario", "Process Model representing the paths of the Scenario" and "Component Diagram representing the Architecture".

4.1 Use Case Diagram representing the Scenario

In this part, the scenario, which is to be tested in the next section, is described by means of a use case diagram, accompanied by a textual description.

The use case has one main actor, which is consequently referred to as "user". From the user's perspective, there is one main use case "Review Incident" within the application. "Review Incident" is the process of reviewing if the user's security policy corresponds to the IT baseline protection ("Grundschutzprinzip") of the Austrian security manual [47] and if the user's policy was adhered to or not. By means of these two basic criteria, it should be determined if the incident could have been prevented and whether it failed due to the configuration of the policy or due to non-compliance with the policy. As shown in the Use Case diagram (see figure 9) "Review Incident" imports the behaviour of the other use cases "Choose Technique", "Select Testset", "Review Measures", "Inspect Evidences" and "Analyze Violations from Evidences". This means that while the use case "Review Incident" is in progress, the previously mentioned use cases are processed in separate steps, and their results are used in "Review Incident". In these sub-processes, the user's input serves to navigate through the process of the application which is examined in more detail in the subsection "Process Model representing the paths of the Scenario".

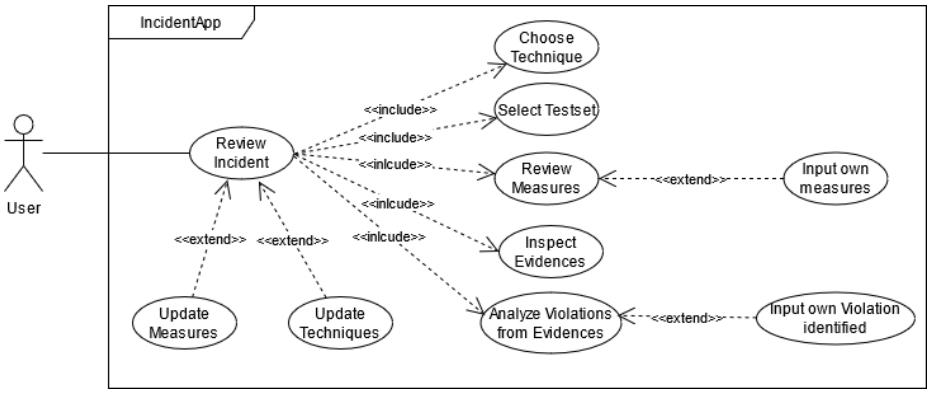


Fig. 9. Use Case Diagram of the Scenario

Furthermore, "Review Incident" can be extended by the two use cases "Update Measures" and "Update Techniques". These have the purpose of guaranteeing that the system is up-to-date, both with the

measures from the IT baseline protection and with the techniques of the attackers. "Review Measures" can be extended with the use case "Input own measures" and "Analyze Violations from Evidences" can be extended with "Input own Violation identified". These two extensions are intended to take into account input from the user that may be helpful for the process.

4.2 Process Model representing the paths of the Scenario

This section describes the process of the prototype. In order to provide a clear presentation of the process, it is divided into three subsections. The overview of the whole process is provided by figure 10.

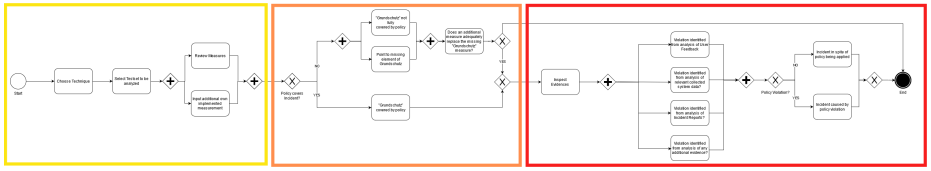


Fig. 10. Process Diagram Overview of the Scenario

The following sub-chapters explain the three colour-coded sections of the process.

4.2.1 Select Technique and Review Policy Section Following figure 11, The first task in the process lets the user select the desired technique in the step "Choose Technique". These techniques are retrieved via an interface from the MITRE ATT&CK matrix [3], displayed and provided with a brief description in order to help the user find the best fit to his incident. The next step "Select Testset to be Analyzed" provides the user with three different options for evidence data. These were prepared in order to conduct a scenario-based testing approach in chapter 6.

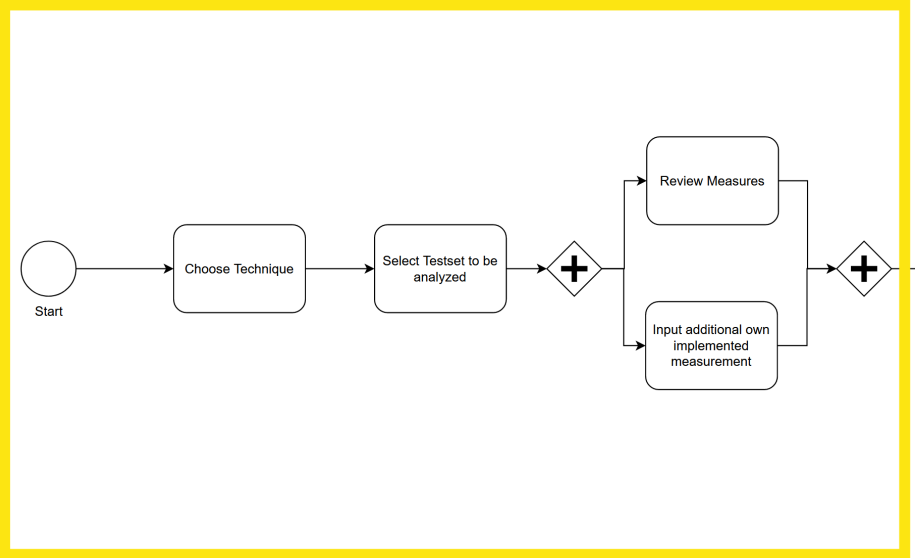


Fig. 11. Process Diagram - Select Technique and Review Policy Phase

In the following step, "Review Measures", questions are derived from relevant security measures based on the IT baseline protection "Grundschutz-Prinzip" of the "Österreichisches Sicherheitshandbuch" [47] which are intended to determine if the necessary measures are considered relevant for the incident and have been applied in the user's policy. Furthermore, the user is given the opportunity to enter his or her own additional measures in the parallel task "Input additional own implemented measurement".

4.2.2 Analyse Policy Section After processing if the measures have been taken, the policy can be considered as covered by the IT baseline protection ("Grundschutz"-covered) or not (see figure 12). If the policy is safe, the decision's branch is finished. If not, the user gets informed about the fact that the "Grundschutz" is not fully covered by policy and given a list of missing elements that should be covered.

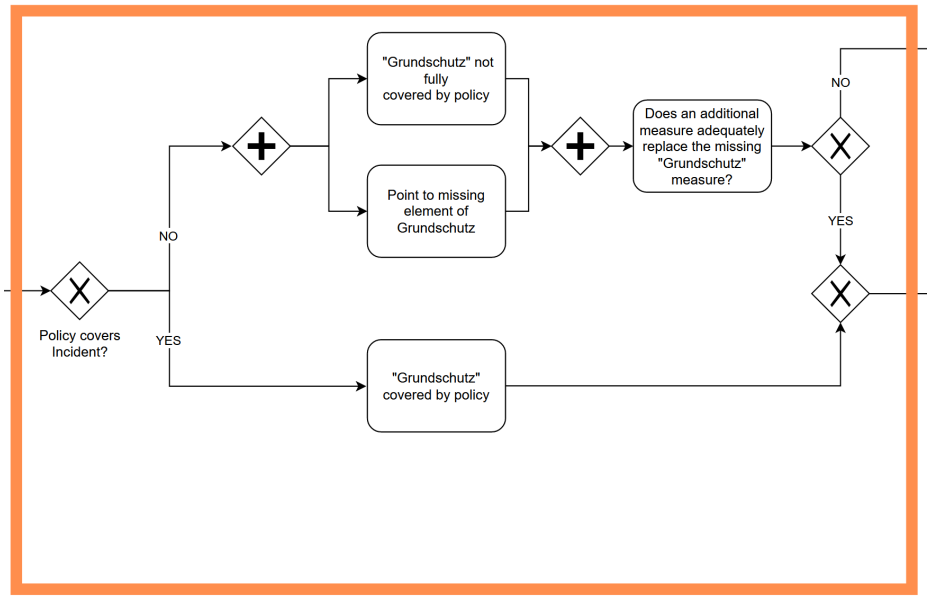


Fig. 12. Process Diagram Part - Analyse Policy Section

In "Does an additional measure adequately replace the missing 'Grundschutz' measure?" the user is asked if there is any other coverage by an additional measure that has been applied. If not, the process "Review Incident" is ended with a reference to the chapter "4.3 Grundschutzansatz" in the "Österreichisches Sicherheitshandbuch".

4.2.3 Inspect and Analyse Evidence Section If the policy has been declared safe in the previous "Analyse Policy" section, the user is instructed to view evidences in "Inspect Evidences" (see figure 13). The next step is to analyse the evidences for possible violations. In this step, the user is asked to evaluate each piece of evidence separately for a violation.

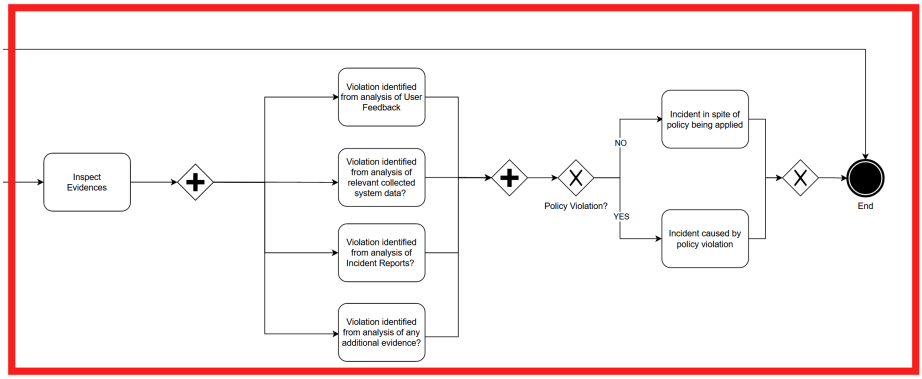


Fig. 13. Process Diagram - Inspect and Analyse Evidence Section

This is followed by the final evaluation, in which the user receives the message "Incident in spite of policy being applied" if the policy is adhered to, or "Incident caused by policy violation" if not.

4.3 Component Diagram representing the Architecture

Figure 14 shows the structure of the system in the form of a component diagram. The system is designed as a web application which ensures device-independent access and based on the RESTful architecture, which is lightweight and maintainable but still offers scalability. The architecture was divided into components which are extendable but also exchangeable and support the implementation as a web application based on the RESTful architecture.
The component "Frontend WebService" represents the interface for the user and is transferring data over HTTP requests with the component "Configuration Backend in both directions. The Component "Configuration Backend" handles the logic. Due to that, it is connected with all other components. The Component "System Database" offers CRUD Operations for the "Configuration Backend" and the components "Österreichisches Informationssicherheitshandbuch Measures", "ATT&CK Knowledge Base" and "Evidences Testsets". The component "Österreichisches Informationssicherheitshandbuch Measures" retrieves information from the most current version of the information security manual "Österreichisches Sicher-

heitshandbuch" [47], restructures it, and stores it in the database. Furthermore, an interface is provided to find, load and update measures, which is also used by the "Configuration Backend".

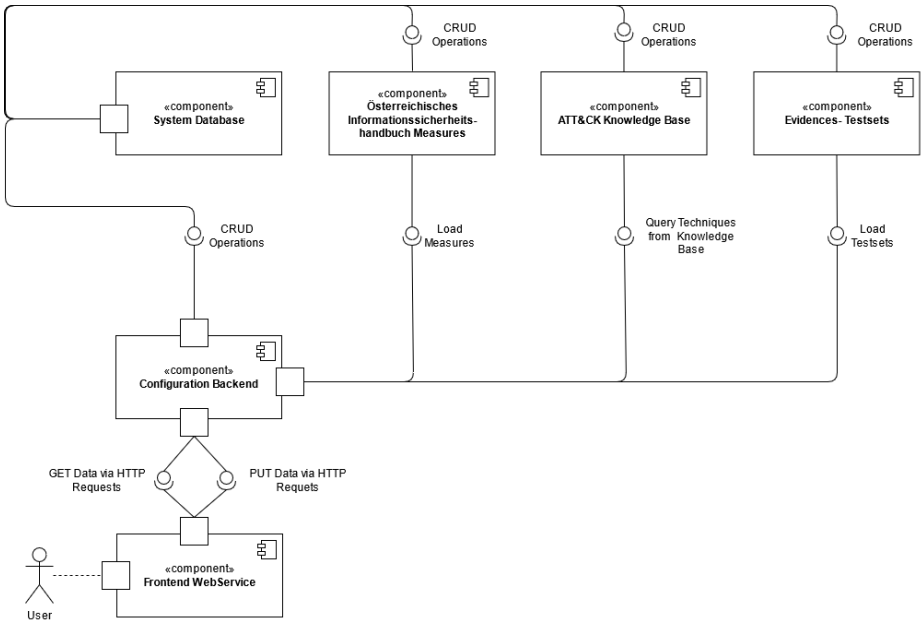


Fig. 14. Component Diagram Overview

The "ATT&CK Knowledge Base" component connected is to be perceived as an interface to the ATT&CK matrix and serves to retrieve crucial up-to-date information on techniques from attackers. Again, an interface for the "Configuration Backend" is provided here to be able to query techniques. The Evidences Test Sets" component contains the data preparation for the scenario-based testing. As before, an interface is provided to allow convenient access. The overall structure of the components mainly serves purposes of actuality with regard to prescribed measures and with regard to the techniques of attackers. Furthermore, this module structure is intended to provide scalability.

5 Prototype Implementation

This part explains and justifies the technology stack with which the model from the previous chapter is to be implemented in the course of a fast prototyping approach in order to achieve a proof of concept. The focus for the implementation is on the core competence of the model and thus, on answering the research question presented in the chapter "Goals". Furthermore, the state of the art techniques used here are intended to achieve scalability while at the same time using lightweight tools and techniques for fast prototyping purposes.

5.1 Reasons for implementing a RESTful Web App

A web application is made available via a server and can therefore be called up in the browser via a URL with different devices. The programming for this is cross-browser. Web apps are developed using CSS, Javascript and HTML. They function according to the client-server model and are therefore not installed on the respective end devices, so that the supply, processing and evaluation of the data takes place on a web server. Only the results of the data processing are displayed on the client device. This operating system-independent design ensures uncomplicated access to the application. [4] The design of the web app in the form of a RESTful architecture was chosen because of its lightweight nature, its easy development and use, its scalability and its widespread use in practice. [46] [16] Therefore, the prototype is implemented as a web application with a "RESTful Architecture".

5.2 Technology Stack

For the implementation, the technologies used as well as the reasoning for their application are stated as follows:

- **Python 3.9** [12]
  Python can be seen as an open source, general purpose, high level programming language that is efficient in terms of rapid development. This is possible due to the availability of standard libraries provided by the python community that can be downloaded on

demand. This keeps the code for web development short and simple. Python is robust, can work with all types of databases, is powerful in text processing and is compatible with a wide range of web technologies and remains the most popular programming language in 2020. [1] [37]

- **Flask 1.1.2** [33]
  Flask is a micro framework for python which fulfils the basic functionality of a web framework and at the same time offers extensibility. The term micro framework comes from the design of the core functionality being simple but extensible in terms of development. This makes it possible to develop web applications in a time-saving way. Flask, like the programming language it is based on, is open source and popular within the community.[1] [43]
- **Jinja2** [32] With Flask comes Jinja2 as a templating language. This templating language is needed to define placeholders in html templates for the varying number of elements and the examination of variables by means of control structures. Jinja2 is modern, fast, secure and widely used. [1] [32]
- **beautifulsoup 4** [40]
  Beautiful Soup is a Webscrape library which makes it easy to use to scrape information from web pages. It provides iterating, searching and modifying the parse of HTML or XML. Within this work, it is used for Webscraping the Website of the Österreichisches Sicherheitshandbuch. [35]
- **Mongo DB** [26]
  MongoDB is a universal, document-based, distributed database for modern application development that meets the highest demands in terms of productivity and is used worldwide for numerous products and services. [25] Mongo DB is used as a Database for central storage of data.
- **Flask Pymongo 2.3.0** [41]
  Flask Pymongo is a library that adds convenient features for applications which combine MongoDB with Flask. [41]
- **STIX 2 Python API and Taxii2client v20** [28] [24] [29]
  STIX 2 and Taxii2client are python libraries used to access the ATT&CK Data Model.

In the next part, the use of the technologies, frameworks and libraries presented here will be explained on the basis of the prototype by showing excerpts from the code.

5.3 Code Documentation

This part gives more details about the code and its documentation. The purpose of the documentation is to make the programming approach comprehensible and thus expandable in future work. The documentation is structured according to the components from the component diagram (see figure 14) from the previous chapter.

5.3.1 The ATT&CK Knowledge Base Component This Component is defined in the implementation by the class "mitre_api". The imported libraries that we see in figure 15 are used to access the content of the "MITRE ATT&CK Knowledge Base". [24] In the application, the content of the knowledge base is retrieved directly from the "ATT&CK TAXII server". TAXII, which stands for "Trusted Automated Exchange of Intelligence Information", is a protocol which enables the exchange of "Cyber Threat Intelligence" over HTTPS and defines a RESTful API. [31] In the component we import **Server** at first, which can be understood as an Instance of the TAXII API. Furthermore, we import **Collection** which is an interface to a Cyber Threat Intelligence object provided by a TAXII Server [31] and which is in this case hosted by ATT&CK.[24] Stix2, which stands for Structured Threat Information Expression is a language and serialization format that is is used to exchange cyber threat intelligence. As the content of the ATT&CK TAXII server is expressed in stix objects **TAXIICollectionSource** and **Filter** are imported (see figure 15). The first one provides an interface for retrieving STIX objects from the TAXII Collection endpoint, while the latter allows us to use filters for queries.[42] At first, we instantiate a server object with the url for the ATT&CK Knowledge Base and afterwards we take an API root instance. We then choose the "enterprise_attack" collection for querying it. The code fot the access to the repository described here was taken from the section "Accessing ATT&CK data in python" from the official documentation referenced in [24].

```python
from stix2 import TAXIICollectionSource
from stix2 import Filter
from taxii2client.v20 import Collection # only specify v20 if your installed version is >= 2.0.0
from taxii2client.v20 import Server # only specify v20 if your installed version is >= 2.0.0
from flask import Flask, jsonify
from flask_pymongo import PyMongo


# Connection to MITRE ATT&CK Framework
server = Server("https://cti-taxii.mitre.org/taxii/")
api_root = server.api_roots[0]
collections = {
    "enterprise_attack": "95ecc380-afe9-11e4-9b6c-751b66dd541e",
    "mobile_attack": "2f669986-b40b-4423-b720-4396ca6a462b",
    "ics-attack": "02c3ef24-9cd4-48f3-a99f-b74ce24f1d34"
}

collection = Collection(f"https://cti-taxii.mitre.org/stix/collections/{collections['enterprise_attack']}/")
src = TAXIICollectionSource(collection)
```

Fig. 15. ATT&CK Knowledge Base Component - Knowledge Base Connection

```python
# Get Technique by ID
def get_technique(id):
    tec = src.query([ Filter("external_references.external_id", "=", id) ])[0]
    return tec

# Connection to Database
app = Flask("__name__")
mongodb_client = PyMongo(app, uri="mongodb://localhost:27017/incidentwebapp_DB")
db = mongodb_client.db
# List of techniques considered for Incidentwebapp -> can be updated by adding ATT&CK-ID of new Technique into List
techniques = ["T1566","T1134"]
# Upate Techniques in Database
def update_techniques():
    db.techniques.drop()
    for t in techniques:
        technique = get_technique(t)
        db.techniques.insert_one({'name': technique["name"], 'description': technique["description"], 'chosen': False})
```

Fig. 16. ATT&CK Knowledge Base Component - Functions

An important function in this context is "get_technique(id)", see figure 16 which can retrieve the techniques from the selected "Enterprise-Collection" and return them. In a further step, a connection to the database is established to retrieve the techniques occuring in the list which is expandable by adding more techniques.Subsequently, the information relevant for the application is entered into the collection "techniques" in the database.

5.3.2 The Österreichisches Informationssicherheitshandbuch Measures Component

The Österreichisches Informationssicherheitshandbuch Measures Component is defined in the prototype by

the class "measures_api". The imports in figure 17 are mainly used to reverse engineer the website of the Österreichisches Sicherheitshandbuch (see [47]). Therefore, in this components the recommendations of the Österreichisches Sicherheitshandbuch regarding the IT baseline protection "Grundschutzansatz" are made accessible. In this class, in addition to Flask [33] and PyMongo [41], two additional libraries are imported which serve the purpose of downloading the website of the "Sicherheitshandbuches", parsing it and transforming it into objects which remain consistent with the content and structure but are convenient for the structure of the application. On the one hand, this is the "Pywebcopy" library and on the other, "Beautifulsoup" library (see figure 17). With the former, the webpage can be efficiently downloaded during runtime and stored in the application's directory. The content is then made accessible for the second import by means of a file reader. Beautifulsoup is used to navigate the content along the HTML tree and to extract relevant information.

```python
from bs4 import BeautifulSoup
from pywebcopy import save_webpage, save_website, config
from flask import Flask
from flask_pymongo import PyMongo


# READ Sicherheitshandbuch
with open("resources\www.sicherheitshandbuch.gv.at\www.sicherheitshandbuch.gv.at\95b74696__siha.php", "r", encoding='utf-8') as f:
    text= f.read()
soup = BeautifulSoup(text,'html.parser')
soup2 = BeautifulSoup(text,'html.parser')


# relevant measures for General -> can be updated by Inspecting "https://www.sicherheitshandbuch.gv.at/"
meas_list1 = ["topic_733", "topic_734","topic_738", "topic_739", "topic_741", "topic_743", "topic_747", "topic_748", "topic_749", "
# relevant measures for Phishing -> can be updated by Inspecting "https://www.sicherheitshandbuch.gv.at/"
meas_list2 = ["topic_865", "topic_430", "topic_868", "topic_869", "topic_883", "topic_893", "topic_900", "topic_3038", "topic_909",
# put measure into python dictionary

def find_byid(topic):
    result = {}
    cid = soup.find(id=topic)

    result["title"] = cid.find('h3').get_text()

    des = soup2.find(id=topic)
    for h3 in des('h3'):
        h3.decompose()

    for a in des.findAll('a'):
        a.replaceWithChildren()

    result["description"] = des.prettify()
    result["topic"] = topic

    return result
```

Fig. 17. Österreichisches Informationssicherheitshandbuch Measures Component - Queryiing Functions

When the application is started, "pull_url()" is called (see figure 18), which stores the website in the "resources" directory.

```python
# Pull Sicherheitshandbuch from URL
def pull_url():

    url = 'https://www.sicherheitshandbuch.gv.at/'
    kwargs = {
     'bypass_robots':False,
     'debug': False,
     'project_folder': 'resources',
     'load_css': False,
     'load_images': False,
     'load_javascript': False,
     'over_write': True,
     'zip_project_folder': False

     }

    save_webpage(url,**kwargs)
```

Fig. 18. Österreichisches Informationssicherheitshandbuch Measures - Webscraping Access

The core competence of this class is to store a "measure object" in the database by means of an "ID" which can be found using the URL of the page. Figure 19 shows that this "ID" can be accessed without developer tools on the official website of the "Österreichisches Sicherheitshandbuch". [47]

Fig. 19. Österreichisches Informationssicherheitshandbuch - Official Website [47]

In order to determine the applicability of the scenarios for the scenario-based testing approach, the topics within the component were limited to the technique "phishing". However, it can be shown that by inserting IDs into the lists, see figure 17, it is easy to build up a catalogue of measures for another technique. The function "find_byid()" itself takes the identification number of the topic and traverses the website. If the topic is found, the title, the description and the topic number are stored in a "topic object", a Python dictionary. The figure 20 shows that "update_measures()" loads the appropriate set of measures into the database for the subsequent tasks, depending on which technique is chosen. The sets of measures are defined by "meas_list1" and "meas_list2" (see figure 18). "meas_list2" is a list containing the measures identified as relevant for the technique "phishing" in the form of the identification num-

ber. "meas_list1" contains general measures and is only prepared for future work.

```
63  # Update Measures in DB
64  def update_measures(choice):
65      db.measures.drop()
66
67      if choice == "Phishing":
68          meas_list = meas_list2
69      elif choice == "Access Token Manipulation Mitigation":
70          meas_list = meas_list1
71
72      for m in meas_list:
73          measure = find_byid(m)
74          db.measures.insert_one({'title': measure["title"], 'description': measure["description"], 'topic': measure["topic"]})
75
76  # get whole chapter
77  def get_chapter(chapterid):
78      resultlist = []
79      #find chapter
80      cid = soup.find(id=chapterid)
81      for topic in cid(class_="topic"):
82          resultlist.append(find_byid(topic['id']))
83
84      for r in resultlist:
85          db.chapter.insert_one(({'title': r["title"], 'description': r["description"], 'topic': r["topic"]}))
86
87  # get SHB Structure
88  def get_shbindex():
89      resultlist = []
90      cid = soup
91      #find all chapters
92      for chapter in cid(class_="chapter"):
93          #print(chapter['id'])
94          print(chapter.h1.get_text())
95          #find all sections
96          for section in chapter(class_="section"):
97              print(section.h2.get_text())
```

Fig. 20. Österreichisches Informationssicherheitshandbuch Measures Component - Update Measures

Furthermore, it can be seen that two additional functions are prepared in figure 20, which also may be used in future work. "get_chapter()" loads all topics from a chapter into a list, which can be stored in the database later. "get_shbindex()" returns the content of the "Sicherheitshandbuch" by chapters and the corresponding measures.

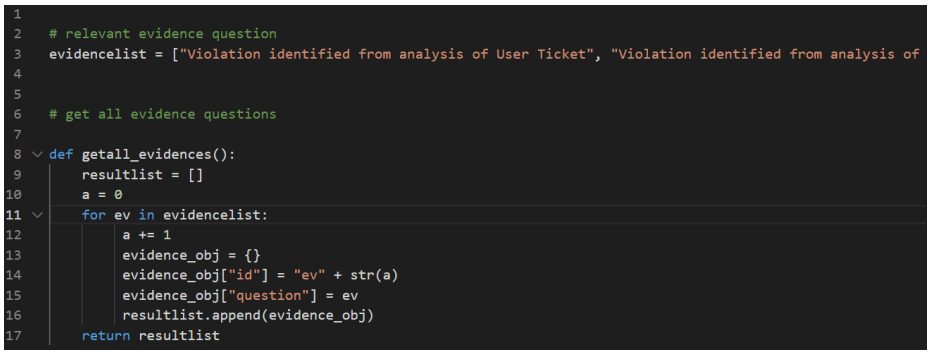
5.3.3 The Evidences-Testsets Component The Evidences-Testsets Component is defined by the class "cases_api" and "evidence_api". There is only on import "datetime" which serves to generate timestamps for evidences. As we can see in figure 21, which shows an excerpt from the program code of the class, within this class evidence is generated and added to lists according to the user's selectable testsets. The three different testsets have the same structure and each consists of a sample user ticket, a sample spam report

and a sample incident report. While this structure was retained to provide comparability within the scenario-based testing approach, the contents were designed differently in order to be able to check different test scenarios.

```python
# Testset Dictionaries

def get_testset_1():



    userticket = {
        "User Ticket": "#83783463",
        "Subject": "Security Issue from Email",
        "Description": "I opened a link in an email, although I did not check the sender beforehand. In this link, I have released data
    }
    spamfilter = {
        "Norton overview": "Spam Report",
        "Number of mails in spam folder": 9,
        "Most recent spam mail": "<admin@iqgtmipt.com> Make money by answering question",
        "Deleted": "Yes"
    }
    incidentreport = {
        "Security Incident Report": "#32432",
        "Date": str(date.today().strftime("%b-%d-%Y")),
        "Reported by": "Mr. Fally",
        "Organizational Unit": "Controlling",
        "Contact": "+9772828",
        "Location": "Mailing Address",
        "Incident Description": "This morning, an employee received an email. At some point before 09:18, the employee clicked one of th

    }
    testsetlist = []
    testsetlist.append(userticket)
    testsetlist.append(spamfilter)
    testsetlist.append(incidentreport)

    return testsetlist


def get_testset_2():
```

Fig. 21. Evidences-Testsets Component - User Ticket, Spam Report, Incident Report

This class can be used to obtain and expand any number of test-sets, which can then be stored in the database.



```
# relevant evidence question
evidencelist = ["Violation identified from analysis of User Ticket", "Violation identified from analysis of

# get all evidence questions

def getall_evidences():
    resultlist = []
    a = 0
    for ev in evidencelist:
        a += 1
        evidence_obj = {}
        evidence_obj["id"] = "ev" + str(a)
        evidence_obj["question"] = ev
        resultlist.append(evidence_obj)
    return resultlist
```

Fig. 22. Evidences-Testsets Component - Questions

In "evidence_api" (see figure 22) another possibility is offered to expand the questions for the policy violation check by adding questions to the list as a string.

5.3.4 The Configuration Backend Component The Configuration Backend Component is defined by the class "routes". In this component, the other components "measures_api", "mitre_api", "cases_api" are brought into contact with each other via the database. Because of this, the aforementioned components can be found in the imports (see figure 23).

The remaining imports are used solely for the operability of the Flask framework. In a first step, a connection to the database is established which is used in the subsequent functions. Before each function, a route is specified which represents the end point in the Flask RESTful Framework. The function "index(), which can be found under the two endpoints "/" and "/index", represents the entry point of the process for the user. A call to the endpoint specified in the route calls the functions defined at the endpoint. A distinction is therefore made between the HTTP commands GET,PUT,POST and DELETE. If not additionally specified, an HTTP command corresponds to GET by default. The corresponding HTML template is then loaded in a return statement and filled with the passed parameters. The endpoints "/updatetechniques" and "/updatemeasures" respectively trigger the update functionalities of the components to guarantee that the process works with the latest techniques and mea-

```python
from flask import Flask, render_template, request, jsonify, url_for
from app import app, measures_api, mitre_api, cases_api
from flask_pymongo import PyMongo
from werkzeug.utils import redirect

# PyMongo Connection Approach
app.config["MONGO_URI"] = "mongodb://localhost:27017/incidentwebapp_DB"
mongo = PyMongo(app)


@app.route('/')
@app.route('/index')
def index():
    user = {'username': 'User'}
    mongo.db.techniques.update_many({}, {"$set": {'chosen': False}})
    return render_template('index.html', title='Home', user=user)


@app.route('/updatetechniques')
def update_tec():
    mitre_api.update_techniques()
    return redirect(url_for('index'))


@app.route('/updatemeasures')
def update_meas():
   measures_api.pull_url()

   return redirect(url_for('index'))
```

Fig. 23. Configuration Backend Component - Imports, DB Connection and "/index" endpoint

sures. However, this functionality is also incorporated into the process and does not have to be executed separately.

```python
@app.route('/technique', methods=['GET', 'POST'])
def get_technique():

    if request.method == 'GET':
        # Update and Get all available Techniques from Database

        techniques = list(mongo.db.techniques.find({}))
        return render_template('technique.html', title='Choose Technique', techniques = techniques)
    elif request.method == 'POST':
        tec = request.form["answer"]
        dbres = mongo.db.techniques.find_one_and_update({"name": tec}, {"$set": {'chosen': True}})

        #load relevant measures set
        choice = mongo.db.techniques.find_one({"chosen": True})
        #load measures for chosen technique
        measures_api.update_measures(choice["name"])

        return redirect(url_for('select_case'))
```

Fig. 24. Configuration Backend Component - "/technique" endpoint

The end point "/technique" distinguishes between the HTTP methods GET and POST (see figure 24). When the endpoint is

called, the available techniques are loaded. The POST method, which is triggered by the user selecting a technique, loads the corresponding measures from the component "Österreichisches Informationssicherheitshandbuch Measures".

```
@app.route('/case')
def select_case():
    return render_template('cases.html', title='Choose TestCase')

@app.route('/case', methods=['POST'])
def get_case():
    global testset

    if "testset_1" in request.form:
        testset = cases_api.get_testset_1()

    elif "testset_2" in request.form:
        testset = cases_api.get_testset_2()

    elif "testset_3" in request.form:
        testset = cases_api.get_testset_3()

    #DELETE FROM DB AND INSERT NEW
    mongo.db.testset.drop()
    mongo.db.testset.insert(testset)
    return redirect(url_for('taskgetall'))


@app.route('/reviewmeas')
def taskgetall():

    measurelist = list(mongo.db.measures.find({}))
    return render_template('review_meas.html', title='Review Measures', measurelist = measurelist)
```

Fig. 25. Configuration Backend Component - Testset selection

In a next step, shown in figure 25, the user is offered three different testsets under the endpoint "/case". Depending on the input and thus the decision for a certain test set, this is then loaded from the component "Evidences-Testsets" and stored in the database. After the user is redirected to the endpoint "/reviewmeas", the actions from the responsible component are listed. The measures are transferred to the template as a list. In figure 26 we see how the list is processed with the help of the templating language Jinja [32] according to the varying number of elements by applying a for loop. In addition to each measure, a response block is also generated in this template.

The POST method at the endpoint "/reviewmeas" transfers the user's input to the backend. Based on the input, a logic is used to

```
{% extends "base.html" %}

{% block content %}
<h1>Review Measures</h1>
{% for measure in measurelist %}
<div>
    <h3>{{ measure.title }}</h3>
<button type="button" onclick="toggle{{ loop.index }}()">Show Details</button>
<div id="ev_element{{ loop.index }}" style="display:none">
{{measure.description|safe}}
</div>

    <script>
        function toggle{{ loop.index }}() {
        var x = document.getElementById("ev_element{{ loop.index }}");
        if (x.style.display === "none") {
            x.style.display = "block";
        } else {
            x.style.display = "none";
        }
        }
    </script>

</div>
<br>
<div class="form">
    <form method="POST" action="/reviewmeas">
        <label class="container"><b>Do you consider this measure as relevant for the incident under investigation?
            <input type="radio" name="rel_{{measure.topic}}" value= "TRUE" checked> Yes
            <input type="radio" name="rel_{{measure.topic}}" value="FALSE"> No
        </label>
        <br><br>
        <label class="container"><b>If considered relevant, has this measure been applied?</b>
            <input type="radio" name="{{measure.topic}}" value= "TRUE" checked> Yes
            <input type="radio" name="{{measure.topic}}" value="FALSE"> No
        </label>
        <br><br>
        {% endfor %}
        <p><b>Did you apply any additional measures?</b></p>

        <textarea id="ownmeas" name="addmeas" rows="4" cols="50">Please enter here.</textarea>
```

Fig. 26. HTML Template for reviewing measures

create a result of the analysis which is displayed to the user. Depending on the result, a different endpoint is called and the distinction in the result between "policy lack" or "policy coverage" corresponds to the process model 10. If the policy complies with the IT baseline protection "Grundschutz", the next step is to try to find out at the endpoint if the policy has been adhered to. As a result, the user is presented with a list of evidences at the endpoint "/inspect", corresponding to the previously selected testset (see figure 27).

```python
@app.route('/reviewmeas', methods=['POST'])
def get_userinput():

    dbmeasurelist = list(mongo.db.measures.find({}))

    for i in range(len(dbmeasurelist)):
        topic = dbmeasurelist[i]["topic"]
        rel = request.form["rel_"+str(topic)]
        apl = request.form[topic]
        mongo.db.measures.find_one_and_update({"topic": topic}, {"$set": {'applied': apl, 'relevant': rel}})

    policylist = list(mongo.db.measures.find({}))

    addmeas = request.form["addmeas"]
    #Results Measures Analysis
    one_applied = "TRUE"
    for pol in policylist:
        if pol["applied"] == "FALSE":
            one_applied = "FALSE"
    if one_applied == "FALSE":
        return render_template('policylack.html', title='Policy Lack', policylist = policylist, addmeas = addmeas)
    else:
        return render_template('policycoverage.html', title='Policy Coverage', policylist = policylist, addmeas = addmeas)


@app.route('/inspect')
def inspect_evidence():
    testset = list(mongo.db.testset.find({}, {'_id': False}))
    return render_template('inspect.html', title='Inspect Evidence', testset = testset)

```

Fig. 27. Configuration Backend Component - "/reviewmeas" and "/inspect" endpoint

```python
@app.route('/analysis')
def get_evidence():
    global evidencelist
    evidencelist = evidence_api.getall_evidences()
    testset = list(mongo.db.testset.find({}, {'_id': False}))
    return render_template('analysis.html', title='Analyse Evidence', evidencelist = evidencelist, testset = testset)


@app.route('/analysis', methods=['POST'])
def get_evidenceinput():
    answerlist = []
    for i in range(len(evidencelist)):
        answer = {}
        answer["id"] = evidencelist[i]["id"]
        answer["question"] = evidencelist[i]["question"]
        answer["violation"] = request.form[evidencelist[i]["id"]]
        answerlist.append(answer)

    print(answerlist)
    ownanalysis = {}
    ownanalysis["ownquestion"] = request.form["addanalysis"]
    ownanalysis["violation"] = request.form["own_violation"]
    #Results Evidence Analysis
    one_violated = "FALSE"
    for ans in answerlist:
        if ans["violation"] == "TRUE":
            one_violated = "TRUE"

    if one_violated == "TRUE" or ownanalysis["violation"] == "TRUE":
        return render_template('pol_violated.html', title='Policy Violation', answerlist = answerlist, ownanalysis = ownanalysis)
    else:
        technique = mongo.db.techniques.find_one({'chosen': True})
        policylist = list(mongo.db.measures.find({}))
        return render_template('pol_applied.html', title='Policy applied', answerlist = answerlist, ownanalysis = ownanalysis, technique


@app.route('/end', methods=['GET'])
def get_earlyend():
    return render_template('earlyend.html', title='End')
```

Fig. 28. Configuration Backend Component - "/analysis" endpoint

Consequently, this evidence is to be analysed in a further step in figure 28. This is realised under the endpoint "/analysis". Using the GET method, the evidences from the corresponding component are listed again and accompanied by a question. Using the POST method, the user's entries are then analysed and evaluated using the application's logic. Within this logic, the question of whether or not the specified measures have been complied with is investigated. Depending on the result, the user is forwarded to another endpoint and receives a different template. The endpoint "/end" exclusively concerns the abbreviated branch that ends if the IT baseline protection "Grundschutz" was not implemented in the incident in question.

Furthermore, there are two classes "__init__.py" and "incidentapp.py" in the application which are not assigned to any component. These are necessary to start the application or to make it plug and play capable in the future. In order to create a foundation for the expandability of a larger project, the project structure was based on the following tutorial referenced in [18].

5.4 Installation of the Prototype

This part explains how to download and run the prototype. In order to be able to proceed with this instruction, the following prerequisites must be met in advance:

1. The Python version from the technology stack (see 5.2) must be installed on the system. Please refer to the Python documentation for an installation guide.
2. Please clone or download zip from `https://github.com/FFally/incidentwebapp` (see figure 29) or copy from the disk attached at the submission.

After the repository has been downloaded or cloned and stored in the file system of the respective operating system, only the dependencies need to be installed. It is recommended to use the package manager "pip" which comes packaged with the recommended version of python for the subsequent steps. It is necessary to navigate to the projects folder using termina and make sure to find the root folder of the incidentwebapp, where the "incidentwebapp.py" are among other files and directories. First, a virtual environment should be created

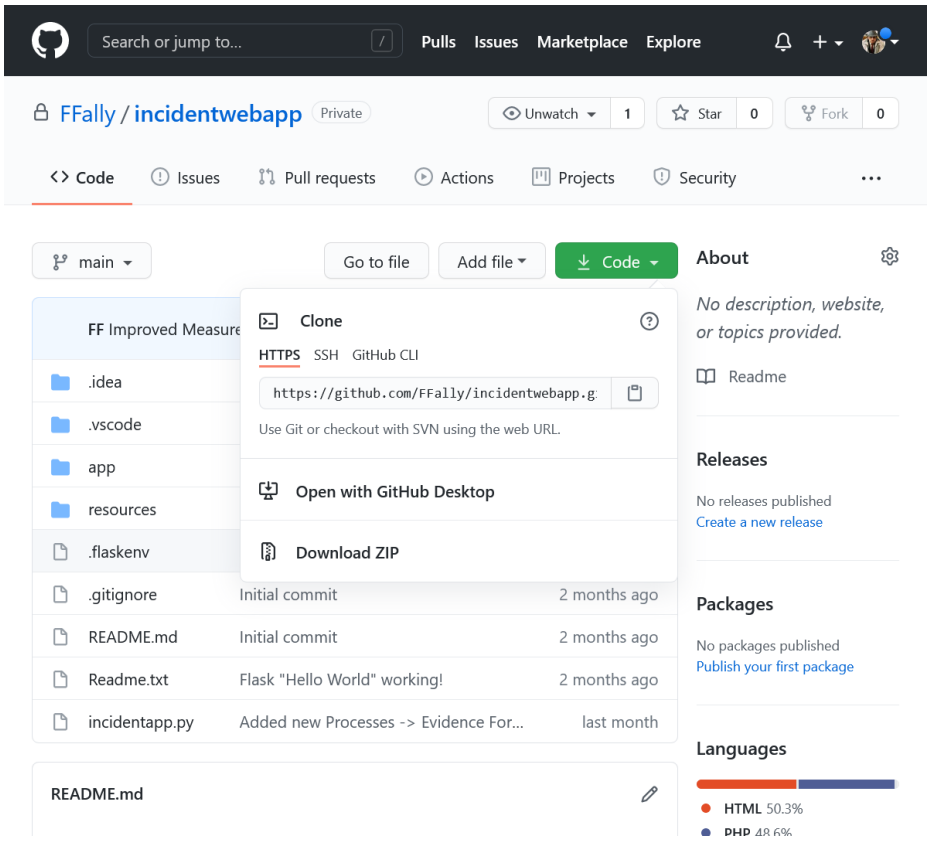


Fig. 29. Clone/Download Repository

to prevent the packages to be installed from interfering with global Python packages. A new virtual environment should be created by using the following command:

```
python -m venv .venv
```

After this command has been executed, the virtual environment is created and can be activated. On Windows:

```
.venv\Scripts\activate.bat
```

On other operating systems:

```
source .venv/bin/activate
```

After running the activate command the name of the environment ".venv" is shown within the terminal. In a last step for the installation the following code is run:

```
pip install -r requirements.txt
```

The directory can then be opened with a development environment such as "Visual Studio Code". Please make sure that the previously created virtual environment is selected as the interpreter of the project. Afterwards, only "incidentwebapp.py" has to be executed to get the web application running. Afterwards, a new browser window has to be opened and this must be typed into the address line:

```
localhost:5000/
```

After successful loading, the user finds himself at the beginning of the process.

6 Scenario-based Testing of the Prototype

In this part, the proof of concept is to be demonstrated by means of a scenario-based testing approach. The aim is to check all possible variants of the scenario for correctness. In doing so, all paths developed in the process model should be reached. The following variants of the scenario are supported in the process:

- The user's policy is covered by the IT baseline protection, but was not adhered to.
- The user's policy does not cover IT baseline protection.
- The incident occurred even though the user's policy covers the IT baseline protection and was adhered to.

These three possible variants of the scenario are examined within the scenario-based testing approach. The circumstance of the user's underlying policy is explained at the beginning of each scenario, as is the selection of one of the three evidence testsets. A cycle of a scenario is divided into two questions. "Does the policy cover the IT baseline protection (Grundschutz)?" and "Has there been a policy violation?". The scenarios are limited to incidents that can be linked to the attack technique "Phishing". The measures are selected

from the "Österreichisches Sicherheitshandbuch [47] and should represent measures that can be linked to the IT baseline protection for "Phishing".

6.1 Phishing as a representative Example

The Verizon report shows that phishing is the second most common threat when it comes to incidents is the main cause of data breaches in 2020, as it was the year before. [36] According to proofpoint [20], more than 5 million suspicious emails were reported via their PhishAlarm tool in 2020.

Modus Operandi of Phishing Attacks According to [3] phishing is classified as an initial access tactic. This set of tactics consists of techniques that serve the purpose of gaining initial access to the victim's system. From this initial access, the attacker then attempts to gain continuous access. The most common technique to gain initial access is known as phishing. In this technique, so-called phishing messages are sent to gain access to the victim's system. Phishing is always characterised by the main element of electronically delivered social engineering. The technique can also be divided into targeted phishing, also known as spearphishing, and non-targeted phishing. While the former mainly targets companies, industries or a specific person, the latter is characterised by spam campaigns addressed to the masses. In particular, the attacker sends emails with malicious content to victims with the aim of executing malicious code on the victim's system or collecting access data. In this context, three sub-techniques arise: spearphishing attachment, spearphishing link, spearphishing via service. Spearphishing attachment means that a file is attached to the spearphishing email and usually triggered by the recipient's action. Attachments can be Microsoft Office documents, executable PDFs or archived files. The text in the email tries to give the addressee a compelling reason why he should open the attachment and, if necessary, also explains how to circumvent the access restrictions of the system. In addition, it may also contain instructions how to decrypt an attachment in order to evade email boundary defenses. Attackers may also manipulate file extensions and icons to disguise executable files and make them appear like

documents. When the user executes the attachment, the message has reached its objective. In the case of "spearphishing link", the malicious email contains a link, again accompanied by social engineering text. The website to be visited may compromise the web browser or the user may be asked to download applications, zip files or other executable files. In "Spearphishing via Service" the attacker sends messages via various social media services or other services not controlled by the company. Again, the goal of the message is to get the victim's attention. Common methods are fake social media accounts or messages to the employee about potential job offers. This gives the attacker a good reason to ask the victim about policies, services or software used in the company. Through these services, the attacker can then send links or attachments. [3] Due to the widespread use of phishing and its effectiveness [10], as mentioned above, this attack technique was chosen for the scenario-based testing approach.

6.2 Selection of Measures to be reviewed

To answer the question "Does the policy cover the IT baseline protection (Grundschutz)?", control variables in the form of measures are needed. Measures that are displayed to the user for review within this scenario-based testing approach are taken from the "Österreichisches Sicherheitshandbuch". [47] All measures were taken from the chapter "13.2 Informations-und Datenaustausch", which is concerned with the exchange of information and data. Since it has already been stated that the technique of phishing is mainly carried out through the exchange of messages, these measures were found to be most relevant for incidents concerning phishing.

6.3 Types of Evidence to be analyzed

In order to answer the question "Has there been a policy violation?", the user is presented with evidence that can be checked for a violation of the policy. In order to create comparison, each evidence testset is equipped with the same types of evidence, but with different content. Within each testset there is a user ticket, a spam report and a security incident report. These types of evidence were chosen because they could be easily linked to phishing. For the purposes

of this scenario-based testing approach, these three types of digital evidence will be treated as evidence that determines whether or not a violation of the security policy has occurred. [8]

6.4 Scenario for an Incident with Policy Violation

In this first scenario, a test set is loaded that depicts the situation of an incident in a company with an extensive policy which covers the IT baseline protection for phishing. Goal of this test is to check if the algorithm responds correctly where a user enters an incident that is caused by policy violation. In the following, a user ticket, a spam report and an incident report serve as evidence. The first two steps, shown in figure 30 and figure 31, are identical throughout the scenario-based testing approach, since the scope is only on phishing. This means that "Start" is selected at the beginning of each scenario and then "Choose" for phishing. The technique "Access Token Manipulation Mitigation" in figure 31 does not store any function and only serves demonstration purposes for the expandability of the component "ATT&CK Knowledge Base".

Fig. 30. Scenario "Policy Violation" - "Start" Page

IncidentResponseApp: Restart

Choose Technique

Phishing

Adversaries may send phishing messages to gain access to victim systems. All forms of phishing are electronically delivered social engineering. Phishing can be targeted, known as spearphishing. In spearphishing, a specific individual, company, or industry will be targeted by the adversary. More generally, adversaries can conduct non-targeted phishing, such as in mass malware spam campaigns. Adversaries may send victims emails containing malicious attachments or links, typically to execute malicious code on victim systems or to gather credentials for use of [Valid Accounts] (https://attack.mitre.org/techniques/T1078). Phishing may also be conducted via third-party services, like social media platforms.

Choose

Access Token Manipulation Mitigation

Access tokens are an integral part of the security system within Windows and cannot be turned off. However, an attacker must already have administrator level access on the local system to make full use of this technique; be sure to restrict users and accounts to the least privileges they require to do their job. Any user can also spoof access tokens if they have legitimate credentials. Follow mitigation guidelines for preventing adversary use of [Valid Accounts](https://attack.mitre.org/techniques/T1078). Limit permissions so that users and user groups cannot create tokens. This setting should be defined for the local system account only. GPO: Computer Configuration > [Policies] > Windows Settings > Security Settings > Local Policies > User Rights Assignment: Create a token object. (Citation: Microsoft Create Token) Also define who can create a process level token to only the local and network service through GPO: Computer Configuration > [Policies] > Windows Settings > Security Settings > Local Policies > User Rights Assignment: Replace a process level token. (Citation: Microsoft Replace Process Token) Also limit opportunities for adversaries to increase privileges by limiting Privilege Escalation opportunities.

Choose

Fig. 31. Scenario "Policy Violation" - "Choose Technique" Page

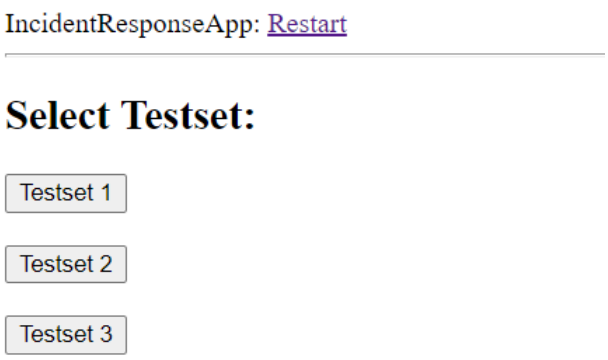


Fig. 32. Scenario "Policy Violation" - "Select Testset" Page

The user selects Testset 1 in this scenario, which has been created for the scenario "Scenario for an incident with policy violation" (see figure 32).

6.4.1 Does the Policy cover the "Grundschutz"? The question "Does the policy cover the "Grundschutz" can be determined after analyzing the user's input, as shown in figures 33, 35, 36. By clicking on "Show Details", further information on the measure can be called up, as shown in figure 34.

IncidentResponseApp: Restart

Review Measures

13.2.1 Richtlinien beim Datenaustausch mit Dritten

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.2 Vertraulichkeitsvereinbarungen

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3 E-Mail

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.1 Festlegung einer Sicherheitspolitik für E-Mail-Nutzung

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.2 Regelung für den Einsatz von E-Mail

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

Fig. 33. Scenario "Policy Violation" - Review Measures I

IncidentResponseApp: Restart

Review Measures

13.2.1 Richtlinien beim Datenaustausch mit Dritten

Show Details

Beim regelmäßigen Datenaustausch mit Dritten ist die Festlegung von Richtlinien bzw. der Abschluss von Vereinbarungen mit allen Beteiligten sinnvoll. Dabei spielt es keine Rolle, wie der Datenaustausch selbst erfolgt (z.B. Datenträgeraustausch, E-Mail, etc.). In einer derartigen Vereinbarung können Angaben zu folgenden Aspekten enthalten sein:

- Bestimmung der Verantwortlichen,
- Benennung von Ansprechpartnern (in technischen, organisatorischen und sicherheitstechnischen Belangen),
- Notwendigkeit eines Non-Disclosure-Agreements (NDA),
- Einstufung von Übertragungsverfahren für klassifizierte Informationen nach [InfoSiG],
- Festlegung der Datennutzung,
- Definition von Anwendungen und Datenformaten,
- Festlegung technischer Übertragungskanäle,
- Definition von Programmen zum Schutz der Daten,
- Festlegung technischer Mittel zur Prüfung der Datenintegrität,
- Definition von Details zu Überprüfungen auf Schadsoftware,
- Festlegung von Fristen zur Datenlöschung,
- Regelung des Managements kryptographischer Schlüssel und Zertifikate, falls erforderlich,
- Einhaltung einschlägiger Gesetze (bspw. Datenschutzgesetz , etc.) und
- Umgang mit Pflichten, die sich aus relevanten Gesetzen (z.B. Datenschutz-Folgenabschätzung) ergeben.

Weitere Punkte, die in eine solche Vereinbarung aufgenommen werden sollten, finden sich in 8.3.2 Datenträgerverwaltung und 8.3.3 Datenträgeraustausch .

Do you consider this measure as relevant for the incident under investigation? ◉ Yes ○ No

If considered relevant, has this measure been applied? ◉ Yes ○ No

Fig. 34. "Review Measures" Page - Detail

13.2.3.3 Sicherer Betrieb eines E-Mail-Servers

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.4 Einrichtung eines Postmasters

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.5 Geeignete Auswahl eines E-Mail-Clients/-Servers

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.6 Sichere Konfiguration der E-Mail-Clients

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.7 Verwendung von „Webmail“ externer Anbieter

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

Fig. 35. Scenario "Policy Violation" - Review Measures II

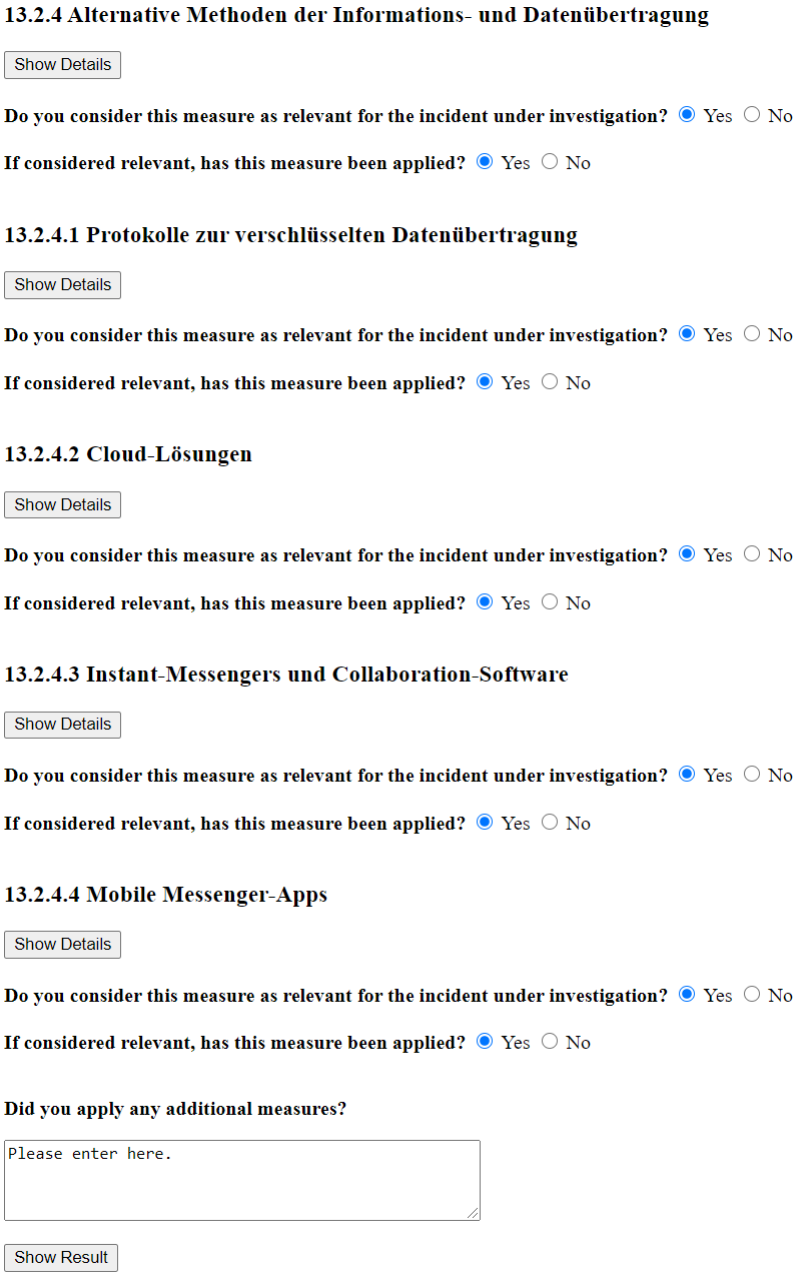


Fig. 36. Scenario "Policy Violation" - Review Measures III

Fig. 37. Scenario "Policy Violation" - Result

In the present scenario, the user has assessed both as relevant and applied all the measures listed to him. Due to this, the policy is rated as "Grundschutz covered by policy" by the prototype, meaning that it aligns with the IT baseline protection (see figure 37), which is correct.

6.4.2 Has there been a Policy Violation? In this step, we investigate the question if there was a policy violation during the incident. For this purpose, the evidence assigned to the testset is loaded and after confirmation, visualized to the user for inspection (see figure 38).

IncidentResponseApp: Restart

Evidence to inspect:

Evidence #1:

User Ticket:

#83783463

Subject:

Security Issue from Email

Description:

I opened a link in an email, although I did not check the sender beforehand. In this link, I have released data that could endanger the security of the company.

Evidence #2:

Norton overview:

Spam Report

Number of mails in spam folder:

9

Most recent spam mail:

<admin@iqgtmipt.com> Make money by answering question

Deleted:

Yes

Evidence #3:

Security Incident Report:

#32432

Date:

Apr-27-2021

Reported by:

Mr. Fally

Organizational Unit:

Controlling

Contact:

+9772828

Location:

Mailing Address

Incident Description:

This morning, an employee received an email. At some point before 09:18, the employee clicked one of the links in that email and was taken to a page that misrepresented itself to be a Google login screen. She entered their Google login details but was presented with an error message so the employee closed the browser window and contacted the recipient to notify them that the employee couldn't access the file. By entering their Google login details into the malicious web page, their account details were compromised. The attackers then used their details to log into their account 07:23 and send the malicious email out to their address book contacts – almost certainly using an automated tool. We traced the access to a Virgin Media IP in the UK but that's probably just an infected computer or other proxy.

Analyze Evidence

Fig. 38. Scenario "Policy Violation" - Inspect Evidences

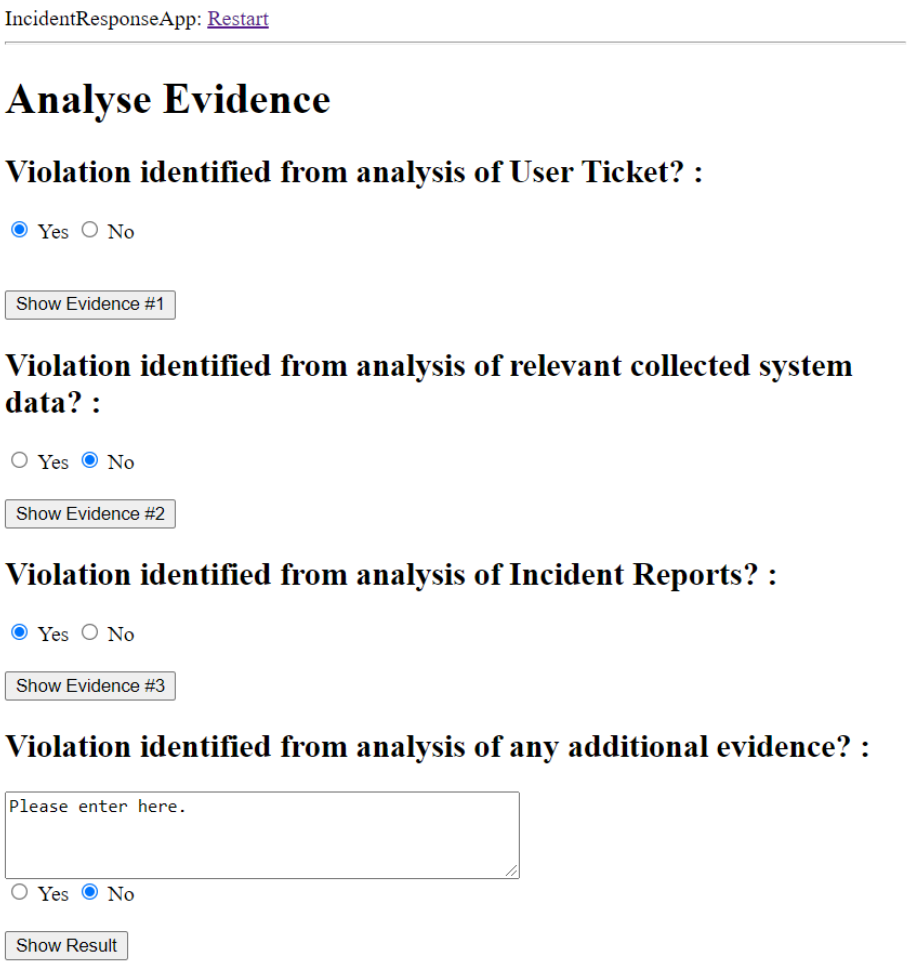


Fig. 39. Scenario "Policy Violation" - Analyse Evidences

Subsequently, the respective evidence is analyzed (see figure 39) and evaluated. In this incident, there is a violation of the policy in Evidence #1 "User Ticket" and there is a violation in Evidence #3 "Security Incident Report." In the case of the first element of evidence, it is assumed that, among other things, the measure "13.2.1 Richtlinien beim Datenaustausch mit Dritten" ("Guidelines for data exchange with third parties") was disregarded. The third element of evidence is to be interpreted in such a way that the measure "13.2.3.1

Festlegung einer Sicherheitspolitik für E-Mail-Nutzung" ("Establishment of a security policy for e-mail use") is violated. There is no additional evidence that hast to be analyzed, therefore the input box is left blank.

Fig. 40. Scenario "Policy Violation" - Result "Policy violated"

As a result, the system returns the result "Incident caused by policy violation", which is correct (see figure 40). If there was any additional evidence that has been violated, it would also be listed here by the system's logic.

6.5 Scenario for an Incident with an insufficient Policy

In this scenario, a testset is loaded that depicts the situation in a company with a security policy not covered by the IT baseline protection principle. Again, a user ticket, a spam report and the underlying incident report serve as evidence. Goal of this test is to check if the algorithm responds correctly where a user enters a policy that does not cover the "Grundschutz". The first two steps, shown in figure 30 and figure 31, are identical throughout the scenario-based testing, since the scope is only on the technique phishing. The user selects Testset 2 in this scenario (see figure 32).

6.5.1 Does the policy cover the "Grundschutz"? The question "Does the policy cover the "basic protection" can be determined after analyzing the user's input, as shown in figure 44. In the present scenario, the user has assessed all measures as relevant but did not apply all of them, as shown in the figures 41, 42, 43. According to the user's input "13.2.1 Richtlinien beim Datenaustausch mit Dritten" ("13.2.1 Guidelines when exchanging data with third parties), "13.2.2 Vertraulichkeitsvereinbarungen" ("13.2.2 Confidentiality agreements") and "13.2.3 Email" have not been applied and furthermore, no additional measure has been applied as evident in 43.

IncidentResponseApp: Restart

Review Measures

13.2.1 Richtlinien beim Datenaustausch mit Dritten

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? ( ) Yes (•) No

13.2.2 Vertraulichkeitsvereinbarungen

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? ( ) Yes (•) No

13.2.3 E-Mail

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? ( ) Yes (•) No

13.2.3.1 Festlegung einer Sicherheitspolitik für E-Mail-Nutzung

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.2 Regelung für den Einsatz von E-Mail

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

Fig. 41. Scenario "Insufficient Policy" - Review Measures I

13.2.3.3 Sicherer Betrieb eines E-Mail-Servers

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.4 Einrichtung eines Postmasters

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.5 Geeignete Auswahl eines E-Mail-Clients/-Servers

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.6 Sichere Konfiguration der E-Mail-Clients

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

13.2.3.7 Verwendung von „Webmail“ externer Anbieter

Show Details

Do you consider this measure as relevant for the incident under investigation? (•) Yes ( ) No

If considered relevant, has this measure been applied? (•) Yes ( ) No

Fig. 42. Scenario ”Insufficient Policy” - Review Measures II

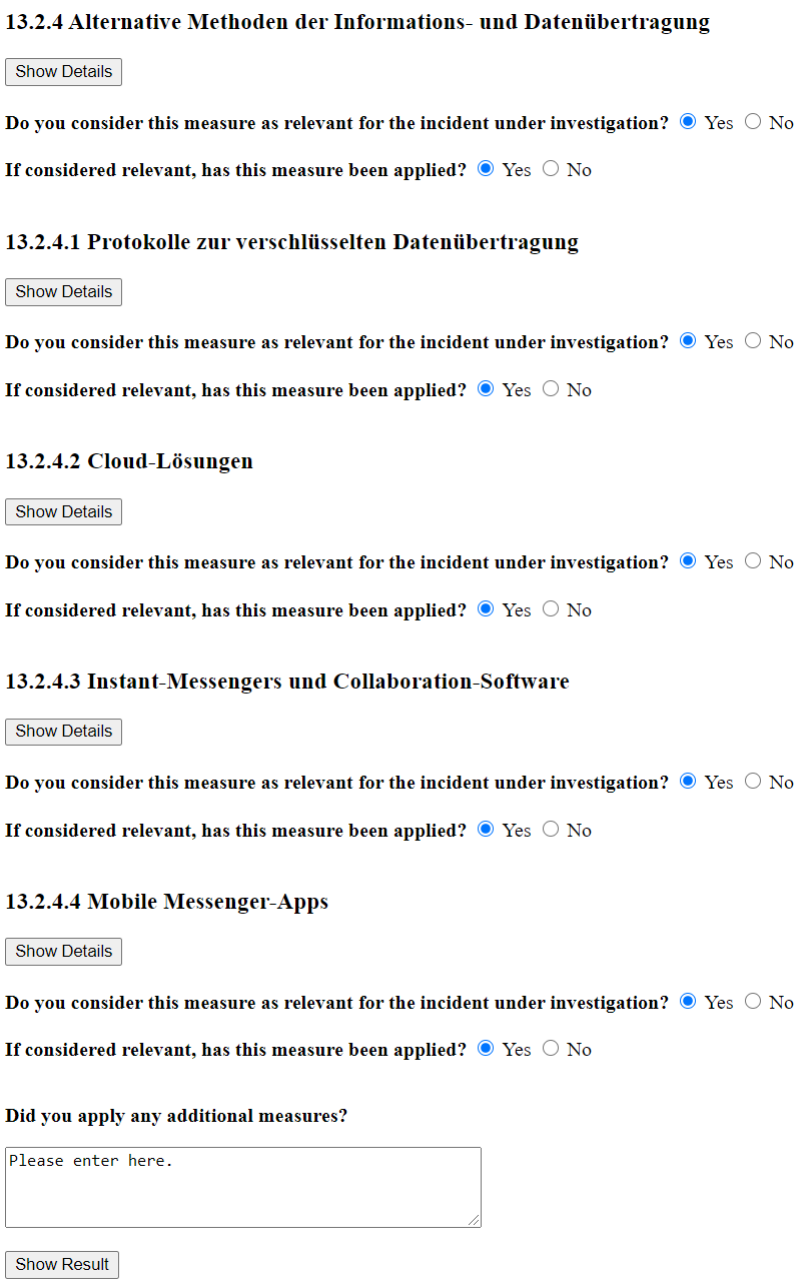


Fig. 43. Scenario "Insufficient Policy" - Review Measures III

Fig. 44. Scenario "Insufficient Policy" - Policy Result

The prototype shows that measures that have not been applied but considered relevant are missing. Due to this, the policy is rated as "Grundschutz not fully covered by policy", meaning that it does not align with the IT baseline protection (see figure 44). When at this point in the process missing measures are covered by an additional measure, which can be input in the previous step, the process returns to the same path as in the previous scenario by clicking on "Yes" and the security policy is considered to be covered. As there are no additional measures which adequately replace the missing "Grundschutz" measures, the user selects "No" and the process ends with the result "Grundschutz not fully Covered by Policy" (see figure 45), which is correct.

Fig. 45. Scenario "Insufficient Policy" - End

As the policy does not cover the Incident, any additional violation of the policy is not assessed. This results in an abbreviated ending of the scenario, as shown in figure 45.

6.6 Scenario of an Incident despite a "Grundschutz" covered Policy and Policy being applied

In this scenario, a testset is loaded that depicts the situation in a company with an IT baseline protection "Grundschutz" covered security policy. Again, a user ticket, a spam report and the underlying incident report serve as evidence. The aim of this test is to check if the algorithm responds correctly when a user enters an Incident that is "Grundschutz" covered by policy and when the policy has been applied. The first two steps, shown in figure 30 and figure 31, are identical throughout the scenario-based testing approach, since the scope is only on phishing. The user selects Testset 3 in this scenario (see figure 32).

6.6.1 Does the policy cover the "Grundschutz"?

The question "Does the policy cover the IT baseline protection('Grundschutz')?" can be determined after analyzing the user's input.

IncidentResponseApp: Restart

Review Measures

13.2.1 Richtlinien beim Datenaustausch mit Dritten

Show Details

Do you consider this measure as relevant for the incident under investigation? ○ Yes ◉ No

If considered relevant, has this measure been applied? ◉ Yes ○ No

13.2.2 Vertraulichkeitsvereinbarungen

Show Details

Do you consider this measure as relevant for the incident under investigation? ○ Yes ◉ No

If considered relevant, has this measure been applied? ◉ Yes ○ No

13.2.3 E-Mail

Show Details

Do you consider this measure as relevant for the incident under investigation? ◉ Yes ○ No

If considered relevant, has this measure been applied? ◉ Yes ○ No

13.2.3.1 Festlegung einer Sicherheitspolitik für E-Mail-Nutzung

Show Details

Do you consider this measure as relevant for the incident under investigation? ◉ Yes ○ No

If considered relevant, has this measure been applied? ◉ Yes ○ No

13.2.3.2 Regelung für den Einsatz von E-Mail

Show Details

Do you consider this measure as relevant for the incident under investigation? ◉ Yes ○ No

If considered relevant, has this measure been applied? ◉ Yes ○ No

Fig. 46. Scenario "Policy covered and applied" - Review Measures I

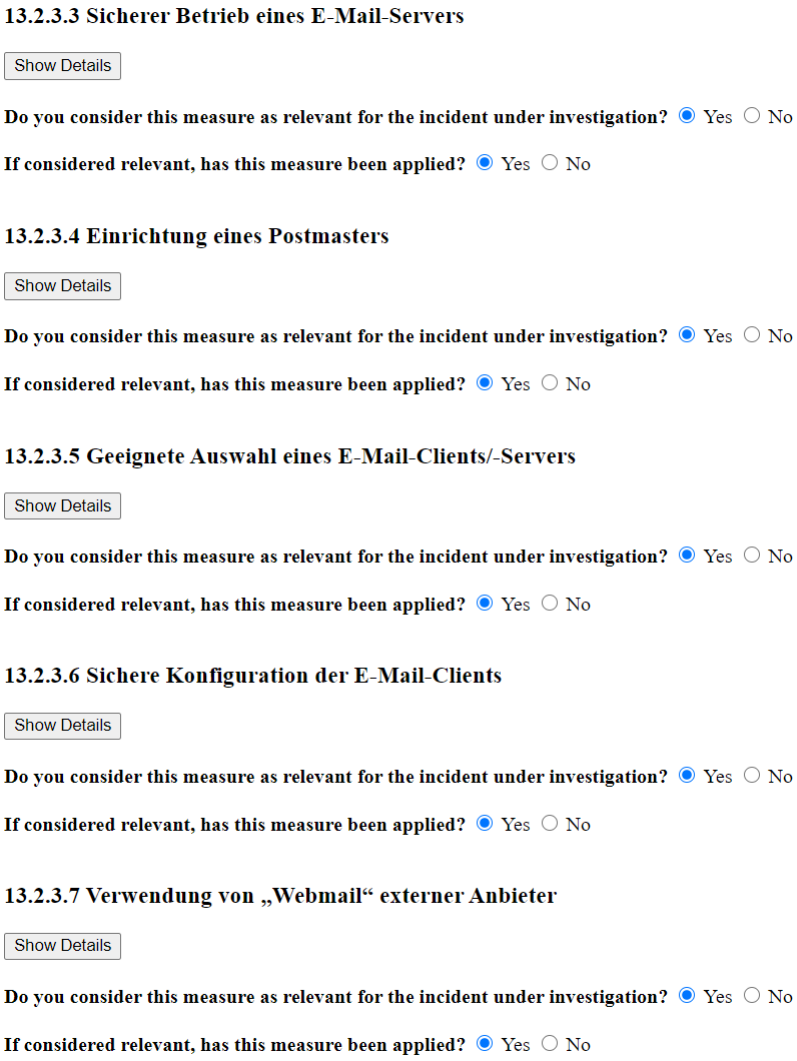


Fig. 47. Scenario "Policy covered and applied" - Review Measures II

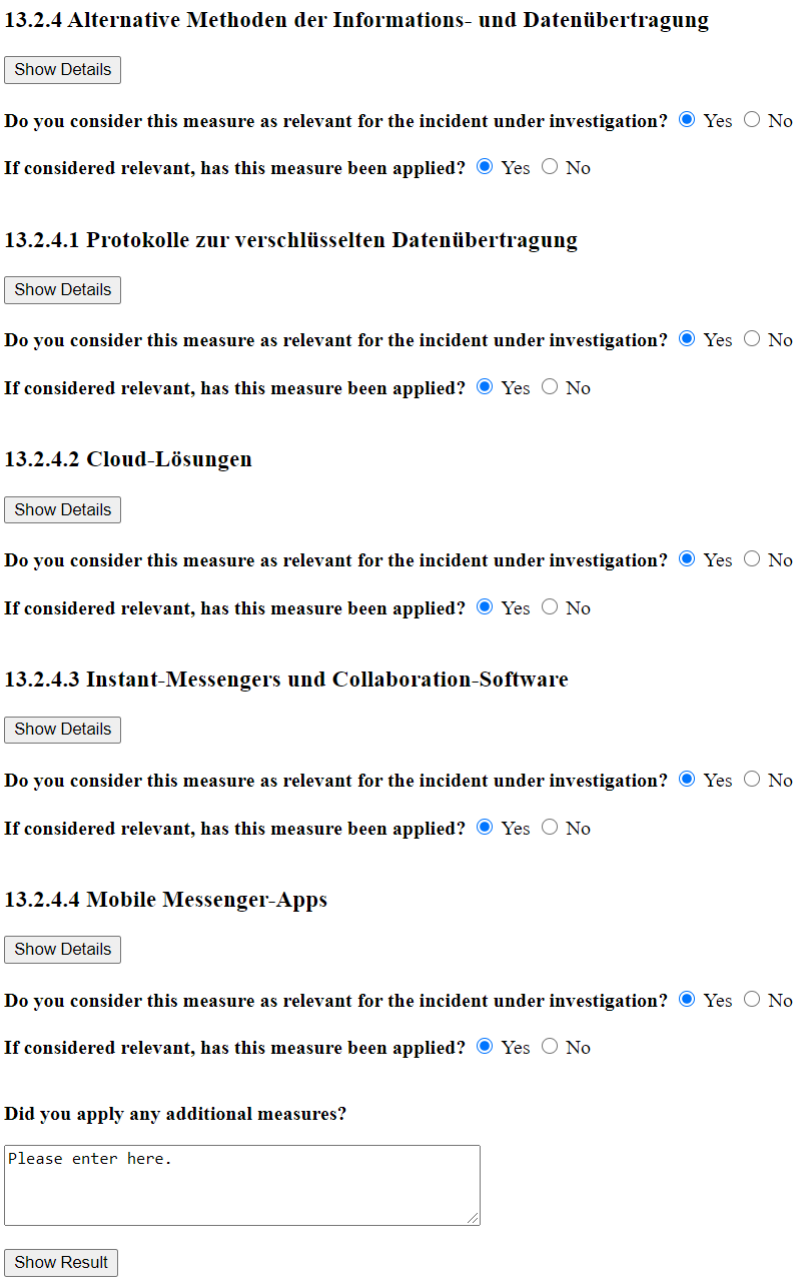


Fig. 48. Scenario "Policy covered and applied" - Review Measures III

Fig. 49. Scenario "Policy covered and applied" - Policy Result

In the present scenario, the user has nearly all measures assessed as relevant, except "13.2.1 Richtlinien beim Datenaustausch mit Dritten" ("13.2.1 Guidelines when exchanging data with third parties) and "13.2.2 Vertraulichkeitsvereinbarungen" ("13.2.2 Confidentiality agreements"). However, all measures considered relevant have been applied, see figures 46, 47, 48. Due to this, the policy is rated as "Grundschutz covered by policy" (see figure 49, which is correct.

6.6.2 Has there been a Policy Violation? In this step, we try to investigate the question of if there was a policy violation during the incident. For this purpose, the evidence assigned to the test set is loaded and after confirmation, visualized to the user for inspection (see figure 50. Subsequently, the respective evidence is analyzed (see figure 39 and evaluated. In this incident, there is no violation of the policy. In Evidence#1, a technical fault was reported, in Evidence#2 an email deleted in the spam folder was reported and in Evidence#3 a suspicious mail was deleted according to the guidelines. All these three elements of evidence are to be understood as compliant with the prescribed behaviour of the policy and no additional evidence has been entered.

IncidentResponseApp: Restart

Evidence to inspect:

Evidence #1:

User Ticket:

#6457

Subject:

Outlook not working

Description:

Since today I can not open my email program. I have followed all security instructions. It must be a technical defect.

Evidence #2:

Norton overview:

Spam Report

Number of mails in spam folder:

1

Most recent spam mail:

<toni@lottototto.com> Earn Money @ Work

Deleted:

Yes

Evidence #3:

Security Incident Report:

#322

Date:

Mar-19-2021

Reported by:

Mr. Hamann

Organizational Unit:

Board of Directors

Contact:

+345456444

Location:

Mailing Address

Incident Description:

This morning, an employee received a suspicious email and deleted it according to our security policy guidelines.

Analyze Evidence

Fig. 50. Scenario "Policy covered and applied" - Inspect Evidences

IncidentResponseApp: Restart

Incident in spite of policy being applied

Your policy concerning "Phishing" might need revision.

These are measures you did not consider relevant:

13.2.1 Richtlinien beim Datenaustausch mit Dritten

Show Details

Beim regelmäßigen Datenaustausch mit Dritten ist die Festlegung von Richtlinien bzw. der Abschluss von Vereinbarungen mit allen Beteiligten sinnvoll. Dabei spielt es keine Rolle, wie der Datenaustausch selbst erfolgt (z.B. Datenträgeraustausch, E-Mail, etc.).
In einer derartigen Vereinbarung können Angaben zu folgenden Aspekten enthalten sein:

- Bestimmung der Verantwortlichen,
- Benennung von Ansprechpartnern (in technischen, organisatorischen und sicherheitstechnischen Belangen),
- Notwendigkeit eines Non-Disclosure-Agreements (NDA),
- Einstufung von Übertragungsverfahren für klassifizierte Informationen nach [InfoSiG],
- Festlegung der Datennutzung,
- Definition von Anwendungen und Datenformaten,
- Festlegung technischer Übertragungskanäle,
- Definition von Programmen zum Schutz der Daten,
- Festlegung technischer Mittel zur Prüfung der Datenintegrität,
- Definition von Details zu Überprüfungen auf Schadsoftware,
- Festlegung von Fristen zur Datenlöschung,
- Regelung des Managements kryptographischer Schlüssel und Zertifikate, falls erforderlich,
- Einhaltung einschlägiger Gesetze (bspw. Datenschutzgesetz , etc.) und
- Umgang mit Pflichten, die sich aus relevanten Gesetzen (z.B. Datenschutz-Folgenabschätzung) ergeben.

Weitere Punkte, die in eine solche Vereinbarung aufgenommen werden sollten, finden sich in 8.3.2 Datenträgerverwaltung und 8.3.3 Datenträgeraustausch .

13.2.2 Vertraulichkeitsvereinbarungen

Show Details

Fig. 51. Scenario "Policy covered and applied" - Result

As a result, the system returns the result "Incident in spite of policy being applied", which is correct (see figure 51). The user's attention is then drawn to the measures he or she has not considered relevant.

6.7 Results of the Scenario-based Testing

Using the three scenarios "Scenario for an Incident with Policy Violation", "Scenario for an insufficient Policy" and "Scenario of an Incident despite a "Grundschutz" covered Policy" within the scenario-based testing approach, the logic of the prototype's algorithm could be verified. As long as the user input is correct, this prototype can be used to determine whether the relevant cause is a violation of policy or an insufficient coverage of the policy. In the first scenario "Scenario for an Incident with Policy Violation", it could be proven that the cause of an incident in a situation where the policy is covered by the IT baseline protection but evidence for the violation of the policy is found was correctly identified. In the second scenario "Scenario for an insufficient Policy" where the recommended measures according to the IT baseline protection were not implemented in the user's policy, the system again delivered the correct result. The policy was classified as not covering the IT baseline protection. In the third and final scenario "Scenario of an Incident despite a 'Grundschutz' covered Policy" in which the policy is covered and no violation of it can be detected, the expected result, the recommendation to revise the policy, is returned. Therefore, it can be stated that by means of the model elaborated in this paper and the prototype based on it, the research question Q1 "How can a process model be constructed and prototypically implemented that helps the user in determining why an incident has resulted in a successful attack?" is answered. In order to answer research question Q2 "What can be achieved with the developed model and what are the limitations?", it is useful to go into more detail about achievements and limitations: With the help of the sequence of tasks developed in the process model, incidents that have already been assigned to a technique can be checked in a targeted manner for non-implementation of prescribed measures, linked to that technique, or violation of the current policy. The process defined in this paper is generic for all techniques and can easily be extended to include other techniques and the related measures due to the architecture of the prototype. Limitations in this context are the coupling of a selection of measures with an attack technique and the verification of the correctness respectively, the dependence of the system on the manual input of the user. In the case of the

first point, there is a conflict of objectives between the completeness of the measures to be checked and the usability of the application. In the current process, it is assumed that all measures not directly related to the attack technique selected for the incident have been implemented according to the IT baseline protection "IT Grundschutz Prinzip". Without this assumption, the user would have to assess all measures of the IT baseline protection and that, in turn, would significantly slow down the process. Regarding the second point, the input of the user in the design phase was considered necessary to develop the process with methods which have already been applied in the field. Thus, the user is expected to check the correctness of his input in order to obtain an adequate result. An incorrect input would otherwise lead to an incorrect result. In order to prevent the result from being falsified by an incorrect input, precise documentation of the incident and the security policy is necessary.

7 Conclusion and Future Work

To sum up, given the increasing number of cyber attacks and the need to combat the problems involved, the demand for countermeasures regarding incidents is given. With the prototypically implemented model in this work, a possibility was successfully demonstrated on how the analysis of an incident by means of an easily accessible, device-independent tool can lead to a targeted diagnosis of the cause of an incident at the security policy level. The approach shows that by means of the developed prototype the process of the follow-up to an incident can be supported. The prototype thus fits into the "Follow-Up" phase of the "Process Model of Incident Handling" presented in the chapter "State of the Art in Literature and Practice". More precisely, it is part of "Policy Revision" and "Policy Violation". The basis for a correct outcome of the prototype is the proper preparation of all evidence by the parallel activities "Documentation" and "Analysis". In conclusion, it can be said that by means of the prototype developed in this work, an approach, with a strong focus on identifying whether a policy violation or an insufficient coverage of the policy is the relevant cause, has been successfully developed to determine why an incident has resulted in a successful attack.

Building on this approach, there are several possibilities to extend the prototype. In general, the prototype offers scalability with regard to the expandability of various attack techniques and their associated measures. The first possibility here is to extend the prototype with further techniques. Since the prototype in its current stage can mainly be classified in the phase of "lessons learned" after an incident, the focus can be placed on two different directions starting from there. First, it would be possible to collect data by using the prototype in the field in order to get a statistical overview of the human error of incidents. In this way it could be easily determined in which section the user needs to improve in regard to the implementation of and compliance with policies. Another option is be to expand the part of the process where the policy violation is analyzed, and to use it for staff training purposes. This is expected to reduce the rate of violated policies through policy awareness. Another possible extension of this concept could be to support the user's input automatically and machine-based. Subsequently, it would be interesting to analyse the user's speech-based input by means of speech recognition in order to simplify the input during the follow-up of an incident. On the other hand, it could be interesting to automate the analysis of evidence provided by the user in order to not be solely dependent on the user's assessment. This is expected to result in an increased efficiency in the follow-up phase.

List of Figures

References

1. Fankar Armash Aslam, Hawa Nabeel Mohammed, and P Lokhande. Efficient way of web development using python and flask. *International Journal of Advanced Research in Computer Science*, 6(2):54–57, 2015.
2. MITRE ATT&CK. Was ist MITRE ATT&CK und wozu dient es? | WeLiveSecurity. *Available at https://www.welivesecurity.com/deutsch/2019/09/03/mitre-att-ck-framework*, Sep 2019. [Online; accessed 1. Dec. 2020].
3. MITRE ATT&CK. Matrix - Enterprise | MITRE ATT&CK®. *Available at https://attack.mitre.org/matrices/enterprise*, Nov 2020. [Online; accessed 1. Dec. 2020].
4. Stephan Augsten. Was ist eine Web App? *Dev-Insider*, Sep 2018.
5. Marco Barros Lourenço and Louis Marinos. ENISA Threat Landscape 2020 - Main Incidents. *Available at https://www.enisa.europa.eu/publications/enisa-threat-landscape-2020-main-incidents*, Feb 2021. [Online; accessed 10. Apr. 2021].
6. Andreea Bendovschi. Cyber-attacks–trends, patterns and security countermeasures. *Procedia Economics and Finance*, 28:24–31, 2015.
7. Seo-hui Byeon and Woo-jong Suh. A study on the government's countermeasures against cyber attacks. In *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 495–499. IEEE, 2020.
8. Eoghan Casey. *Digital evidence and computer crime: Forensic science, computers, and the internet*. Academic press, 2011.
9. Critical Infrastructure Cybersecurity. Framework for improving critical infrastructure cybersecurity. *Framework*, 1(11), 2014.
10. Rachna Dhamija, J Doug Tygar, and Marti Hearst. Why phishing works. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 581–590, 2006.
11. Marites V Fontanilla. Cybercrime pandemic. *Eubios Journal of Asian and International Bioethics*, 30(4):161–165, 2020.
12. Python Software Foundation. Python 3.9.1 Documentation. *Available at https://docs.python.org/3.9*, Mar 2021. [Online; accessed 22. Mar. 2021].
13. Bundesamt für Sicherheit in der Informationstechnik. BSI - DER: Detektion und Reaktion - DER.2.1 Behandlung von Sicherheitsvorfällen, Dec 2020. [Online; accessed 1. Dec. 2020].
14. Bundesamt für Sicherheit in der Informationstechnik. BSI - IT-Grundschutz - BSI-Standards. *Available at https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzStandards/ITGrundschutzStandards_node.html*, Nov 2020. [Online; accessed 26. Nov. 2020].
15. Bundesamt für Sicherheit in der Informationstechnik. BSI-Standard 100-2 - IT-Grundschutz-Vorgehensweise. *Available at https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/ITGrundschutzstandards/BSI-Standard_1002.html*, Apr 2021. [Online; accessed 7. Apr. 2021].
16. Pascal Giessler, Michael Gebhart, Dmitrij Sarancin, Roland Steinegger, and Sebastian Abeck. Best practices for the design of restful web services. In *International Conferences of Software Advances (ICSEA)*, pages 392–397, 2015.
17. Tim Grance, Karen Kent, and Brian Kim. Computer security incident handling guide. *NIST Special Publication*, 800(61):11, 2004.

18. Miguel Grinberg. The Flask Mega-Tutorial Part I: Hello, World! *Available at https://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world*, Apr 2021. [Online; accessed 13. Apr. 2021].
19. Alan R Hevner, Salvatore T March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS quarterly*, pages 75–105, 2004.
20. Proofpoint Inc. Thank you - 2021 Proofpoint State of the Phish Threat Report | Proofpoint US. *Available at https://www.proofpoint.com/us/resources/threat-reports/state-of-phish/thank-you*, Mar 2021. [Online; accessed 25. Mar. 2021].
21. Julian Jang-Jaccard and Surya Nepal. A survey of emerging threats in cybersecurity. *Journal of Computer and System Sciences*, 80(5):973–993, 2014.
22. Valentine Legoy, Marco Caselli, Christin Seifert, and Andreas Peter. Automated retrieval of att&ck tactics and techniques for cyber threat reports. *arXiv preprint arXiv:2004.14322*, 2020.
23. Patrick Mikalef, Ilias Pappas, John Krogstie, and Paul A Pavlou. *Big data and business analytics: A research agenda for realizing business value*. Elsevier, 2020.
24. mitre. cti. *Available at https://github.com/mitre/cti/blob/master/USAGE.md*, Mar 2021. [Online; accessed 22. Mar. 2021].
25. MongoDB. Die beliebteste Datenbank für moderne Apps. *Available at https://www.mongodb.com/de*, Apr 2021. [Online; accessed 14. Apr. 2021].
26. MongoDB. Welcome to the MongoDB Documentation — MongoDB Documentation. *Available at https://docs.mongodb.com*, Mar 2021. [Online; accessed 22. Mar. 2021].
27. NASA. Information security handbook. *Available at https://www.nasa.gov*, Nov 2020. [Online; accessed 24. Nov. 2020].
28. oasis open. cti-python-stix2. *Available at https://github.com/oasis-open/cti-python-stix2*, Mar 2021. [Online; accessed 22. Mar. 2021].
29. oasis open. cti-taxii-client. *Available at https://github.com/oasis-open/cti-taxii-client*, Mar 2021. [Online; accessed 22. Mar. 2021].
30. Capital One. 2019 Capital One Cyber Incident | What Happened | Capital One. *Available at https://www.capitalone.com/facts2019*, Nov 2020. [Online; accessed 28. Nov. 2020].
31. OASIS Open. Introduction to TAXII. *Available at https://oasis-open.github.io/cti-documentation/taxii/intro*, Feb 2021. [Online; accessed 26. Mar. 2021].
32. Pallets. Jinja — Jinja Documentation (2.11.x). *Available at https://jinja.palletsprojects.com/en/2.11.x*, Feb 2021. [Online; accessed 30. Mar. 2021].
33. Pallets. Welcome to Flask — Flask Documentation (1.1.x). *Available at https://flask.palletsprojects.com/en/1.1.x*, Mar 2021. [Online; accessed 22. Mar. 2021].
34. FIPS Pub. Minimum security requirements for federal information and information systems. 2005.
35. Leonard Richardson. beautifulsoup4. *Available at https://pypi.org/project/beautifulsoup4*, Apr 2021. [Online; accessed 14. Apr. 2021].
36. Verizon Enterprise Solutions. 2020 Data Breach Investigations Report. *Available at https://enterprise.verizon.com/resources/reports/2020-data-breach-investigations-report.pdf*, Mar 2021. [Online; accessed 25. Mar. 2021].

37. statista. Infographic: Python Remains Most Popular Programming Language. *Available at https://www.statista.com/chart/21017/most-popular-programming-languages*, Apr 2021. [Online; accessed 14. Apr. 2021].
38. Blake E Strom, Andy Applebaum, Doug P Miller, Kathryn C Nickels, Adam G Pennington, and Cody B Thomas. Mitre att&ck: Design and philosophy. *Technical report*, 2018.
39. Blake E Strom, Joseph A Battaglia, Michael S Kemmerer, William Kupersanin, Douglas P Miller, Craig Wampler, Sean M Whitley, and Ross D Wolf. Finding cyber threats with att&ck-based analytics. *The MITRE Corporation, Bedford, MA, Technical Report No. MTR170202*, 2017.
40. Read the Docs Project. Beautiful Soup Documentation — Beautiful Soup 4.4.0 documentation. *Available at https://beautiful-soup-4.readthedocs.io/en/latest*, Jan 2021. [Online; accessed 22. Mar. 2021].
41. Read the Docs Project. Flask-PyMongo — Flask-PyMongo 2.3.0 documentation. *Available at https://flask-pymongo.readthedocs.io/en/latest*, Jan 2021. [Online; accessed 22. Mar. 2021].
42. Read the Docs Project. STIX 2 Python API Documentation — stix2 2.1.0 documentation. *Available at https://stix2.readthedocs.io/en/latest*, Mar 2021. [Online; accessed 26. Mar. 2021].
43. Patrick Vogel, Thijs Klooster, Vasilios Andrikopoulos, and Mircea Lungu. A low-effort analytics platform for visualizing evolving flask-based python web services. In *2017 IEEE Working Conference on Software Visualization (VISSOFT)*, pages 109–113. IEEE, 2017.
44. Michael E Whitman and Herbert J Mattord. *Management of information security*. Nelson Education, 2013.
45. Davey Winder. Data Breaches Expose 4.1 Billion Records In First Six Months Of 2019. *Forbes*, Aug 2019.
46. Audira Zuraida. Why Should We Choose REST (Client-Server) Model to Develop Web Apps ? *Medium*, Nov 2018.
47. Österreichisches Bundeskanzleramt. Österreichisches Informationssicherheitshandbuch. *Available at https://www.sicherheitshandbuch.gv.at*, Nov 2020. [Online; accessed 14. Nov. 2020].